Variance Reduction of Stochastic Hypergradient Estimation by Mixed Fixed-Point Iteration

Anonymous authors
Paper under double-blind review

Abstract

Hypergradient represents how the hyperparameter of an optimization problem (or inner-problem) changes an outer-cost through the optimized inner-parameter, and it takes a crucial role in hyperparameter optimization, meta learning, and data influence estimation. This paper studies hypergradient computation involving a stochastic inner-problem, a typical machine learning setting where the empirical loss is estimated by minibatches. Stochastic hypergradient estimation requires estimating products of Jacobian matrices of the inner iteration. Current methods struggle with large estimation variance because they depend on a specific sequence of Jacobian samples to estimate this product. This paper overcomes this problem by *mixing* two different stochastic hypergradient estimation methods that use distinct sequences of Jacobian samples. Furthermore, we show that the proposed method enables almost sure convergence to the true hypergradient through the stochastic Krasnosel'skiĭ-Mann iteration. Theoretical analysis demonstrates that, compared to existing approaches, our method achieves lower asymptotic variance bounds while maintaining comparable computational complexity. Empirical evaluations on synthetic and real-world tasks verify our theoretical results and superior variance reduction over existing methods.

1 Introduction

Hypergradient is essential in a wide range of tasks, including bilevel optimization which encompasses hyperparameter optimization (Maclaurin et al., 2015; Franceschi et al., 2017; Liu et al., 2018; Lorraine et al., 2019) and meta learning (Andrychowicz et al., 2016; Finn et al., 2017; Franceschi et al., 2018), as well as an explainability technique called data influence estimation (Koh & Liang, 2017; Hara et al., 2019; Pruthi et al., 2020; Xue et al., 2021). Using a hyperparameter (or outer-parameter) λ and an inner-parameter x that parameterize a scalar cost f and a vector mapping φ, the hypergradient can be defined as

$$\underbrace{\nabla_\lambda f(x(\lambda), \lambda)}_{\text{hypergradient}} \quad \text{with} \quad x(\lambda) = \varphi(x(\lambda), \lambda).$$

A common choice for φ is to set $\varphi(x) = x - \gamma \nabla_x g(x, \lambda)$, with $\gamma \in \mathbb{R}_{++}$ and a strongly-convex function g. In this case, $x(\lambda) = \varphi(x(\lambda), \lambda)$ is equivalent to the minimization problem $x(\lambda) = \arg\min_x g(x, \lambda)$.

Calculating exact hypergradients is prohibitive for large-scale problems as it requires the inverse of a parameter-sized matrix, making their efficient approximations crucial. A method called approximate implicit differentiation (Pedregosa, 2016; Lorraine et al., 2019; Rajeswaran et al., 2019) considers the following form derived from the chain rule and implicit function theorem:

$$\nabla_\lambda f(x(\lambda), \lambda) = \partial_\lambda \varphi (I - \partial_x \varphi)^{-1} \partial_x f + \partial_\lambda f,$$

where the arguments $(x(\lambda), \lambda)$ are omitted and $\partial_x f$ represents $\partial f / \partial x$. Lorraine et al. (2019) proposed the fixed-point method that approximates hypergradients while avoiding explicit calculation of the inverse matrix and Jacobian. The fixed-point method computes a truncated Neumann series approximation with M terms:

$$(I - \partial_x \varphi)^{-1} \approx \sum_{m=0}^{M-1} \partial_x \varphi^m.$$

When the Jacobian matrix $\partial_x \varphi$ is stochastically estimated, a typical scenario where the empirical loss is estimated by minibatches, computing hypergradient becomes even more challenging. To perform such an estimation, Koh & Liang (2017) and Grazzi et al. (2021) used the stochastic fixed-point method which performs the following iteration with independent samples of $\partial_x \varphi$ denoted by $(\hat{A}_m)_{m \in \mathbb{N}}$:

$$\hat{w}_{m+1} = \hat{A}_m \hat{w}_m + \partial_x f. \tag{1}$$

After M iterations, $\hat{w}_M$ estimates the truncated Neumann series as:

$$\hat{w}_M = \sum_{m=0}^{M-1} \left(\prod_{k=m}^{M-1} \hat{A}_k \right) \partial_x f \approx \sum_{m=0}^{M-1} \partial_x \varphi^m \partial_x f.$$

This method can be seen as estimating the Jacobian product by,

$$\partial_x \varphi^{M-m} \approx \hat{A}_{M-1} \hat{A}_{M-2} \cdots \hat{A}_{m+1} \hat{A}_m,$$

for each $m < M$. This approach incurs large estimation variance because each order of the Jacobian product is sampled only once using a specific sequence of Jacobian samples.

This work aims at reducing estimation variance by employing multiple different sequences of Jacobian samples to estimate any order of Jacobian product, without sacrificing computational complexity. We first point out that there are two hypergradient estimators that use different sequences of Jacobian samples to estimate their product: Stochastic fixed-point method (StocFP) (Koh & Liang, 2017; Grazzi et al., 2021) and Stochastic recurrent backpropagation (StocRB) (Ji et al., 2021; Yang et al., 2021). Next, we propose MixedFP, which updates a weighted average of StocFP and StocRB at each step. MixedFP realizes variance reduction by estimating the Jacobian product of a given order using various sample sequences. Furthermore, we improve our MixedFP so it converges almost surely to the true hypergradient by applying the stochastic Krasnosel'skiĭ-Mann iteration (Grazzi et al., 2021; Bravo & Cominetti, 2024).

Additionally, this paper quantifies the variance reduction performance of the proposed method both theoretically and empirically. Our analysis showed that the proposed method improves the expected error compared to the existing stochastic bilevel optimization methods (Grazzi et al., 2021; Ji et al., 2021; Arbel & Mairal, 2022) in terms of one-shot hypergradient estimation. Numerical experiments using real-world data also support the theoretical observations and demonstrate that proposed methods can estimate hypergradients more accurately than the existing methods.

The main contributions of this work are as follows:

- We point out that there are two algorithms in stochastic hypergradient estimation methods that estimate Jacobian products using different sequences of samples: Stochastic fixed-point method (StocFP) and Stochastic recurrent backpropagation (StocRB).
- We propose MixedFP, which reduces the estimation variance by iteratively updating a weighted average of StocFP and StocRB.
- We achieve almost sure convergence to the true hypergradient by applying the stochastic Krasnosel'skiĭ-Mann iteration to MixedFP.
- Through theoretical analysis of error bounds, we demonstrate improvement over previous research (Grazzi et al., 2021; Ji et al., 2021; Arbel & Mairal, 2022).
- Our experiments using real-world data demonstrate a smaller estimation variance of the proposed methods compared to the existing approaches.

2 Related Work

Stochastic hypergradient estimation has been studied in the context of stochastic bilevel optimization (Ghadimi & Wang, 2018; Couellan & Wang, 2016; Ji et al., 2021) to minimize estimation variances

of solutions for both inner- and outer-problems. Ji et al. (2021) and Yang et al. (2021) propose methods that focus on two-time scale updates and stepsize adjustments to accelerate inner-problem optimization with a warm-up strategy. However, these studies do not address the variance reduction of stochastic hypergradient estimation because the noise on the hypergradient estimation is manageable by decreasing stepsizes of their outer-optimization. In contrast, our work specifically aims at reducing the variance of hypergradient estimation itself. This is particularly important for tasks like influence estimation (Koh & Liang, 2017; Khanna et al., 2019), which estimates the impact of excluding training data on performance by a single-shot hypergradient estimation.

There are a few studies that addressed the variance reduction of hypergradient estimation. Grazzi et al. (2021) applied the stochastic Krasnosel'skiĭ-Mann iteration to (1), which is equivalent to solving (2b) in Section 3.1 by stochastic gradient descent (Arbel & Mairal, 2022), to achieve almost sure convergence to the true hypergradient. Moreover, Arbel & Mairal (2022) applies the warm-up to the hypergradient estimation to improve convergence properties. Our MixedFP incorporates their solution as a special case, enabling a more general and effective framework for stochastic hypergradient estimation. Note that this paper does not restrict the context of hypergradients to bilevel optimization, and therefore the warm-up strategy is not employed. Introducing a warm-up is a promising direction for future research, which can enhance the convergence of the bilevel optimization solved by our MixedFP.

3 Hypergradient and Stochastic Approximate Implicit Differentiation

In this section, we first redefine the hypergradient and then introduce two existing methods for computing stochastic hypergradients. We also highlight the difference in the sequences of Jacobian samples used for estimation in these methods.

3.1 Hypergradient and Its Approximation

The hypergradient is the gradient of an outer-cost function $f(x(\lambda), \lambda) \in \mathbb{R}$ with respect to the outer-parameter $\lambda \in \mathbb{R}^{d_\lambda}$, where the inner-parameter $x(\lambda) \in \mathbb{R}^{d_x}$ is defined by the stationary point of a mapping $\varphi : \mathbb{R}^{d_x} \times \mathbb{R}^{d_\lambda} \to \mathbb{R}^{d_x}$. Namely,

$$\nabla_\lambda f(x(\lambda), \lambda) \quad \text{with} \quad x(\lambda) = \varphi(x(\lambda), \lambda).$$

Approximate implicit differentiation (Lorraine et al., 2019; Pedregosa, 2016) uses the implicit function theorem and the chain rule to rewrite the hypergradient in a form that involves an inverse matrix:

$$\nabla_\lambda f = \partial_\lambda \varphi \nabla_x f + \partial_\lambda f, \tag{2a}$$

$$\text{where} \quad \nabla_x f = (I - \partial_x \varphi)^{-1} \partial_x f. \tag{2b}$$

Here and hereafter, for any vector function $h : \mathbb{R}^m \to \mathbb{R}^n$, its partial derivative is denoted by $\partial_x h(x) \in \mathbb{R}^{m \times n}$, and the arguments $(x(\lambda), \lambda)$ are omitted when clear from the context. (2) is justified under the following assumption.

Assumption 1. *For every $\lambda \in \mathbb{R}^{d_\lambda}$, we assume:*

(i) *$\varphi(\cdot, \lambda)$ is a contraction; i.e., there exists a constant $\rho < 1$ such that $\|\partial_x \varphi(x, \lambda)\| \leq \rho$ for any $x \in \mathbb{R}^{d_x}$.*

(ii) *$\varphi(x, \lambda)$ and $f(x, \lambda)$ are differentiable at any $x \in \mathbb{R}^{d_x}$.*

Since calculating the inverse matrix is expensive for a large d_x, Lorraine et al. (2019) proposed the fixed-point method, which approximates (2b) using a truncated Neumann series:

$$\nabla_x f \approx \sum_{m=0}^{M-1} \partial_x \varphi^m \partial_x f. \tag{3}$$

This approximation becomes exact as M approaches infinity.

3.2 Stochastic Approximate Implicit Differentiation

From here on, we assume that we only have access to a stochastic estimator of $\varphi(x, \lambda)$ denoted by $\hat{\varphi}(x, \lambda; \xi)$, where ξ is some random variable whose values lie within a measurable space $\mathcal{X}$. For convenience, we introduce the following notations:

$$A = \partial_x \varphi(x(\lambda), \lambda), \qquad \hat{A} = \partial_x \hat{\varphi}(x(\lambda), \lambda; \xi), \qquad \hat{A}_m = \partial_x \hat{\varphi}(x(\lambda), \lambda; \xi_m), \qquad c = \partial_x f(x(\lambda), \lambda),$$

where $(\xi_m)_{m \in \mathbb{N}}$ are independent copies of ξ. We assume that $\hat{\varphi}$ satisfies the following conditions:

Assumption 2. *For every $x \in \mathbb{R}^{d_x}$ and $\lambda \in \mathbb{R}^{d_\lambda}$,*

(i) *$\hat{\varphi}(x, \lambda; \xi)$ is an unbiased estimator of $\varphi(x, \lambda)$; i.e., $\mathbb{E}[\hat{\varphi}(x, \lambda; \xi)] = \varphi(x, \lambda), \forall x, \lambda$.*

(ii) *$\hat{\varphi}$ is differentiable with respect to the first and second arguments at any $\xi \in \mathcal{X}$.*

Note that, unlike common settings of stochastic bilevel optimization (e.g., Ghadimi & Wang (2018)), we do not consider estimation of f. This is solely for clarity, as our primary focus is on the estimation error caused by the inverse matrix approximation with stochastic $\hat{\varphi}$, and our primary distinction from the previous studies (Arbel & Mairal, 2022; Grazzi et al., 2020) is the improvement on this approximation.

3.2.1 Stochastic Fixed-Point Method (StocFP)

The studies by Grazzi et al. (2021) and Koh & Liang (2017) employ an iteration that we call Stochastic fixed-point method (StocFP), a stochastic variant of the fixed-point method (Lorraine et al., 2019).

$$\hat{w}_0 = c \tag{4a}$$

$$\text{For } m = 0, \dots, M-1\text{:}$$

$$\quad \hat{w}_{m+1} = \hat{A}_m \hat{w}_m + c \tag{4b}$$

The iteration (4b) accumulates the products of Jacobian samples and their sum simultaneously. Assumption 2 guarantees that $\hat{w}_M$ is an unbiased estimator of the right-hand side of (3). A key advantage of this method is that $\hat{A}_m w_m \in \mathbb{R}^{d_x}$ can be calculated in $O(d_x)$ time using the Hessian-vector product technique. The total computation time is therefore $O(M d_x)$ with a memory requirement of $O(d_x)$.

3.2.2 Stochastic Recurrent Backpropagation (StocRB)

Ji et al. (2021) independently proposed a method for estimating $\nabla_x f$ using a different form of iterations:

$$\hat{y}_0 = 0 \in \mathbb{R}^{d_x}, \ \hat{u}_0 = c \tag{5a}$$

$$\text{For } m = 0, \dots, M-1\text{:}$$

$$\begin{aligned} &\hat{y}_{m+1} = \hat{y}_m + \hat{u}_m \\ &\hat{u}_{m+1} = \hat{A}_m \hat{u}_m \end{aligned} \tag{5b}$$

The final output $\hat{y}_M$ is an estimator of $\nabla_x f$. Unlike StocFP (4b), this method divides the computation of (3) into the product of Jacobians estimated by $\hat{u}_m$ and their sum estimated by $\hat{y}_m$. We refer to (5) as Stochastic recurrent backpropagation (StocRB) because this method can be understood as backpropagation from f with respect to x that has recurrently passed through $\hat{\varphi}$ for M times.

While both StocFP (4) and StocRB (5) provide the same hypergradient approximation in expectation, they yield different biases. This is because they estimate the expected Jacobian product using different sequences of Jacobian samples:

Remark 1. *For any $m = 1, \dots, M$, StocFP estimates A^m using $\hat{A}_{M-1} \cdots \hat{A}_{M-m}$, while StocRB uses $\hat{A}_{m-1} \cdots \hat{A}_0$ to estimate the same value:*

Stochastic Fixed-Point Method (**StocFP**) (4)

$$
\begin{aligned}
\hat{w}_M = \quad & c && (= A^0 c) \\
& + \hat{A}_{M-1} c && (\approx A^1 c) \\
& + \hat{A}_{M-1}\hat{A}_{M-2} c && (\approx A^2 c) \\
& \vdots \\
& + \hat{A}_{M-1}\hat{A}_{M-2}\cdots\hat{A}_1 c && (\approx A^{M-1} c) \\
& + \hat{A}_{M-1}\hat{A}_{M-2}\cdots\hat{A}_1\hat{A}_0 c && (\approx A^{M} c)
\end{aligned}
$$

Stochastic Recurrent Backprop. (**StocRB**) (5)

$$
\begin{aligned}
\hat{y}_M = & && c && (= A^0 c) \\
& + && \hat{A}_0 c && (\approx A^1 c) \\
& + && \hat{A}_1\hat{A}_0 c && (\approx A^2 c) \\
& \vdots \\
& + && \hat{A}_{M-2}\cdots\hat{A}_1\hat{A}_0 c && (\approx A^{M-1}) \\
& + && \hat{A}_{M-1}\hat{A}_{M-2}\cdots\hat{A}_1\hat{A}_0 c && (\approx A^{M} c)
\end{aligned}
$$

4 Variance Reduction of Stochastic Approximate Implicit Differentiation

The proposed method is based on two ideas. First, in Section 4.1, we show that the estimation variance can be reduced by a weighted average-like iteration between StocRB and StocFP. Second, Section 4.2 explains that unbiased hypergradient estimation is achieved by using the stochastic Krasnosel'skiĭ-Mann iteration (Bravo & Cominetti, 2024). For clarity, proofs of all theorems and lemmas are deferred in the appendix.

4.1 Mixed Fixed-Point Iteration (MixedFP)

Our MixedFP mixes StocFP and StocRB, motivated by the observation in Remark 1 that the sequences of Jacobian samples used for estimation are different from each other.

We introduce a parameter called the mixing rate $\alpha \in [0, 1]$ and derive the following MixedFP algorithm:

$$
\begin{aligned}
& \hat{v}_0 = 0 \in \mathbb{R}^{d_x},\ \hat{w}_0 = \hat{u}_0 = c && \text{(6a)} \\
& \text{For } m = 0, \dots, M-1\text{:} \\
& \quad \hat{v}_{m+1} = \alpha(\hat{v}_m + \hat{u}_m) + (1-\alpha)\hat{w}_m \\
& \quad \hat{w}_{m+1} = \hat{A}_m \hat{w}_m + c && \text{(6b)} \\
& \quad \hat{u}_{m+1} = \hat{A}_m \hat{u}_m
\end{aligned}
$$

Here, $\hat{w}_m$ and $\hat{u}_m$ are the same as in StocFP (4) and StocRB (5), respectively. $\hat{v}_{m+1}$ estimates the hypergradient (3) and is equal in expectation to $\hat{w}_m$ and $\hat{y}_m$. The update iteration for $\hat{v}_m$ is a slightly modified weighted average of $\hat{w}_m$ in (4b) and $\hat{y}_m$ in (5b). In fact, one can verify that $\hat{v}_{m+1} = \hat{w}_m$ when $\alpha = 0$ and $\hat{v}_{m+1} = \hat{y}_m$ when $\alpha = 1$. It is noteworthy that the update for $\hat{v}_{m+1}$ is not a simple weighted average of $\hat{w}_m$ and $\hat{y}_m$, because it uses the previously computed value $\hat{v}_m$ instead of $\hat{y}_m$. This recursive structure allows MixedFP to estimate Jacobian products using diverse sequences of Jacobian samples:

Remark 2. *When $\alpha \in (0, 1)$ and given $m \in \{1, \dots, M\}$, $\hat{v}_M$ uses $\hat{A}_{k-1}\cdots\hat{A}_{k-m}$ for any k such that $m \le k \le M$ to estimate A^m. More specifically, with some $\sum_{k=m}^{M} a_{m,k-1} = 1$ such that $a_{m,k} > 0$*

Mixed Fixed-Point Iteration (**MixedFP**) (6)

$$
\begin{aligned}
\hat{v}_{M+1} = \quad & c && (= A^0 c) \\
& + \Big(\underbrace{a_{1,M-1}\boxed{\hat{A}_{M-1}} + \cdots + a_{1,0}\boxed{\hat{A}_0}}_{M \text{ terms}} \Big) c && (\approx A^1 c)
\end{aligned}
$$

$$
\begin{aligned}
&+ \Bigg(\underbrace{a_{2,M-1} \boxed{\hat{A}_{M-1}\hat{A}_{M-2}} + \cdots + a_{2,1} \boxed{\hat{A}_1\hat{A}_0}}_{M-1 \text{ terms}} \Bigg) c && (\approx A^2 c) \\
&\vdots \\
&+ \left(a_{M-1,M-1} \boxed{\hat{A}_{M-1} \cdots \hat{A}_1} + a_{M-1,M-2} \boxed{\hat{A}_{M-2} \cdots \hat{A}_0} \right) c && (\approx A^{M-1} c) \\
&+ \boxed{\hat{A}_{M-1}\hat{A}_{M-2} \cdots \hat{A}_1\hat{A}_0}\, c && (\approx A^M c)
\end{aligned}
$$

This diversity in the products of Jacobian samples cannot be obtained by simply taking a weighted average of $\hat{w}_m$ and $\hat{y}_m$. Additionally, thanks to the Hessian-vector product, MixedFP enjoys the same complexity as StocFP and StocRB in both time and space.

Under the following regularity conditions, we show that the expected error of MixedFP improves over StocFP and StocRB.

Assumption 3. *For every $\lambda \in \mathbb{R}^{d_\lambda}$, we assume that:*

(i) $\mathbb{E}[\|\hat{\varphi}(x, \lambda; \xi)\|^2] < \infty$ *for every* $x \in \mathbb{R}^{d_x}$.

(ii) *There exists a constant* $\hat{\rho} < 1$ *such that* $\|\partial_x \hat{\varphi}(x, \lambda; \xi)\| \leq \hat{\rho}$ *for every* $x \in \mathbb{R}^{d_x}$ *and* $\xi \in \mathcal{X}$.

(iii) *The function* $f(\cdot, \lambda)$ *is Lipschitz continuous with some constant* $L_f \geq 0$.

Theorem 4.1 (MixedFP)**.** *Suppose Assumptions 1 to 3 hold and* $\alpha \in [0, 1]$*, then for any* $m \geq 0$*,*

$$
\mathbb{E}\left[\|\hat{v}_m - \nabla_x f\|^2\right] \leq \begin{cases} \sigma_1 + O\left(\rho_1^m\right) & \text{if } \alpha \in \{0, 1\}, \\ \dfrac{1 + \alpha_1}{(1 + \sqrt{\alpha_1})^2}\sigma_1 + O\left(\max\{\rho_1, \alpha_1\}^m\right) & \text{otherwise,} \end{cases}
$$

where

$$
\sigma_1 = \frac{L_f^2 \hat{\rho}^2}{(1 - \hat{\rho})^2 (1 - \rho^2)}, \quad \rho_1 = \hat{\rho}^2, \quad \alpha_1 = \alpha^2.
$$

This result shows that the scale of the non-decaying term, $\frac{1+\alpha_1}{(1+\sqrt{\alpha_1})^2} \leq 1$, becomes smaller as α increases, but when α is too large, more specifically when $\alpha > \hat{\rho}$, the decay rate becomes slower than $O(\rho_1^m)$. Recalling that $\alpha \in \{0, 1\}$ corresponds to StocFP and StocRB, this indicates that α chosen in $0 < \alpha < \hat{\rho}$ improves the non-decaying term and the decaying term over these conventional methods. Table 1 summarizes these findings. This analysis also suggests that for sufficiently large m, the smallest error is achieved when α is infinitely close to, but strictly less than, one. However, for finite m, a smaller α may be optimal if accelerating the decaying term outweighs the reduction of the non-decaying term. These observations align with the empirical results in Section 5.1.

4.2 MixedFP-KM: Application of the Stochastic Krasnosel'skiĭ-Mann (KM) Iteration

This section shows that the stochastic KM iteration can be applied to MixedFP, enabling almost sure convergence to the true hypergradient.

The stochastic KM iteration (Grazzi et al., 2021; Bravo & Cominetti, 2024) is an algorithm that finds the fixed point of a contraction mapping using its unbiased estimation:

Theorem 4.2 (Stochastic KM iteration (Grazzi et al., 2021))**.** *Let ζ be a random variable with values in a measurable space $\mathcal{Z}$ and $(\zeta_m)_{m \in \mathbb{N}}$ be independent copies of ζ. Let $T : \mathbb{R}^d \to \mathbb{R}^d$ be a contraction mapping*

Method	Base estimator	KM iteration	Expected ℓ_2 error bound
Grazzi et al. (2021) with $\eta = 1$	StocFP		$\sigma_1 + O(\rho_1^m)$
Ji et al. (2021)	StocRB		$\sigma_1 + O(\rho_1^m)$
MixedFP	**StocFP & StocRB**		$\frac{1+\alpha_1}{(1+\sqrt{\alpha_1})^2}\sigma_1 + O(\rho_1^m)$
Grazzi et al. (2021) with constant η	StocFP	✓	$\sigma_\eta + O(\rho_\eta^m)$
MixedFP-KM with constant η	**StocFP & StocRB**	✓	$\frac{1+\alpha_\eta}{(1+\sqrt{\alpha_\eta})^2}\sigma_\eta + O(\rho_\eta^m)$

Table 1: Comparison of the upper bound of the expected ℓ_2 error in stochastic hypergradient estimation. Our MixedFP and MixedFP-KM can reduce non-decaying errors by $\frac{1+\alpha_\eta}{(1+\sqrt{\alpha_\eta})^2} < 1$ times smaller than their counterparts. All symbols are defined in Theorems 4.1 and 4.5 and we assume $\alpha \leq \hat{\rho}$ in this table.

and suppose a mapping $\hat{T} : \mathbb{R}^d \times \mathcal{Z} \to \mathbb{R}^d$ *satisfies* $\mathbb{E}[\hat{T}(s;\zeta)] = T(s)$ *and* $\mathbb{V}[\hat{T}(s;\zeta)] \leq \sigma_1 + \sigma_2\|T(s) - s\|$ *for any* $s \in \mathbb{R}^d$*. Consider the stochastic Krasnosel'skiĭ-Mann iteration defined by*

$$s_{m+1} = (1 - \eta_m)s_m + \eta_m \hat{T}(s_m; \zeta_m), \tag{7}$$

where the stepsize η_m *satisfy*

$$\sum_{k=0}^{\infty} \eta_k = \infty, \qquad \sum_{k=0}^{\infty} \eta_k^2 < \infty, \qquad \eta_m \leq \frac{1}{1+\sigma_2}, \qquad \forall m \in \mathbb{N}, \tag{8}$$

Then the sequence $(s_m)_{m\in\mathbb{N}}$ *converges almost surely to the fixed point of* T*.*

Next, we show that MixedFP is a contraction mapping. To do this, we reformat MixedFP into the following mapping of a concatenated vector.

$$\hat{F}(z;\xi) = \hat{B}z + d, \tag{9}$$
$$\text{where} \quad z = \begin{bmatrix} v \\ w \\ u \end{bmatrix} \in \mathbb{R}^{3d_x}, \quad \hat{B} = \begin{bmatrix} \alpha I & \alpha I & (1-\alpha)I \\ O & \hat{A} & O \\ O & O & \hat{A} \end{bmatrix} \in \mathbb{R}^{3d_x \times 3d_x}, \quad d = \begin{bmatrix} 0 \\ c \\ 0 \end{bmatrix} \in \mathbb{R}^{3d_x}.$$

The mapping $\hat{F}$ has the following favorable properties that enable the application of the stochastic KM iteration.

Lemma 4.3. *When Assumptions 1 and 2 hold,* $F(z) = \mathbb{E}[\hat{F}(z;\xi)]$ *with* $\alpha \in [0, 1)$ *is a contraction mapping that has the unique fixed point* $[\nabla_x f^\top \ \nabla_x f^\top \ 0^\top] \in \mathbb{R}^{3d_x}$*.*

By applying (7) to the mapping $\hat{F}$ in (9), we obtain the following algorithm, which we name MixedFP-KM.

$v_0 = 0 \in \mathbb{R}^{d_x}$, $w_0 = u_0 = c$
For $m = 0, \ldots, M-1$:
$\quad v_{m+1} = (1-\eta_m)v_m + \eta_m(\alpha(v_m + u_m) + (1-\alpha)w_m)$
$\quad w_{m+1} = (1-\eta_m)w_m + \eta_m(\hat{A}_m w_m + c)$
$\quad u_{m+1} = (1-\eta_m)u_m + \eta_m \hat{A}_m u_m$

From Theorem 4.2 and Lemma 4.3, it follows that v_m converges to $\nabla_x f$ with appropriate scheduling of η_m.

Theorem 4.4. *Let* $\alpha \in [0, 1)$*,* $q = \max\{\alpha, \rho\}$*, and* $\hat{q} = \max\{\alpha, \hat{\rho}\}$*. When Assumptions 1 to 3 hold and* η_m *satisfies* (8) *with* $\sigma_2 = 2(\hat{q}^2 + q^2)/(1-q)$*, then*

$$\lim_{m\to\infty} v_m = \nabla_x f \quad a.s.$$

For fixed stepsizes, the error can be bounded by an exponentially decaying term and a non-decaying term.

Theorem 4.5 (MixedFP-KM)**.** *Let $\alpha \in [0, 1)$. When $\eta_m = \eta < \frac{1}{1-\hat{\rho}}$ and Assumptions 1 to 3 hold, then for any $m \geq 0$,*

$$\mathbb{E}\left[\|v_m - \nabla_x f\|^2\right] \leq \frac{1+\alpha_\eta}{\left(1+\sqrt{\alpha_\eta}\right)^2}\sigma_\eta + O\left(\max\{\rho_\eta, \alpha_\eta\}^m\right),$$

where

$$\sigma_\eta = \frac{\eta L_f^2 \hat{\rho}^2}{(1-\hat{\rho})^2(2-\eta(1-\rho))(1-\rho)}, \quad \rho_\eta = (1-\eta+\eta\hat{\rho})^2, \quad \alpha_\eta = (1-\eta+\eta\alpha)^2.$$

For comparison, we show our result for the existing method (Grazzi et al., 2021), which applies the stochastic KM iteration to StocFP[1].

Theorem 4.6 (Stochastic KM iteration of StocFP)**.** *Suppose $\eta_m = \eta < \frac{1}{1-\hat{\rho}}$ and Assumptions 1 to 3 hold. Then for any $m \geq 0$,*

$$\mathbb{E}\left[\|w_m - \nabla_x f\|^2\right] \leq \sigma_\eta + O\left(\rho_\eta^m\right),$$

where σ_η and ρ_η are as defined in Theorem 4.5.

The non-decaying terms in both Theorems 4.5 and 4.6 can be made arbitrarily small by the choice of η at the cost of a slow decay rate. The observations obtained from comparing these two theorems are essentially the same as those obtained in Theorem 4.1. That is, when $0 < \alpha < \hat{\rho}$, both the non-decaying and decaying term improve over the existing method (Grazzi et al., 2021). This was also empirically confirmed in Section 5.2.

5 Experiments

In this section, we first investigate the relationship between the newly introduced parameter α and the convergence properties of MixedFP. Then, we compare the hypergradient estimation accuracy of our proposed method with existing approaches in various real-world task settings.

5.1 Effect of Mixing Rate

5.1.1 Settings

This experiment evaluates the hypergradient estimation error in a synthetic setting where we can explicitly control the Jacobian matrix samples. Specifically, we compute the hypergradient for the following case:

$$f(x(\lambda), \lambda) = c^\top x(\lambda) + d^\top \lambda, \qquad \varphi(x, \lambda) = (I - \gamma\hat{H})x + B\lambda,$$

where $\hat{H} \in \mathbb{R}^{d_x \times d_x}$ is a random variable sampled from a discrete uniform distribution $\hat{H} \sim \mathrm{Uniform}(\{H_1, \ldots, H_n\})$. Each matrix in $H_1, \ldots, H_n$ was constructed such that its eigenvalues follow the uniform distribution over $[0, 1-\epsilon]$ with a small constant $\epsilon \in \mathbb{R}_{++}$ to meet our assumptions. We tested different γ over $(0, 1]$, which corresponds to varying the values of ρ and $\hat{\rho}$. The other coefficients $c \in \mathbb{R}^{d_x}$, $d \in \mathbb{R}^{d_\lambda}$, and $B \in \mathbb{R}^{d_x \times d_\lambda}$ were generated by sampling their elements independently from a uniform distribution over $[0, 1]$. Note that x and λ vanish upon differentiation, thus their values do not affect the hypergradient estimation. The experiment was run 100 times with different seed values for sampling $\hat{H}$.

5.1.2 Results and Discussion

Fig. 1 shows the mean squared error of hypergradients at different γ and α.

Estimations with $\alpha = 0.99$ or $\alpha = 0.999$ consistently outperform both $\alpha = 0$ (StocFP) and $\alpha = 1$ (StocRB) across different values of γ, highlighting the benefit of our mixing strategy. The results suggest that the optimal value of α is close to but not exactly one, aligning with the observation from Theorem 4.1. The superior performance of $\alpha = 0.99$ over the larger $\alpha = 0.999$ is also consistent with Theorem 4.1: for finite iterations, a slightly smaller α can be optimal when faster decay outweighs the increased non-decaying term.

[1]Theorem 4.6 differs from a result obtained from Grazzi et al. (2021, Theorem 4.2). As discussed in Appendix B.2, this is solely for a fair comparison with our method, and in some cases Theorem 4.6 is even tighter than the original bound.

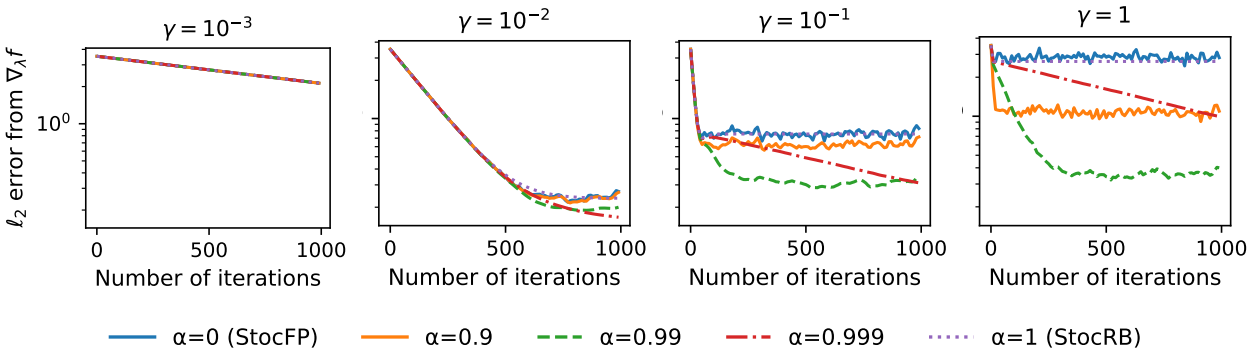

Figure 1: Mean L2 error of hypergradients estimated by MixedFP using different values of α. We adopted a synthetic inner-problem $\varphi(x(\lambda), \lambda) = (I - \gamma\hat{H})x(\lambda) + B\lambda$ and plotted results obtained with different γ.

Additionally, we observe that increasing γ accelerates the decay rate while enlarging the non-decay term. Higher values of γ result in smaller values for both ρ and $\hat{\rho}$, which aligns with a theoretical finding in Theorem 4.1 that implies a smaller $\hat{\rho}$ leads to a larger non-decaying term σ_1 and a faster decay rate ρ_1.

Furthermore, some settings show interesting patterns in their error curves. When $\gamma = 10^{-1}$ with $\alpha = 0.999$ or 0.99, the error seems to decay at two distinct rates: initially faster, then slower after a certain number of iterations (around $m = 50$). This seems to reflect $O(\max\{\rho_1, \alpha_1\}^m)$; The error comprises two distinct decay rates, and for sufficiently large m, the slower decay rate eventually dominates.

5.2 Comparison with Existing Approaches

In this section, we consider several real-world machine learning settings to compare the hypergradient estimation accuracy of our proposed method with existing approaches across various combinations of the dataset, inner-problem, and outer-cost.

5.2.1 Tasks

We evaluated the performance of hypergradient estimation methods for the following task settings:

- **Synthetic problem** evaluates the estimation accuracy of hypergradients of the synthetic outer-cost with the synthetic inner-problem used in Section 5.1.
- **Hyperparameter optimization** performs binary classification on Adult Income dataset (Becker & Kohavi, 1996), where the outer-parameter is a vector of regularization coefficients, each corresponding to a dimension of the inner-parameter.
- **Influence estimation** conducts multi-class classification on Fashion-MNIST (Xiao et al., 2017), where the outer-parameter is a vector of loss masks, each assigned to a corresponding sample.
- **Meta learning** tackles a regression problem on the California Housing dataset (Pace & Barry, 1997), where the outer-parameter determines a biased regularization term.

Except for the synthetic setting, we adopt $\hat{\varphi}(x, \lambda; \xi) = x - \gamma\nabla_x g(x, \lambda; \xi)$ and define f and g for each setting as follows. We denote training inputs and outputs by ξ_{in} and ξ_{out}, and use $(\xi'_{\text{in}}, \xi'_{\text{out}}) \in \Xi_{\text{val}}$ for the independent validation dataset.

Hyperparameter optimization refers to the hypergradient estimation problem in a bilevel problem where the regularization coefficients serve as hyperparameters. Specifically, as in Grazzi et al. (2021), we compute the hypergradient for an outer-problem that finds the optimal hyperparameters λ whose elements are different

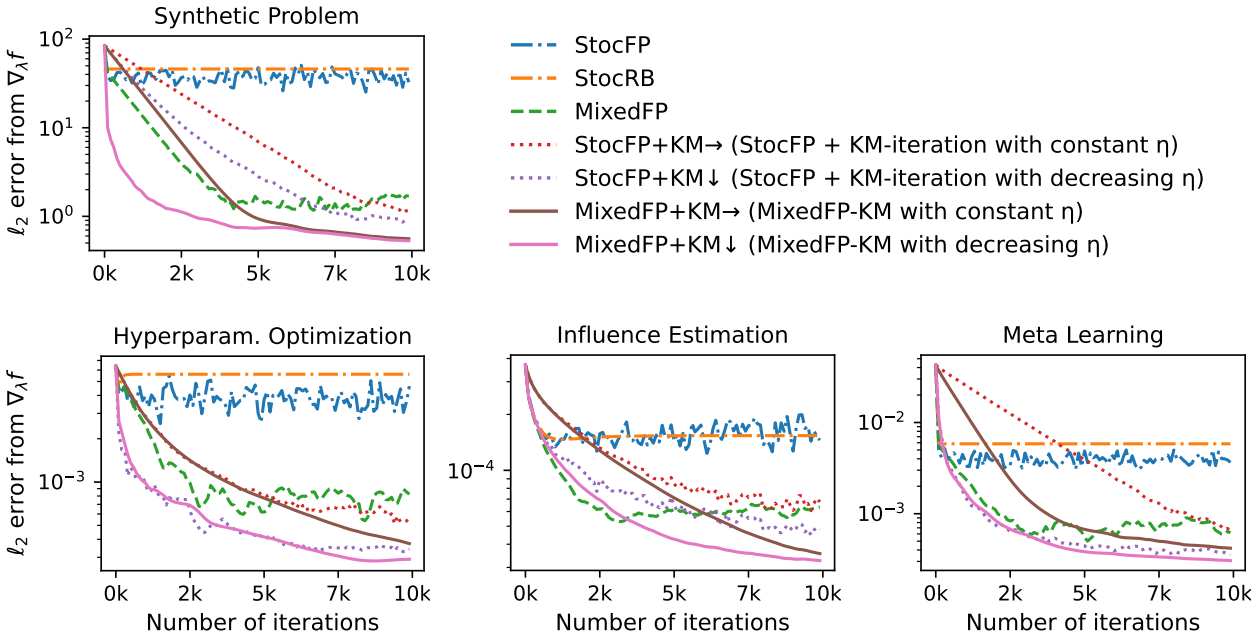

Figure 2: Comparison of stochastic hypergradient estimation methods evaluated on different tasks. Each line shows a mean squared error over 10 trials with different Jacobian sample sequences $(\hat{A}_m)_{m\in\mathbb{N}}$. Method-specific parameters were optimized by grid search, selecting settings with minimal average error over the last 1,000 iterations.

regularization coefficients for each dimension of x:

$$f(x,\lambda) = \sum_{(\xi'_{\text{in}},\xi'_{\text{out}})\in\Xi_{\text{val}}} \text{BCE}(\xi'_{\text{out}}, \xi'^{\top}_{\text{in}} x), \qquad g(x,\lambda;(\xi_{\text{in}},\xi_{\text{out}})) = \text{BCE}(\xi_{\text{out}}, \xi^{\top}_{\text{in}} x) + \frac{1}{2} x^{\top} \text{diag}(\lambda) x, \tag{10}$$

where $\xi_{\text{in}} \in \mathbb{R}^{14}$, $\xi_{\text{out}} \in \{0,1\}$, $\text{BCE}(p,q) = \log\left(1+\exp(-pq)\right)$, and $\text{diag}(\lambda)$ denotes the diagonal matrix whose diagonal entries are the elements of λ.

Influence estimation (Koh & Liang, 2017; Khanna et al., 2019) is a task that estimates the change in performance when a sample in the inner-problem is excluded. This is equivalent to estimating the hypergradient of an outer-parameter that determines the loss mask of each sample. That is, with the sample index $j = \text{Uniform}(\{1,\dots,N\})$ with dataset size N,

$$f(x,\lambda) = \sum_{(\xi'_{\text{in}},\xi'_{\text{out}})\in\Xi_{\text{val}}} \text{CE}(\xi'_{\text{out}}, \xi'^{\top}_{\text{in}} x), \qquad g(x,\lambda;j) = \lambda_j \text{CE}(\xi_{\text{out},j}, \xi^{\top}_{\text{in},j} x), \tag{11}$$

where, $\text{CE}(p,q) = -\sum_{c=1}^{10} p_c \log\left(\exp(q_c)/\sum_{c'=1}^{10}\exp(q_{c'})\right)$ and $(\xi_{\text{in},j}, \xi_{\text{out},j}) \in \mathbb{R}^{784}\times\{0,1\}^{10}$ denotes the j-th sample in the training dataset. The outer-parameter $\lambda \in \mathbb{R}^N$ is a ones-vector whose j-th element $\lambda_j = 1$ is multiplied by the loss of the j-th sample. The j-th element of the hypergradient $-\nabla_{\lambda_j} f \in \mathbb{R}$ is a linear approximation of the change in f when the j-th sample is excluded from the training dataset.

Meta learning is the following task adopted in Grazzi et al. (2020), which simplifies the task called implicit meta learning (Denevi et al., 2019; Rajeswaran et al., 2019).

$$f(x,\lambda) = \frac{1}{2} \sum_{(\xi'_{\text{in}},\xi'_{\text{out}})\in\Xi_{\text{val}}} \|\xi'^{\top}_{\text{in}} x(\lambda) - \xi'_{\text{out}}\|^2, \qquad g(x,\lambda;(\xi_{\text{in}},\xi_{\text{out}})) = \frac{1}{2}\|\xi^{\top}_{\text{in}} x - \xi_{\text{out}}\|^2 + \frac{\mu}{2}\|x-\lambda\|^2, \tag{12}$$

where $(\xi_{\text{in}},\xi_{\text{out}}) \in \mathbb{R}^8\times\mathbb{R}$, $\mu \in \mathbb{R}_{++}$, and $\lambda \in \mathbb{R}^{d_x}$.

5.2.2 Baselines and Procedure

Hypergradient estimation is performed using the five methods: `StocFP`, `StocRB`, `MixedFP`, StocFP with KM-iteration (denoted as `StocFP+KM→` and `StocFP+KM↓`), and MixedFP with KM-iteration (denoted as `MixedFP+KM→` and `MixedFP+KM↓`). `KM→` and `KM↓` represent constant and linearly decreasing η_m, respectively, and we used $\eta_m = \beta/(\delta + m)$ as the decreasing stepsize following Grazzi et al. (2021).

For each setting and method, we conducted the experiment 10 times using different seed values to vary the sequence $(\hat{A}_m)_{m<M}$, and calculated the mean squared error ℓ_2 with respect to $\nabla_x f$. Since our primary focus is on reducing the variance of $\nabla_x f$, we computed the exact values of $\partial_x f$, $\partial_\lambda f$, and $\partial_\lambda \varphi$ in (2) to ensure that no additional randomness was introduced apart from $(\hat{A}_m)_{m<M}$. Furthermore, for this purpose, we approximated $x(\lambda)$ as accurately as possible by minimizing g using full-batch optimizations until they converge. For the outer-parameter λ, we used task-specific initialization values without performing outer-optimization.

To ensure a fair comparison between the baseline and proposed methods, we use the parameters that yield the best performance for each method. These parameters include η, α, β and δ, which were determined via grid search by minimizing the average error over the last 1,000 steps.

More detailed settings are deferred to our appendix.

5.2.3 Results and Discussion

In all tasks, `MixedFP+KM↓` yielded the best performance (Table 1). Our theoretical analysis quantifies the improvements achieved under a fixed stepsize (by comparing Theorem 4.5 and Theorem 4.6), while for a decreasing stepsize, we have only established asymptotic convergence (Theorem 4.4). Nevertheless, these results demonstrate that, in practice, MixedKM can also improve estimation accuracy with decreasing stepsizes. When using KM-iteration with constant stepsizes, `MixedFP+KM→` achieves lower error than its counterpart `StocFP-KM→`, which supports our results of Theorem 4.5. Among methods without KM-iteration, `StocFP` and `StocRB` tend to reach plateaus in the early iterations, while `MixedFP` continues to reduce error over a relatively larger number of steps.

6 Conclusion

We presented a stochastic hypergradient estimation method that reduces variance incurred by estimation error of the product of the inner-iteration's Jacobian matrices. Our proposed MixedFP combines existing StocFP and StocRB estimators to leverage their distinct sequences of Jacobian samples, thereby reducing estimation variance without increasing computational complexity. Moreover, by applying the stochastic KM iteration into MixedFP, we established almost sure convergence to the true hypergradient. Empirical evaluations on synthetic and real-world tasks, including hyperparameter optimization, data influence estimation, and meta learning, validated our theoretical findings and demonstrated improved estimation accuracy in practical scenarios.

Future work will focus on extending our analysis to the case of decreasing stepsizes and exploring efficiency improvements in the application, including bilevel optimization.

References

Marcin Andrychowicz, Misha Denil, Sergio Gomez, Matthew W Hoffman, David Pfau, Tom Schaul, Brendan Shillingford, and Nando De Freitas. Learning to learn by gradient descent by gradient descent. In *Advances in Neural Information Processing Systems*, pp. 3981–3989, 2016.

Michael Arbel and Julien Mairal. Amortized implicit differentiation for stochastic bilevel optimization. In *International Conference on Learning Representations*, 2022.

Barry Becker and Ronny Kohavi. Adult. UCI Machine Learning Repository, 1996. DOI: 10.24432/C5XW20.

Mario Bravo and Roberto Cominetti. Stochastic fixed-point iterations for nonexpansive maps: Convergence and error bounds. *SIAM Journal on Control and Optimization*, 62(1):191–219, 2024.

Nicolas Couellan and Wenjuan Wang. On the convergence of stochastic bi-level gradient methods. *Optimization*, pp. 13833, 2016.

Giacomo Denevi, Carlo Ciliberto, Riccardo Grazzi, and Massimiliano Pontil. Learning-to-learn stochastic gradient descent with biased regularization. *arXiv preprint arXiv:1903.10399*, 2019.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, pp. 1126–1135, 2017.

Luca Franceschi, Michele Donini, Paolo Frasconi, and Massimiliano Pontil. Forward and reverse gradient-based hyperparameter optimization. In *International Conference on Machine Learning*, pp. 1165–1173, 2017.

Luca Franceschi, Paolo Frasconi, Saverio Salzo, Riccardo Grazzi, and Massimiliano Pontil. Bilevel programming for hyperparameter optimization and meta-learning. In *International Conference on Machine Learning*, pp. 1563–1572, 2018.

Saeed Ghadimi and Mengdi Wang. Approximation methods for bilevel programming. *arXiv preprint arXiv:1802.02246*, 2018.

Riccardo Grazzi, Luca Franceschi, Massimiliano Pontil, and Saverio Salzo. On the iteration complexity of hypergradient computation. *arXiv preprint arXiv:2006.16218*, 2020.

Riccardo Grazzi, Massimiliano Pontil, and Saverio Salzo. Convergence properties of stochastic hypergradients. In *International Conference on Artificial Intelligence and Statistics*, pp. 3826–3834, 2021.

Satoshi Hara, Atsushi Nitanda, and Takanori Maehara. Data cleansing for models trained with sgd. In *Advances in Neural Information Processing Systems*, 2019.

Kaiyi Ji, Junjie Yang, and Yingbin Liang. Bilevel optimization: Convergence analysis and enhanced design. In *International Conference on Machine Learning*, pp. 4882–4892, 2021.

Rajiv Khanna, Been Kim, Joydeep Ghosh, and Sanmi Koyejo. Interpreting black box predictions using fisher kernels. In *International Conference on Artificial Intelligence and Statistics*, pp. 3382–3390. PMLR, 2019.

Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *International Conference on Machine Learning*, pp. 1885–1894, 2017.

Hanxiao Liu, Karen Simonyan, and Yiming Yang. Darts: Differentiable architecture search. *arXiv preprint arXiv:1806.09055*, 2018.

Jonathan Lorraine, Paul Vicol, and David Duvenaud. Optimizing millions of hyperparameters by implicit differentiation. *arXiv preprint arXiv:1911.02590*, 2019.

Dougal Maclaurin, David Duvenaud, and Ryan Adams. Gradient-based hyperparameter optimization through reversible learning. In *International Conference on Machine Learning*, pp. 2113–2122, 2015.

R Kelley Pace and Ronald Barry. Sparse spatial autoregressions. *Statistics & Probability Letters*, 33(3): 291–297, 1997.

Fabian Pedregosa. Hyperparameter optimization with approximate gradient. In *International Conference on Machine Learning*, pp. 737–746, 2016.

Garima Pruthi, Frederick Liu, Satyen Kale, and Mukund Sundararajan. Estimating training data influence by tracing gradient descent. In *Advances in Neural Information Processing Systems*, pp. 19920–19930, 2020.

Aravind Rajeswaran, Chelsea Finn, Sham M Kakade, and Sergey Levine. Meta-learning with implicit gradients. In *Advances in Neural Information Processing Systems*, pp. 113–124, 2019.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

Yihao Xue, Chaoyue Niu, Zhenzhe Zheng, Shaojie Tang, Chengfei Lyu, Fan Wu, and Guihai Chen. Toward understanding the influence of individual clients in federated learning. In *the AAAI Conference on Artificial Intelligence*, pp. 10560–10567, 2021.

Junjie Yang, Kaiyi Ji, and Yingbin Liang. Provably faster algorithms for bilevel optimization. In *Advances in Neural Information Processing Systems*, pp. 13670–13682, 2021.

A Theoretical Analysis of MixedFP-KM

In this section, we provide a detailed analysis of the proposed MixedFP-KM. We first recall the iteration schemes, then present key lemmas giving explicit forms of the iterates and the corresponding error bounds.

A.1 Recap of the Iterations

Let $\eta \in \mathbb{R}_{++}$ be a stepsize. We denote by (v_m), (w_m), and (u_m) the sequences generated by the following iteration:

$$v_0 = 0, \tag{13}$$
$$w_0 = c, \tag{14}$$
$$u_0 = c, \tag{15}$$

and for $m = 0, 1, 2, \ldots$

$$v_{m+1} = (1-\eta)v_m + \eta\left(\alpha\left(u_m + v_m\right) + \overline{\alpha} w_m\right) = r v_m + \eta\left(\alpha u_m + \overline{\alpha} w_m\right), \tag{16}$$
$$w_{m+1} = (1-\eta)w_m + \eta\left(\hat{A}_m w_m + c\right) = \hat{P}_m w_m + \eta c, \tag{17}$$
$$u_{m+1} = (1-\eta)u_m + \eta\left(\hat{A}_m u_m\right) = \hat{P}_m u_m, \tag{18}$$

where $\alpha \in [0,1]$, $\overline{\alpha} = 1-\alpha$, $r = 1-\eta(1-\alpha)$, and $\hat{P}_m = I - \eta\left(I - \hat{A}_m\right)$.

A.2 Proof of Asymptotical Convergence of MixedFP-KM

This section provides the proofs for Lemma 4.3 and Theorem 4.4.

Lemma 4.3. *When Assumptions 1 and 2 hold, $F(z) = \mathbb{E}[\hat{F}(z;\xi)]$ with $\alpha \in [0,1)$ is a contraction mapping that has the unique fixed point $[\nabla_x f^\top \ \nabla_x f^\top \ 0^\top] \in \mathbb{R}^{3d_x}$.*

Proof. Let $B = \mathbb{E}[\hat{B}]$. Because B is a block triangle matrix, its eigenvalues are those of diagonal blocks, that are ensured to lie in $[0,1)$ by Assumption 1 and $\alpha \in [0,1)$. Hence, T is a contraction mapping which has a unique fixed point.

Let the first, second, and third blocks of the fixed point $z^* = T(z^*)$ be v^*, w^*, and u^*. From Assumption 1, we know that $u^* = 0$. Under Assumption 2, w^* satisfies $w^* = Aw^* + c$ thus $w^* = (I-A)^{-1}c = \nabla_x f$. Finally, since we know that $v^* = \alpha v^* + (1-\alpha)\nabla_x f$, which is true only when $v^* = \nabla_x f$ since $1-\alpha \neq 0$. □

Theorem 4.4. *Let $\alpha \in [0,1)$, $q = \max\{\alpha, \rho\}$, and $\hat{q} = \max\{\alpha, \hat{\rho}\}$. When Assumptions 1 to 3 hold and η_m satisfies (8) with $\sigma_2 = 2(\hat{q}^2 + q^2)/(1-q)$, then*

$$\lim_{m\to\infty} v_m = \nabla_x f \quad a.s.$$

Proof. Let $q = \|B\|$ and $\hat{q} = \|\hat{B}\|$, where $B = \mathbb{E}[\hat{B}]$. Since B is a block triangular matrix, its eigenvalues are determined by those of its diagonal blocks, giving us $q = \max\{\alpha, \rho\}$ from Assumption 1. Similarly, we can establish that $\hat{q} = \max\{\alpha, \hat{\rho}\}$ under Assumption 3. Applying Theorem 4.3 from Grazzi et al. (2021), we obtain $\sigma_2 = 2\frac{\hat{q}^2+q^2}{(1-q)}$. Therefore, combining Theorem 4.2 and Lemma 4.3, we conclude that the statement holds true. □

A.3 Closed-Form Expressions of MixedFP-KM

We first derive closed-form expressions (or general forms) for the iterates $(v_m), (w_m)$, and (u_m) under the MixedFP-KM iteration (16) to (18).

Lemma A.1 (General-Term Representation)**.** *Under the MixedFP-KM iteration* (13) *to* (18)*, for $m \geq 0$ we have:*

$$v_{m+1} = \eta \sum_{k=0}^{m} r^{m-k} \left(\alpha u_k + \overline{\alpha} w_k\right), \tag{19}$$

$$w_{m+1} = \left(\prod_{j=0}^{m} \hat{P}_j\right) c + \eta \sum_{k=0}^{m} \left(\prod_{j=k+1}^{m} \hat{P}_j\right) c, \tag{20}$$

$$u_{m+1} = \left(\prod_{j=0}^{m} \hat{P}_j\right) c, \tag{21}$$

where $r = 1 - \eta\,(1-\alpha)$ and $\hat{P}_m = I - \eta\left(I - \hat{A}_m\right)$.

Proof. We prove each identity by induction.

(Proof of (19)) For $m = 0$, (16) matches (19) since

$$v_1 = \eta r^0 \left(\alpha u_0 + \overline{\alpha} w_0\right).$$

Assume (19) holds for some $m \geq 0$. Then from (16),

$$\begin{aligned} v_{m+1} &= r v_m + \eta \left(\alpha u_m + \overline{\alpha} w_m\right) \\ &= r \left(\eta \sum_{k=0}^{m-1} r^{m-1-k} \left(\alpha u_k + \overline{\alpha} w_k\right)\right) + \eta \left(\alpha u_m + \overline{\alpha} w_m\right) \\ &= \eta \sum_{k=0}^{m} r^{m-k} \left(\alpha u_k + \overline{\alpha} w_k\right). \end{aligned}$$

This completes the inductive step.

(Proof of (20)) For $m = 0$, from (17), we have

$$w_1 = \hat{P}_0 c + \eta c,$$

since $w_0 = c$. This is consistent with (20) for $m = 0$ because

$$w_1 = \left(\prod_{j=0}^{0} \hat{P}_j\right) c + \eta \sum_{k=0}^{0} \left(\prod_{j=k+1}^{0} \hat{P}_j\right) c = \hat{P}_0 c + \eta c.$$

Assume (20) holds for some $m \geq 0$. Then from (17),

$$\begin{aligned} w_{m+1} &= \hat{P}_m w_m + \eta c \\ &= \hat{P}_m \left(\prod_{j=0}^{m-1} \hat{P}_j \right) c + \hat{P}_m \eta \sum_{k=0}^{m-1} \left(\prod_{j=k+1}^{m-1} \hat{P}_j \right) c + \eta c \\ &= \left(\prod_{j=0}^{m} \hat{P}_j \right) c + \eta \sum_{k=0}^{m} \left(\prod_{j=k+1}^{m} \hat{P}_j \right) c. \end{aligned}$$

This completes the proof for w_{m+1}.

(Proof of (21)) The derivation is analogous to that of w_m, except that no extra ηc term added. For $m = 0$, (21) and (18) are consistent because both are

$$u_1 = \hat{P}_0 c.$$

from $u_0 = c$.

The inductive step is essentially the same as above, giving

$$u_{m+1} = \hat{P}_m u_m = \left(\prod_{j=0}^{m} \hat{P}_j \right) c.$$

Hence (21) holds. □

A.4 General Error Terms

This section considers differences between (v_m, w_m, u_m) and their deterministic counterparts $(\overline{v}_m, \overline{w}_m, \overline{u}_m)$, namely,

$$\overline{v}_0 = 0, \tag{22}$$
$$\overline{w}_0 = c, \tag{23}$$
$$\overline{u}_0 = c, \tag{24}$$

and for $m = 0, 1, 2, \ldots$

$$\overline{v}_{m+1} = (1-\eta)\,\overline{v}_m + \eta\left(\alpha\left(\overline{u}_m + \overline{v}_m\right) + \overline{\alpha w}_m\right) = r\overline{v}_m + \eta\left(\alpha\overline{u}_m + \overline{\alpha w}_m\right), \tag{25}$$
$$\overline{w}_{m+1} = (1-\eta)\,\overline{w}_m + \eta\left(A\overline{w}_m + c\right) = P_m\overline{w}_m + \eta c, \tag{26}$$
$$\overline{u}_{m+1} = (1-\eta)\,\overline{u}_m + \eta\left(A\overline{u}_m\right) = P_m\overline{u}_m, \tag{27}$$

where $P = I - \eta(I - A)$. The next lemma gives a useful expression for these differences.

Lemma A.2 (General Error Terms)**.** *Under the MixedFP-KM iteration* (13) *to* (18) *and its deterministic counterpart* (22) *to* (27)*, for $s \geq 1$ we have: Let $P = I - \eta(I - A)$. Then:*

$$v_s - \overline{v}_s = \eta \sum_{k=0}^{s-1} r^{s-1-k} \left(\alpha\left(u_k - \overline{u}_k\right) + \overline{\alpha}\left(w_k - \overline{w}_k\right)\right), \tag{28}$$

$$w_s - \overline{w}_s = \eta \sum_{j=0}^{s-1} P^{s-1-j} (\hat{A}_j - A) \left\{ \left(\prod_{i=0}^{j-1} \hat{P}_i \right) c + \eta \sum_{k=0}^{j-1} \left(\prod_{i=k+1}^{j-1} \hat{P}_i \right) c \right\}, \tag{29}$$

$$u_s - \overline{u}_s = \eta \sum_{k=0}^{s-1} P^{s-1-k} \left(\hat{A}_k - A \right) \left(\prod_{i=0}^{k-1} \hat{P}_i \right) c. \tag{30}$$

Proof. We give the argument for each of the three differences, using induction and the results in Lemma A.1.

(Difference for v_m) As the direct consequence of Lemma A.1, the general term of the deterministic case is given by

$$\overline{v}_{s+1} = \eta \sum_{k=0}^{s} r^{s-k} \left(\alpha \overline{u}_k + \overline{\alpha w}_k \right), \tag{31}$$

One can directly obtain (28) by comparing the general terms (19) and (31).

(Difference for w_s) By subtracting (26) from (17), the following recursion is shown to hold:

$$\begin{aligned} w_{s+1} - \overline{w}_{s+1} &= \hat{P}_s w_s - P\overline{w}_s + \eta c - \eta c \\ &= P\left(w_s - \overline{w}_s\right) + (\hat{P}_s - P) w_s \\ &= P\left(w_s - \overline{w}_s\right) + \eta(\hat{A}_s - A) w_s. \end{aligned} \tag{32}$$

Hence, (29) is consistent with (32) when $s = 0$, because both are

$$w_1 - \overline{w}_1 = \eta(\hat{A}_0 - A)c, \tag{33}$$

using the fact that $w_0 = \overline{w}_0 = c$. Assume (29) holds for some $s \geq 1$. Then, from (32) and (20),

$$\begin{aligned} w_{s+1} - \overline{w}_{s+1} &= P\left(w_s - \overline{w}_s\right) + \eta(\hat{A}_s - A) w_s \\ &= \eta P \sum_{j=0}^{s-1} P^{s-1-j} (\hat{A}_j - A) \left\{ \left(\prod_{i=0}^{j-1} \hat{P}_i \right) c + \eta \sum_{k=0}^{j-1} \left(\prod_{i=k+1}^{j-1} \hat{P}_i \right) c \right\} \\ &\quad + \eta(\hat{A}_s - A) \left\{ \left(\prod_{i=0}^{s-1} \hat{P}_i \right) c + \eta \sum_{k=0}^{s-1} \left(\prod_{i=k+1}^{s-1} \hat{P}_i \right) c \right\} \\ &= \eta \sum_{j=0}^{s} P^{s-j} (\hat{A}_j - A) \left\{ \left(\prod_{i=0}^{j-1} \hat{P}_i \right) c + \eta \sum_{k=0}^{j-1} \left(\prod_{i=k+1}^{j-1} \hat{P}_i \right) c \right\} \end{aligned}$$

This completes the proof for $s + 1$.

(Difference for u_s) Similar steps yield (30), now without the extra c term in the iteration for u_{m+1}. Namely, substraction of (18) from (27) yields

$$u_{s+1} - \overline{u}_{s+1} = P\left(u_s - \overline{u}_s\right) + \eta(\hat{A}_s - A) u_s,$$

and one can show that this is consistent with (30) when $s = 0$. When (30) is assumed to be true for some $s \geq 1$, then (30) holds true for $s + 1$ because

$$u_{s+1} - \overline{u}_{s+1} = P\left(u_s - \overline{u}_s\right) + \eta(\hat{A}_s - A) u_s \tag{34}$$

$$= \eta P \sum_{j=0}^{s-1} P^{s-1-j} (\hat{A}_j - A) \left(\prod_{i=0}^{j-1} \hat{P}_i \right) c + \eta(\hat{A}_s - A) \left(\prod_{i=0}^{s-1} \hat{P}_i \right) c \tag{35}$$

$$= \eta \sum_{j=0}^{s} P^{s-j} (\hat{A}_j - A) \left(\prod_{i=0}^{j-1} \hat{P}_i \right) c. \tag{36}$$

□

A.5 Bounding Variance Norm of MixedFP-KM

We prove the bound on $\mathbb{E}\left[\|v_m - \overline{v}_m\|^2\right]$ starting with proving a useful lamma.

Lemma A.3. *Let $x > 1$ and let $m \geq 0$ be an integer. Then*

$$\sum_{k=0}^{m-1} (2(m-k)-1)x^k \leq \frac{x^m(x+1)}{(1-x)^2}.$$

Proof. First, observe that

$$\sum_{k=0}^{m-1} (2(m-k)-1)x^k = (2m-1)\sum_{k=0}^{m-1} x^k - 2\sum_{k=0}^{m-1} kx^k.$$

Since $x \neq 1$, the geometric series partial sum is

$$\sum_{k=0}^{m-1} x^k = \frac{1-x^m}{1-x}.$$

A standard result (or by differentiating the geometric series) gives

$$\sum_{k=0}^{m-1} kx^k = \frac{x - mx^m + (m-1)x^{m+1}}{(1-x)^2},$$

Multiplying the desired inequality

$$\sum_{k=0}^{m-1} (2(m-k)-1)\, x^k \leq \frac{x^m(x+1)}{(1-x)^2}$$

by $(1-x)^2$ (which is positive for $x > 1$) reduces the task to proving

$$(2m-1)(1-x^m)(1-x) - 2\left(x - mx^m + (m-1)x^{m+1}\right) \leq x^m(x+1).$$

A short computation shows that the left-hand side simplifies to

$$(2m-1) - (2m+1)x + x^m + x^{m+1},$$

so that the inequality becomes

$$2m - 1 \leq (2m+1)x.$$

Since $x > 1$, the last inequality holds, and the proof is complete. □

Lemma A.4. *Suppose $0 < \eta < \frac{1}{1-\hat{\rho}}$, $\alpha \leq \rho$, and Assumption 3 hold, then for any $m \geq 0$,*

$$\mathbb{E}\left[\|v_m - \overline{v}_m\|^2\right] \leq 2L_f^2\hat{\rho}^2 \frac{\eta}{(1-\hat{\rho})^2\,(2-\eta(1-\rho))\,(1-\rho)\,(2-\eta(1-\alpha))^2} + O\left((1-\eta(1-\hat{\rho}))^{2m}\right).$$

Proof. Below, we assume $\mathbb{E}\left[\|\hat{A}_j - A\|^2\right] \neq 0$, because otherwise $\mathbb{E}\left[\|v_m - \overline{v}_m\|^2\right] = 0$, which is the case of the desired inequality. By letting

$$X_{k,j} = P^{k-1-j}, \quad \hat{X}_{k,j} = \prod_{i=j}^{k-1} \hat{P}_i, \quad \hat{Y}_k = \eta\sum_{j=0}^{k-1}\left(\prod_{i=j+1}^{k-1}\hat{P}_i\right) = \eta\sum_{j=0}^{k-1}\hat{X}_{k,j+1}$$

and by recalling Lemma A.2, $v_s - \overline{v}_s$ can be written as

$$\begin{aligned}
v_m - \overline{v}_m &= \eta\sum_{k=0}^{m-1} r^{m-1-k}\sum_{j=0}^{k-1}\eta\left\{\alpha X_{k,j}\left(\hat{A}_j - A\right)\hat{X}_{j,0} + \overline{\alpha}\left(X_{k,j}(\hat{A}_j - A)(\hat{X}_{j,0} + \hat{Y}_j)\right)\right\} c \\
&= \eta^2\sum_{k=0}^{m-1} r^{m-1-k}\sum_{j=0}^{k-1} X_{k,j}\left(\hat{A}_j - A\right)\left(\hat{X}_{j,0} + \overline{\alpha}\hat{Y}_j\right) c \\
&= \eta^2\sum_{k=0}^{m-1} Z_k c.
\end{aligned}$$

where

$$Z_k = r^{m-k-1} \sum_{j=0}^{k-1} X_{k,j}(\hat{A}_j - A)\left(\hat{X}_{j,0} + \overline{\alpha}\hat{Y}_j\right).$$

(Bounding $\mathbb{E}[\|v_m - \overline{v}_m\|^2]$ using the martingale sequence) Z_k is the martingale sequence with respect to its input $\mathcal{F}_k = (\hat{A}_0, \ldots, \hat{A}_{k-1})$ because

$$\begin{aligned}\mathbb{E}[Z_k \mid \mathcal{F}_{k-1}, \ldots \mathcal{F}_0] &= r^{m-k-1}\left\{\sum_{j=0}^{k-2} X_{k,j}(\hat{A}_j - A)\left(\hat{X}_{j,0} + \overline{\alpha}\hat{Y}_j\right) + X_{k,k-1}\mathbb{E}\left[\hat{A}_{k-1} - A\right]\left(\hat{X}_{k-1,0} + \overline{\alpha}\hat{Y}_{k-1}\right)\right\} \\ &= Z_{k-1},\end{aligned}$$

Using this fact, we have

$$\begin{aligned}\mathbb{E}\left[\left\|\sum_{k=0}^{m-1} Z_k\right\|^2\right] &= \sum_{k=0}^{m-1}\mathbb{E}\left[\|Z_k\|^2\right] + 2\sum_{k=0}^{m-1}\sum_{j=k+1}^{m-1}\mathbb{E}\left[Z_k^\top Z_j\right] \\ &= \sum_{k=0}^{m-1}\mathbb{E}\left[\|Z_k\|^2\right] + 2\sum_{k=0}^{m-1}\sum_{j=k+1}^{m-1}\mathbb{E}\left[Z_k^\top \underbrace{\mathbb{E}[Z_j \mid \mathcal{F}_k]}_{=Z_k}\right] \\ &= \sum_{k=0}^{m-1}\mathbb{E}\left[\|Z_k\|^2\right] + 2\sum_{k=0}^{m-1}(m-1-k)\mathbb{E}\left[\|Z_k\|^2\right] \\ &= \sum_{k=0}^{m-1}(2(m-k)-1)\mathbb{E}\left[\|Z_k\|^2\right]\end{aligned}$$

$$\mathbb{E}\left[\|v_m - \overline{v}_m\|^2\right] = \mathbb{E}\left[\left\|\eta^2\sum_{k=0}^{m-1} Z_k c\right\|^2\right] \leq \eta^4\|c\|^2\mathbb{E}\left[\left\|\sum_{k=0}^{m-1} Z_k\right\|^2\right]. \tag{37}$$

Hence,

$$\begin{aligned}\mathbb{E}\left[\|v_m - \overline{v}_m\|^2\right] &\leq \eta^4\|c\|^2\mathbb{E}\left[\left\|\sum_{k=0}^{m-1} Z_k\right\|^2\right] \\ &= \eta^4\|c\|^2\sum_{k=0}^{m-1}(2(m-k)-1)r^{2(m-k-1)}\mathbb{E}\left[\left\|\sum_{j=0}^{k-1} X_{k,j}(\hat{A}_j - A)\left(\hat{X}_{j,0} + \overline{\alpha}\hat{Y}_j\right)\right\|^2\right] \\ &= \eta^4\|c\|^2\sum_{k=0}^{m-1}(2(m-k)-1)r^{2(m-k-1)}\sum_{j=0}^{k-1}\|X_{k,j}\|^2\mathbb{E}\left[\|\hat{A}_j - A\|^2\|\hat{X}_{j,0} + \overline{\alpha}\hat{Y}_j\|^2\right],\end{aligned}$$

where the last equation uses the property of martingale *difference* sequence:

$$\mathbb{E}\left[X_{k,j}(\hat{A}_j - A)\left(\hat{X}_{j,0} + \overline{\alpha}\hat{Y}_j\right) \mid \mathcal{F}_{j-1}, \ldots \mathcal{F}_0\right] = 0.$$

(Handling the geometric sums) Since $\|\hat{A}_j - A\|^2 \leq \hat{\rho}^2$ for any j, we factor out $\hat{\rho}^2$, obtaining

$$\mathbb{E}\left[\|v_m - \overline{v}_m\|^2\right] \leq \eta^4\|c\|^2\hat{\rho}^2\sum_{k=0}^{m-1}(2(m-k)-1)r^{2(m-k-1)}\sum_{j=0}^{k-1}\|X_{k,j}\|^2\mathbb{E}\left[\|\hat{A}_j - A\|^2\|\hat{X}_{j,0} + \overline{\alpha}\hat{Y}_j\|^2\right]. \tag{38}$$

To simplify the notations for bound in the summation, let

$$\hat{R} = (1-\eta) + \eta\hat{\rho}, \quad R = (1-\eta) + \eta\rho, \quad r = (1-\eta) + \eta\alpha.$$

Then bound each of $\|X_{k,j}\|$, $\|\hat{X}_{j,0}\|$, and $\|\hat{Y}_j\|$ is obtained by geometric terms in R or $\hat{R}$:

$$\|X_{k,j}\| \le R^{k-j-1}, \quad \|\hat{X}_{j,0}\| \le R^j, \quad \|\hat{Y}_j\| \le \eta \sum_{s=0}^{j-1} \hat{R}^{j-s-1} = \frac{\eta}{1-\hat{R}}\left(1-\hat{R}^j\right).$$

It follows that

$$\begin{aligned}
\mathbb{E}\left[\|\hat{X}_{j,0} + \overline{\alpha}\hat{Y}_j\|^2\right] &= \left(\left(1-\overline{\alpha}\frac{\eta}{1-\hat{R}}\right)\hat{R}^j + \overline{\alpha}\frac{\eta}{1-\hat{R}}\right)^2 \\
&= \left(1-\overline{\alpha}\frac{\eta}{1-\hat{R}}\right)^2 \hat{R}^{2j} - 2\left(\overline{\alpha}\frac{\eta}{1-\hat{R}}\left(\overline{\alpha}\frac{\eta}{1-\hat{R}} - 1\right)\right)\hat{R}^j + \left(\overline{\alpha}\frac{\eta}{1-\hat{R}}\right)^2 \\
&\le \left(1-\overline{\alpha}\frac{\eta}{1-\hat{R}}\right)^2 \hat{R}^{2j} + \left(\overline{\alpha}\frac{\eta}{1-\hat{R}}\right)^2
\end{aligned}$$

where the last inequality uses $\overline{\alpha}\frac{\eta}{1-R} \ge 1$ given from $\alpha \le \rho < \hat{\rho}$ and

$$\overline{\alpha}\frac{\eta}{1-R} = (1-\alpha)\frac{1}{1-\hat{\rho}} \ge (1-\rho)\frac{1}{1-\rho} = 1$$

Then, by denoting

$$\begin{gathered}
R_p = R^2, \quad \hat{R}_p = \hat{R}^2, \quad r_p = r^2, \quad \omega = \frac{R_p}{r_p}, \\
\kappa_1 = \left(1-\overline{\alpha}\frac{\eta}{1-\hat{R}}\right)^2, \quad \kappa_2 = \left(\overline{\alpha}\frac{\eta}{1-\hat{R}}\right)^2, \quad \pi_1 = \frac{\hat{R}_p}{R_p}, \quad \pi_2 = R_p^{-1},
\end{gathered}$$

(38) is expressed as

$$\mathbb{E}\left[\|v_m - \overline{v}_m\|^2\right] \le \eta^4\|c\|^2\hat{\rho}^2\frac{1}{R_p}r_p^{m-1}\sum_{k=0}^{m-1}(2(m-k)-1)\omega^k\left[\kappa_1\sum_{j=0}^{k-1}\pi_1^j + \kappa_2\sum_{j=0}^{k-1}\pi_2^j\right]. \tag{39}$$

We then split this into a decaying part and a non-decaying part.

First, consider the first term of the decaying factor in (39). Recalling $\pi_1 > 1$ implied by $\mathbb{E}\left[\|\hat{A}_j - A\|^2\right] \ne 0$ and Lemma A.3, we have

$$\begin{aligned}
\kappa_1\frac{1}{R_p}r_p^{m-1}\sum_{k=0}^{m-1}(2(m-k)-1)\omega^k\sum_{j=0}^{k-1}\pi_1^j &= \kappa_1\frac{1}{R_p}\frac{1}{\pi_1-1}r_p^{m-1}\sum_{k=0}^{m-1}(2(m-k)-1)\omega^k\left(\pi_1^k-1\right) \\
&\le 2\kappa_1\frac{1}{R_p}\frac{1}{\pi_1-1}r_p^{m-1}\left(\frac{(\omega\pi_1)^{m+1}}{(\omega\pi_1-1)^2}\right) \\
&= 2\kappa_1\frac{1}{\frac{\hat{R}_p}{R_p}-1}\frac{1}{R_p}r_p^{m-1}\frac{\left(\frac{\hat{R}_p}{r_p}\right)^{m+1}}{\left(\frac{\hat{R}_p}{r_p}-1\right)^2} \\
&= 2\kappa_1\frac{1}{r_p^2\left(\hat{R}_p - R_p\right)}\frac{\hat{R}_p^{m+1} - r_p^{m+1}}{\left(\frac{\hat{R}_p}{r_p}-1\right)^2} = O\left(\hat{R}_p^m\right).
\end{aligned} \tag{40}$$

The last equation uses the fact $\alpha \le \rho$, which implies $r_p < \hat{R}_p$.

Now consider the second term of (39), which is the non-decaying part. Similarly, we use Lemma A.3 and $\pi_2 > 1$ from $\mathbb{E}\left[\|\hat{A}_j - A\|^2\right] \neq 0$, having

$$\begin{aligned}
\kappa_2 \frac{1}{R_p} r_p^{m-1} \sum_{k=0}^{m-1} (2(m-k)-1)\omega^k \sum_{j=0}^{k-1} \pi_2^j &= \kappa_2 \frac{1}{R_p} \frac{1}{\pi_2 - 1} r_p^{m-1} \sum_{k=0}^{m-1} (2(m-k)-1)\omega^k \left(\pi_2^k - 1\right) \\
&\leq \kappa_2 \frac{1}{R_p} \frac{1}{\pi_2 - 1} r_p^{m-1} \frac{(\omega\pi_2)^m (\omega\pi_2 + 1)}{(\omega\pi_2 - 1)^2} \\
&= \kappa_2 \frac{1}{1 - R_p} r_p^{m-1} \frac{\left(\frac{1}{r_p}\right)^m \left(1 + \frac{1}{r_p}\right)}{\left(\frac{1}{r_p} - 1\right)^2} \\
&= \kappa_2 \frac{1}{1 - R_p} \frac{1 + r_p}{(1 - r_p)^2}.
\end{aligned}$$

Since the denominator can be replaced with

$$\begin{aligned}
1 - R^2 = 1 - ((1-\eta) + \eta\rho)^2 &= \eta\left(2 - \eta\left(1 - \rho\right)\right)(1 - \rho), \\
(1 - \hat{R})^2 = (1 - ((1-\eta) + \eta\hat{\rho}))^2 &= \eta^2 (1 - \hat{\rho})^2, \\
\left(1 - r^2\right)^2 = \left(1 - (1 - \eta(1-\alpha))^2\right)\right)^2 &= \eta^2 (2 - \eta(1-\alpha))^2 (1 - \alpha)^2,
\end{aligned}$$

the non-decaying part is finally expressed as

$$\kappa_2 \frac{1}{R_p} r_p^{m-1} \sum_{k=0}^{m-1} (2(m-k)-1)\omega^k \sum_{j=0}^{k-1} \pi_2^j \leq \frac{1 + (1 - \eta(1-\alpha))^2}{\eta^3 (1-\hat{\rho})^2 \left(2 - \eta(1-\rho)\right)(1-\rho)\left(2 - \eta(1-\alpha)\right)^2}. \tag{41}$$

(Converting each factor into the final form) By injecting the decaying term (40) and non-decaying term (41) into (39), we present the final result as:

$$\mathbb{E}\left[\|v_m - \overline{v}_m\|^2\right] \leq 2L_f^2 \hat{\rho}^2 \frac{\eta(1 + (1 - \eta(1-\alpha))^2)}{(1-\hat{\rho})^2 \left(2 - \eta(1-\rho)\right)\left(2 - \eta(1-\alpha)\right)^2} + O\left((1 - \eta(1-\hat{\rho}))^{2m}\right).$$

□

A.6 Convergence Analysis of Deterministic MixedFP-KM

Lemma A.5. *Suppose $0 < \eta < \frac{1}{1-\hat{\rho}}$, $\alpha \leq \rho$, and Assumption 3 hold, then for any $m \geq 0$,*

$$\|\overline{v}_m - \nabla_x f\|^2 = O\left((1 - \eta(1-\rho))^{2m}\right).$$

Proof. From Lemma A.1, we have

$$\begin{aligned}
\overline{v}_m &= \eta \sum_{k=0}^{m-1} r^{m-k-1} \left(\alpha \overline{u}_k + \overline{\alpha w}_k\right) \\
&= \eta \sum_{k=0}^{m-1} r^{m-k-1} \left(\alpha P^k c + \overline{\alpha}\left(P^k c + \eta \sum_{s=0}^{k-1} P^{k-s-1} c\right)\right) \\
&= \eta \sum_{k=0}^{m-1} r^{m-k-1} \left(P^k c + \overline{\alpha}\eta \sum_{s=0}^{k-1} P^{k-s-1} c\right).
\end{aligned} \tag{42}$$

Using the identity for the matrix inverse

$$\sum_{i=0}^{n-1} X^i = \left(I - X^n\right)\left(I - X\right)^{-1},$$

we see that

$$\begin{aligned}\eta(I-P)^{-1}c &= \eta\left(I-(I-\eta(I-A))\right)^{-1} \\ &= \eta\left(\eta(I-A)\right)^{-1}c = \nabla_x f.\end{aligned}$$

Therefore, the term inside the sum in (42) is

$$\begin{aligned}P^k c + \overline{\alpha}\eta\sum_{s=0}^{k-1} P^{k-s-1}c &= P^k c + \overline{\alpha}\eta\left((I-P^k)(I-P)^{-1}\right)c \\ &= P^k\left(I-\overline{\alpha}(I-A)^{-1}\right)c + \overline{\alpha}\nabla_x f.\end{aligned}$$

By letting $D = I - \overline{\alpha}(I-A)^{-1}$ and substituting them into (42), we obtain

$$\begin{aligned}\overline{v}_m - \nabla_x f &= \eta\sum_{k=0}^{m-1} r^{m-k-1}\left(P^k Dc + \overline{\alpha}\nabla_x f\right) - \nabla_x f \\ &= \eta\sum_{k=0}^{m-1} r^{m-k-1}P^k Dc + \underbrace{\frac{\overline{\alpha}\eta}{1-r}}_{=1}\left(1-r^m\right)\nabla_x f - \nabla_x f \\ &= \eta\sum_{k=0}^{m-1} r^{m-k-1}P^k D - r^m\nabla_x f.\end{aligned}$$

Using $\|a+b\|^2 \le 2\|a\|^2 + 2\|b\|^2$ and $\alpha \le \rho$, we have

$$\begin{aligned}\|\overline{v}_m - \nabla_x f\|^2 &\le 2\eta^2\|D\|^2\left(\sum_{k=0}^{m-1} r^{m-k-1}R^k\right)^2\|c\|^2 + 2\left(\frac{r^m}{1-\rho}\right)^2\|c\|^2 \\ &= 2\eta^2\|D\|^2\left(\frac{R^m-r^m}{R-r}\right)^2\|c\|^2 + 2\left(\frac{r^m}{1-\rho}\right)^2\|c\|^2 \\ &= O(R^{2m}) = O\left((1-\eta(1-\rho))^{2m}\right).\end{aligned}$$

□

A.7 Main Convergence Theorems

Theorem 4.5 (MixedFP-KM)**.** *Let $\alpha \in [0,1)$. When $\eta_m = \eta < \frac{1}{1-\hat{\rho}}$ and Assumptions 1 to 3 hold, then for any $m \ge 0$,*

$$\mathbb{E}\left[\|v_m - \nabla_x f\|^2\right] \le \frac{1+\alpha_\eta}{\left(1+\sqrt{\alpha_\eta}\right)^2}\sigma_\eta + O\left(\max\{\rho_\eta, \alpha_\eta\}^m\right),$$

where

$$\sigma_\eta = \frac{\eta L_f^2\hat{\rho}^2}{(1-\hat{\rho})^2(2-\eta(1-\rho))(1-\rho)}, \quad \rho_\eta = (1-\eta+\eta\hat{\rho})^2, \quad \alpha_\eta = (1-\eta+\eta\alpha)^2.$$

Proof. We divide the difference into

$$v_m - \nabla_x f = (v_m - \overline{v}_m) + (\overline{v}_m - \nabla_x f),$$

where $\overline{v}_m$ is the deterministic counterpart. Then

$$\mathbb{E}\left[\|v_m - \nabla_x f\|^2\right] \le \mathbb{E}\left[\|v_m - \overline{v}_m\|^2\right] + \mathbb{E}\left[\|\overline{v}_m - \nabla_x f\|^2\right],$$

where the cross term vanishes because $\mathbb{E}[v_m - \overline{v}_m] = 0$. Applying Lemma A.4 (for $\|v_m - \overline{v}_m\|$) and Lemma A.5 (for $\|\overline{v}_m - \nabla_x f\|$) proves the claim.

□

We obtain Theorem 4.1 as the special case of Theorem 4.5 where $\eta = 1$ with additional consideration for $\alpha = 1$.

Theorem 4.1 (MixedFP)**.** *Suppose Assumptions 1 to 3 hold and $\alpha \in [0, 1]$, then for any $m \geq 0$,*

$$\mathbb{E}\left[\|\hat{v}_m - \nabla_x f\|^2\right] \leq \begin{cases} \sigma_1 + O\left(\rho_1^m\right) & \text{if } \alpha \in \{0, 1\}, \\ \dfrac{1+\alpha_1}{(1+\sqrt{\alpha_1})^2}\sigma_1 + O\left(\max\{\rho_1, \alpha_1\}^m\right) & \text{otherwise,} \end{cases}$$

where

$$\sigma_1 = \frac{L_f^2 \hat{\rho}^2}{(1-\hat{\rho})^2(1-\rho^2)}, \quad \rho_1 = \hat{\rho}^2, \quad \alpha_1 = \alpha^2.$$

Proof. For $\alpha \in [0, 1)$, this result is the special case of Theorem 4.5 where $\eta = 1$, which gives $\alpha_\eta = \alpha^2 = \alpha_1$, $\rho_\eta = \hat{\rho}^2 = \rho_1$, and $\sigma_\eta = \frac{L_f^2\hat{\rho}^2}{(1-\hat{\rho})^2(1-\rho^2)} = \sigma_1$. When $\alpha = 1$, $\hat{v}_m$ is equivalent to $\hat{u}_{m-1}$ whose error is given by Corollary B.1. □

B Theoretical Analysis of StocFP with the Stochastic KM Iteration

B.1 Convergence of StocFP with the Stochastic KM Iteration

Theorem 4.6 (Stochastic KM iteration of StocFP)**.** *Suppose $\eta_m = \eta < \frac{1}{1-\hat{\rho}}$ and Assumptions 1 to 3 hold. Then for any $m \geq 0$,*

$$\mathbb{E}\left[\|w_m - \nabla_x f\|^2\right] \leq \sigma_\eta + O\left(\rho_\eta^m\right),$$

where σ_η and ρ_η are as defined in Theorem 4.5.

Proof. The proof is divided into two parts: the variance bound (that is, the stochastic error between w_m and $\overline{w}_m$), and the deterministic error bound (i.e., the bias $\|\overline{w}_m - \nabla_x f\|^2$). We then combine these two parts.

From (29), the error in the w–sequence can be written as

$$w_m - \overline{w}_m = \eta \sum_{j=0}^{m-1} P^{m-1-j}(\hat{A}_j - A)\left\{\left(\prod_{i=0}^{j-1}\hat{P}_i\right)c + \eta\sum_{k=0}^{j-1}\left(\prod_{i=k+1}^{j-1}\hat{P}_i\right)c\right\}.$$

Recall

$$X_{m,j} := P^{m-1-j}, \quad \hat{X}_{j,0} := \prod_{i=0}^{j-1}\hat{P}_i, \quad \hat{Y}_j := \eta\sum_{k=0}^{j-1}\prod_{i=k+1}^{j-1}\hat{P}_i.$$

Then,

$$w_m - \overline{w}_m = \eta\sum_{j=0}^{m-1} X_{m,j}(\hat{A}_j - A)\left(\hat{X}_{j,0} + \hat{Y}_j\right)c.$$

Because $\mathbb{E}[\hat{A}_j] = A$ from Assumption 3, the sequence

$$W_j := X_{m,j}(\hat{A}_j - A)\left(\hat{X}_{j,0} + \hat{Y}_j\right)$$

forms a martingale–difference sequence with respect to the natural filtration $\mathcal{F}_j = \sigma(\hat{A}_0, \ldots, \hat{A}_j)$. Hence, by orthogonality of martingale differences and the Cauchy–Schwarz inequality,

$$\begin{aligned}\mathbb{E}\left[\|w_m - \overline{w}_m\|^2\right] &= \mathbb{E}\left[\left\|\eta \sum_{j=0}^{m-1} W_j c\right\|^2\right] \\ &\leq \eta^2 \|c\|^2 \mathbb{E}\left[\left\|\sum_{j=0}^{m-1} W_j\right\|^2\right] \\ &= \eta^2 \|c\|^2 \sum_{j=0}^{m-1} \mathbb{E}\left[\|W_j\|^2\right].\end{aligned}$$

By definition,

$$\|W_j\| \leq \|X_{m,j}\| \cdot \|\hat{A}_j - A\| \cdot \|\hat{X}_{j,0} + \hat{Y}_j\|.$$

Under Assumption 3(ii) we have $\|\hat{A}_j - A\| \leq \hat{\rho}$. Moreover, since the norm of $P = I - \eta(I - A)$ is bounded by $\|P\| \leq R$ with $R = (1-\eta) + \eta\rho = 1 - \eta(1-\rho)$. Thus,

$$\|X_{m,j}\| \leq R^{m-1-j}$$

Similarly, the product of the stochastic matrices satisfies

$$\|\hat{X}_{j,0}\| \leq \hat{R}^j, \quad \|\hat{Y}_j\| \leq \eta \sum_{k=0}^{j-1} \hat{R}^{j-k-1} = \frac{\eta}{1-\hat{R}}\left(1 - \hat{R}^j\right)$$

with $\hat{R} = (1-\eta) + \eta\hat{\rho} = 1 - \eta(1-\hat{\rho})$. Therefore,

$$\begin{aligned}\|\hat{X}_{j,0} + \hat{Y}_j\|^2 &\leq \left(\hat{R}^j + \frac{\eta}{1-\hat{R}}\left(1 - \hat{R}^j\right)\right)^2. \\ &\leq \left(1 - \frac{\eta}{1-\hat{R}}\right)^2 \hat{R}^{2j} + \left(\frac{\eta}{1-\hat{R}}\right)^2\end{aligned}$$

Substituting them into (37) gives

$$\mathbb{E}\left[\|w_m - \overline{w}_m\|^2\right] \leq \eta^2 \|c\|^2 \hat{\rho}^2 \sum_{j=0}^{m-1} R^{2(m-1-j)} \left(\left(1 - \frac{\eta}{1-\hat{R}}\right)^2 \hat{R}^{2j} + \left(\frac{\eta}{1-\hat{R}}\right)^2\right). \tag{43}$$

By letting

$$\kappa_3 = \left(1 - \frac{\eta}{1-R}\right)^2, \quad \kappa_4 = \left(\frac{\eta}{1-R}\right)^2,$$

We simplify the notation of as (43) as

$$\mathbb{E}\left[\|w_m - \overline{w}_m\|^2\right] \leq \eta^2 \|c\|^2 \hat{\rho}^2 \overline{R}_p^{m-1} \sum_{k=0}^{m-1} \left(\kappa_3 \pi_1^k + \kappa_4 \pi_2^k\right).$$

Since $R = 1 - \eta(1-\rho) < \hat{R} = 1 - \eta(1-\hat{\rho}) < 1$ from $\hat{\rho} > \rho$ and $0 < \eta < \frac{1}{1-\hat{\rho}}$, we have

$$\begin{aligned}\overline{R}_p^{m-1} \sum_{k=0}^{m-1} \left(\kappa_3 \pi_1^k + \kappa_4 \pi_2^k\right) &= \overline{R}_p^{m-1} \left(\kappa_3 \frac{\pi_1^k - 1}{\pi_1 - 1} + \kappa_4 \frac{\pi_2^k - 1}{\pi_2 - 1}\right) \\ &\leq \left(\kappa_3 \frac{R_p^m}{R_p - \overline{R}_p} + \kappa_4 \frac{1}{1 - \overline{R}_p}\right) \\ &= \kappa_4 \frac{1}{1 - \overline{R}_p} + O\left(R_p^m\right) \\ &= \frac{1}{\eta(1-\hat{\rho})^2(2 - \eta(1-\rho))(1-\rho)} + O\left(R_p^m\right)\end{aligned}$$

It follows that

$$\mathbb{E}\left[\|w_m - \overline{w}_m\|^2\right] \leq L_f^2 \hat{\rho}^2 \frac{\eta}{(1-\hat{\rho})^2\,(2-\eta(1-\rho))\,(1-\rho)} + O\left((1-\eta(1-\hat{\rho}))^{2m}\right).$$

We now derive in full the bound for the deterministic error $\|\overline{w}_m - \nabla_x f\|^2$. The deterministic iteration is given by

$$\overline{w}_0 = c, \quad \overline{w}_{m+1} = P\overline{w}_m + \eta c, \quad \text{with} \quad P = I - \eta(I - A).$$

Unrolling this recursion yields

$$\overline{w}_m = P^m c + \eta \sum_{k=0}^{m-1} P^{m-1-k} c.$$

Changing the index (letting $j = m - 1 - k$) we have

$$\overline{w}_m = P^m c + \eta \sum_{j=0}^{m-1} P^j = P^m c + \eta\,(I - P^m)\,(I - P)^{-1} c.$$

Subtracting $\nabla_x f$ from $\overline{w}_m$ yields

$$\begin{aligned}
\overline{w}_m - \nabla_x f &= \left[P^m c + \eta\,(I - P^m)\,(I - P)^{-1} c\right] - \eta(I - P)^{-1} c \\
&= P^m c + \eta\left[(I - P^m) - I\right](I - P)^{-1} c \\
&= P^m c - \eta P^m (I - P)^{-1} c \\
&= P^m \left[c - \eta(I - P)^{-1} c\right].
\end{aligned}$$

Since $\nabla_x f = \eta(I - P)^{-1} c$, we rewrite this as

$$\overline{w}_m - \nabla_x f = P^m\,(c - \nabla_x f).$$

As $\|P\| \leq 1 - \eta(1-\rho) =: R$ holds under Assumption 3, we deduce that

$$\|\overline{w}_m - \nabla_x f\|^2 = O\left(\left(1 - \eta(1-\rho)\right)^{2m}\right).$$

We now decompose the overall error as

$$w_m - \nabla_x f = (w_m - \overline{w}_m) + (\overline{w}_m - \nabla_x f).$$

Since $\mathbb{E}\left[w_m - \overline{w}_m\right] = 0$, we have

$$\mathbb{E}\left[\|w_m - \nabla_x f\|^2\right] \leq \mathbb{E}\left[\|w_m - \overline{w}_m\|^2\right] + \|\overline{w}_m - \nabla_x f\|^2.$$

Inserting the bounds derived above yields

$$\mathbb{E}\left[\|w_m - \nabla_x f\|^2\right] \leq L_f^2 \hat{\rho}^2 \frac{\eta(1 + (1-\eta(1-\alpha))^2)}{(1-\hat{\rho})^2\,(2-\eta(1-\rho))\,(1-\rho)} + O\left((1-\eta(1-\hat{\rho}))^{2m}\right).$$

□

We obtain the error bounds for StocFP (4) StocRB (5) as the special case of Theorem 4.6 where $\eta = 1$.

Corollary B.1 (StocFP and StocRB)**.** *Suppose Assumptions 1 to 3 hold. Then for any $m \geq 0$,*

$$\mathbb{E}\left[\|\hat{w}_m - \nabla_x f\|^2\right] = \mathbb{E}\left[\|\hat{y}_m - \nabla_x f\|^2\right] \leq \sigma_1 + O\left(\rho_1^m\right),$$

where σ_1 and ρ_1 are as defined in Theorem 4.1.

B.2 Comparison with the Existing Result

Theorem 4.6 for StocFP with the stochastic KM iteration (StocFP-KM) slightly differs from a bound derived from results in the original paper (Grazzi et al., 2021).

Grazzi et al. (2021, Theorem 5.1) presents a non-asymptotic error bound for StocFP-KM with a decreasing stepsize, while the bound for a fixed stepsize is partially provided in their Theorem 4.1, which is a non-asymptotic version of our Theorem 4.2. By combining their Theorem 4.1 and their proof of Theorem 5.1, we can derive the following result for StocFP-KM with a constant stepsize.

Theorem B.2 (Fixed stepsize and asymptotic version of Grazzi et al. (2021, Theorem 5.1))**.** *Let* $\sigma_2 = \frac{2\left(\hat{\rho}^2+\rho^2\right)}{(1-\rho)^2}$ *and suppose* $0 < \eta \leq \frac{1}{1+\sigma_2}$ *and Assumptions 1 to 3 hold. Then for any* $m \geq 0$*,*

$$\mathbb{E}\left[\|w_m - \nabla_x f\|^2\right] \leq 2L_f^2\hat{\rho}^2\frac{\eta}{(1-\rho^2)(1-\rho)^2} + O\left((1-\eta(1-\rho^2))^m\right)$$

Although both Theorem 4.6 and Theorem B.2 evaluate the same method, the derivations differ. We chose to evaluate the error of the existing method using an approach similar to our own analysis in Theorem 4.5 in order to ensure a fair comparison between our proposed method and that of Grazzi et al. (2021). Consequently, our Theorem 4.6 does not unfairly penalize the method in Grazzi et al. (2021); in fact, there are cases where Theorem 4.6 even yields tighter bounds than Theorem B.2. One example is obtained by choosing

$$\rho = 0.3, \quad \hat{\rho} = 0.4, \quad \eta = 0.5.$$

These values satisfy the side conditions $0 < \rho < 1$, $\rho < \hat{\rho}$, and $\eta < \frac{1}{1-\hat{\rho}}$. By dividing both errors by $\eta L_f^2\hat{\rho}^2$, we have

$$\underbrace{\frac{1}{(1-\hat{\rho})^2(2-\eta(1-\rho))(1-\rho)}}_{\approx 2.404} + O\left(\left(\underbrace{(1-\eta(1-\hat{\rho}))^2}_{=0.490}\right)^m\right) < \underbrace{\frac{2}{(1-\rho^2)(1-\rho)^2}}_{\approx 4.484} + O\left(\left(\underbrace{1-\eta(1-\rho^2)}_{=0.545}\right)^m\right)$$

This is an example where both the non-decaying term and the decaying term of our Theorem 4.6 are superior to Theorem B.2 obtained from the existing results.

C Experiment Setup Details

In this section, we provide detailed information about the setup used in our empirical evaluations.

C.1 Effect of Mixing Rate

The experiment in Section 5.1 evaluates the estimation error of hypergradients in the synthetic setting where the Jacobian matrix is a random variable. We implemented the following configuration:

The experiment used the following parameter settings: dimension of inner-parameter $d_x = 10$; dimension of outer-parameter $d_\lambda = 10$; number of parent matrices $H_1, \ldots, H_n$ for sampling $n = 1000$; eigenvalue bound parameter $\epsilon = 10^{-2}$; maximum iterations $M = 1000$; scale parameter values $\gamma \in \{10^{-3}, 10^{-2}, 10^{-1}, 1.0\}$; and mixing rate values $1-\alpha \in \{0, 0.001, 0.01, 0.1, 1.0\}$, where $\alpha = 0$ corresponds to StocFP and $\alpha = 1.0$ corresponds to StocRB.

Any matrix in $H_1, \ldots, H_n$ was constructed to ensure that each eigenvalue follows a uniform distribution in $[0, 1-\epsilon]$. The matrix $H_i \in \{H_1, \ldots, H_n\}$ was constructed through the following process: first, generating a random matrix $R \in \mathbb{R}^{d_x \times d_x}$ with entries sampled from a standard normal distribution; then creating a symmetric matrix $S = (R + R^\top)/2$; next, computing the QR decomposition of S to obtain orthogonal eigenvectors Q; followed by sampling eigenvalues $\{\lambda_1, \ldots, \lambda_{d_x}\}$ uniformly from $[0, 1-\epsilon]$; and finally constructing $H_i = Q \cdot \mathrm{diag}(\lambda_1, \ldots, \lambda_{d_x}) \cdot Q^\top$.

The coefficients $c \in \mathbb{R}^{d_x}$, $d \in \mathbb{R}^{d_\lambda}$, and $B \in \mathbb{R}^{d_x \times d_\lambda}$ were sampled independently from a uniform distribution in $[0, 1]$ for each element.

Task	Dataset	n_{train}	n_{val}	d_x	d_λ	γ
Hyperparameter Optimization	Adult Income	5000	5000	14	14	10^{-1}
Influence Estimation	Fashion MNIST	5000	5000	784	5000	10^{-2}
Meta Learning	California Housing	5000	5000	8	8	10^{-1}

Table 2: Experiment settings for the real-world tasks.

C.2 Compare with Existing Approaches

In this section, we provide detailed information about the setup used in Section 5.2.

The synthetic problem setting differs from that in Section 5.1 in the following aspects: we used fixed $\gamma = 1.0$ instead of varying it over $[0, 1]$, the dimensions of inner-parameter x and outer-parameter λ were both set to 100, we ran experiments with 10 different random seeds (instead of 100), and we used 10,000 iterations for each method. For the hyperparameter optimization task, we used binary classification with the Adult Income dataset. For the influence estimation task, we standardized the Fashion MNIST data with zero mean and unit variance. For the meta learning task, we used the min-max scaling on the California Housing dataset with the regularization parameter $\beta = 0.1$. Table 2 shows the dataset sizes and the values of γ for $\varphi(x, \lambda) = (I - \gamma\hat{H})x + B\lambda$ used in the three real-world tasks.

For the initialization of inner-parameter x, we used samples from the normal distributions scaled by a constant factor for both logistic regression models used in the hyperparameter optimization and influence estimation tasks and linear regression models for the meta learning. For the outer-parameter λ, the initialization varies by task. In the synthetic task, λ is initialized with values sampled uniformly from [0,1]. In the Hyperparameter optimization task, λ serves as a regularization coefficient initialized with a small constant value. In the influence estimation task, λ represents loss weights for each training sample, which is a ones-vector. In the meta learning task, λ functions as a biased regularization term set to the initial value of the model's parameters x.

Any inner-problem optimization was performed using the Adam optimizer with a learning rate of 0.01. To rule out the effect incurred by inexact $x(\lambda)$, for any task, we used the full-batch inner loss to compute gradients for Adam and ran 1,000 epochs to ensure the convergence.

We optimized the hyperparameters using grid search to find the configuration that minimizes the average error over the last $M/10$ iterations. For each method, we performed a grid search over the following hyperparameter ranges:

- Mixing rate $1 - \alpha \in \{0, 10^{-4}, 2 \times 10^{-4}, 5 \times 10^{-4}, 10^{-3}, 2 \times 10^{-3}, 5 \times 10^{-3}, 10^{-2}, 2 \times 10^{-2}, 5 \times 10^{-2}, 10^{-1}, 2 \times 10^{-1}, 5 \times 10^{-1}\}$
- Stepsize $\eta \in \{10^{-3}, 2 \times 10^{-3}, 5 \times 10^{-3}, 10^{-2}, 2 \times 10^{-2}, 5 \times 10^{-2}, 10^{-1}, 2 \times 10^{-1}, 5 \times 10^{-1}, 1.0\}$
- For decreasing stepsize schedule: $\beta \in \{10, 20, 50, 100, 200, 500, 1000, 2000, 5000, 10000\}$ and $\delta \in \{10, 20, 50, 100, 200, 500, 1000, 2000, 5000, 10000\}$ with the constraint $\beta \leq \delta$