Conditional Gradient Methods for Stochastically Constrained Convex Minimization

Maria-Luiza Vladarean [1] **Ahmet Alacaoglu** [1] **Ya-Ping Hsieh** [1] **Volkan Cevher** [1]

Abstract

We propose two novel conditional gradient-based methods for solving structured stochastic convex optimization problems with a large number of linear constraints. Instances of this template naturally arise from SDP-relaxations of combinatorial problems, which involve a number of constraints that is polynomial in the problem dimension. The most important feature of our framework is that only a subset of the constraints is processed at each iteration, thus gaining a computational advantage over prior works that require full passes. Our algorithms rely on variance reduction and smoothing used in conjunction with conditional gradient steps, and are accompanied by rigorous convergence guarantees. Preliminary numerical experiments are provided for illustrating the practical performance of the methods.

1. Introduction

We study the following optimization template:

$$\begin{aligned} &\min_{x\in\mathcal{X}} \; f(x) := \mathbb{E}\left[f(x,\xi)\right] \\ &A(\xi)x \in b(\xi) \text{ almost surely,} \end{aligned} \tag{1}$$

where $f(x,\xi) : \mathbb{R}^d \to \mathbb{R}$ are random convex functions with L_f-Lipschitz gradient, $\mathcal{X}$ is a convex and compact set of $\mathbb{R}^d$, $A(\xi)$ is an $m \times d$ matrix-valued random variable, and $b(\xi)$ is a closed and projectable random convex set in $\mathbb{R}^m$.

Stochastically constrained convex optimization problems have recently gained interest in the machine learning community, as they provide a convenient and powerful framework for handling instances subject to a large, or even infinite number of constraints. For example, convex feasibility and optimal control problems have variables lying in a possibly infinite intersection of stochastic, projectable constraint sets, and hence are tackled through this lens by Patrascu & Necoara (2017). Xu (2018) also studies the minimization of a stochastic objective controlled by a very large number of stochastic functional constraints, with application to stochastic linear programming. Finally, put forth by Fercoq et al. (2019), extensions to situations where the number of constraints is unknown (e.g. online settings) can be modeled by a template highly similar to (1), thus addressing important applications such as online portfolio optimization.

In this paper, we are interested in a class of applications which can benefit from being cast under template (1), namely semidefinite programs (SDPs) with a large number of linear constraints, such as arise in combinatorial optimization. A prominent example in machine learning is the k-means clustering problem, whose SDP relaxation comprises $\mathcal{O}\left(d^2\right)$ linear constraints where d is the number of data samples (Peng & Wei, 2007). Maximum a posteriori estimation (Huang et al., 2014), quadratic assignment (Burer & Monteiro, 2005), k-nearest neighbor classification (Weinberger & Saul, 2009) and Sparsest cut (Arora et al., 2009) are other relevant SDP instances with linear constraints of order $\mathcal{O}\left(d^2\right)$ or $\mathcal{O}\left(d^3\right)$. Coupled with large input dimensions, such SDPs become problematic for most existing methods, due to the high cost of processing the constraints in-full during optimization.

In contrast, casting such SDPs into (1) suggests a simple solution: treat the linear constraints stochastically by only accessing a random subset at each iteration, then solve (1) using cheap gradient methods. However, the bottleneck in executing this idea is that existing methods require the constraint $\mathcal{X}$ to posses an efficient projection oracle, whereas projecting onto the semidefinite cone amounts to full singular value decompositions, an operation that is prohibitively expensive even when the problem dimension is moderate. We hence ask:

> *Does a scalable method exist for solving* (1) *when the set* $\mathcal{X}$ *does not have an efficient projection oracle?*

The present work resolves the above challenge in the pos-

[1]École Polytechnique Fédérale de Lausanne, Switzerland. Correspondence to: Maria-Luiza Vladarean <maria-luiza.vladarean@epfl.ch>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020.

itive. To this end, we borrow tools from the conditional gradient methods (CGM) (Frank & Wolfe, 1956; Jaggi, 2013), which rely on the generally cheaper *linear minimization oracles* (lmo), rather than their projection counterparts. In particular, as the Lanczos method enables an efficient lmo computation for the spectrahedron (Arora et al., 2005), CGMs have already been proposed for solving SDPs (Jaggi, 2013; Garber & Hazan, 2016; Yurtsever et al., 2018; Locatello et al., 2019). However, none of these methods can handle the constraints stochastically.

In a nutshell, our approach relies on *homotopy smoothing* of the stochastic constraints in conjunction with CGM steps and a carefully chosen variance reduction procedure. Our analysis gives rise to two fully stochastic algorithms for solving problem (1) without projections onto $\mathcal{X}$. The first of the methods, H-SFW1, relies on a single sample (or fixed batch size) for computing the variance-reduced gradient and converges at a cost of $\mathcal{O}(\epsilon^{-6})$ lmo calls and $\mathcal{O}(\epsilon^{-6})$ stochastic first-order oracle (sfo) calls. The second, H-SPIDERFW, uses batches of increasing size under the SPIDER variance reduction scheme (Fang et al., 2018) and attains a theoretical complexity of $\mathcal{O}(\epsilon^{-2})$ lmo calls and $\mathcal{O}(\epsilon^{-4})$ sfo calls. The difference in convergence rates emphasizes the trade-off between between the computational cost per-iteration and the number of iterations required to reach the constrained optimum.

2. Related Work

The present work lies at the intersection of several lines of research, whose relevant literature we describe in the following sections.

Proximal Methods for Almost Sure Constraints. Problems of similar formulation to (1) have been addressed in prior literature under the assumption of an efficient projection oracle over $\mathcal{X}$. Works such as (Patrascu & Necoara, 2017; Xu, 2018; Fercoq et al., 2019) solve these problems via stochastic proximal methods and attain a complexity of $\mathcal{O}(\epsilon^{-2})$ sfo calls, which is known to be optimal even for unconstrained stochastic optimization. In particular, Patrascu & Necoara (2017) study convex constrained optimization, where the constraints are expressed as a (possibly infinite) intersection of stochastic, closed, convex and projectable sets X_ξ. Problem (1) can be partly cast to this template, with $A(\xi)X \in b(\xi)$ being the homologue of X_ξ. However, our additional set $\mathcal{X}$ does not allow for efficient projections, making this framework inapplicable.

Xu (2018) solves a convex constrained optimization problem over a convex set $\mathcal{X}$, subject to a large number of convex functional constraints f_j, $j = 1 \dots M$. The functions f_j are sampled uniformly at random during optimization, which corresponds to a finitely sampled instance of problem (1) for affine f_j. However, we meet again with the limiting condition that projections onto $\mathcal{X}$ are computationally expensive in our setting.

Finally, Fercoq et al. (2019) study convex problems subject to a possibly infinite number of almost sure linear inclusion constraints, a template which closely resembles ours. The limitation, however, lies in their inclusion of a proximal-friendly component in the objective used to perform stochastic proximal gradient steps. This assumption does not hold for our problem formulation.

Conditional Gradient Methods for Constrained Optimization. CGM was first proposed in the seminal work of Frank & Wolfe (1956) and its academic interest has witnessed a resurgence in the past decade. The advantage of CGMs lies in the low per-iteration cost of the lmo, alongside their ability to produce sparse solutions. In comparison to projection-based approaches, the lmo is cheaper to compute for several important domains, amongst which the spectrahedron, polytopes emerging from combinatorial optimization, and ℓ_p norm-induced balls (Garber, 2016). Consequently, CG-type methods have been studied under varying assumptions in (Hazan, 2008; Clarkson, 2010; Hazan & Kale, 2012; Jaggi, 2013; Lan, 2013; Balasubramanian & Ghadimi, 2018), and have been incorporated as cheaper subsolvers into algorithms which originally relied on projection oracles (Lan & Zhou, 2016; Liu et al., 2019).

CGMs have been further extended to the setting of convex composite minimization via the Augmented Lagrangian framework in (Gidel et al., 2018; Silveti-Falls et al., 2019; Yurtsever et al., 2019a). Most relevant to our work, CGM-based quadratic penalty methods have been studied for convex problems with constraints of the form $Ax - b \in \mathcal{K}$, where $\mathcal{K}$ is a closed, convex set (Yurtsever et al., 2018; Locatello et al., 2019). We compare our methods against the latter two in Section 4.5.

Variance Reduction. Stochastic variance reduction (VR) methods have gained popularity in recent years following their initial study by (Roux et al., 2012; Johnson & Zhang, 2013; Mahdavi et al., 2013). The VR technique relies on averaging schemes to reduce the variance inherent to stochastic gradients, with several different flavors having emerged in the past decade: SAG (Schmidt et al., 2017), SVRG (Johnson & Zhang, 2013), SAGA (Defazio et al., 2014), SVRRG++ (Allen-Zhu & Yuan, 2016), SARAH (Nguyen et al., 2017) and SPIDER (Fang et al., 2018). Such methods outperform the classical SGD under the finite sum model, a fact which led to their widespread use in large-scale applications and their further inclusion into other stochastic optimization algorithms (see for example (Xiao & Zhang, 2014; Hazan & Luo, 2016)).

Relevant to our setting, VR has been studied in the context of CGMs for convex minimization by (Mokhtari et al., 2018; Hazan & Luo, 2016; Locatello et al., 2019; Yurtsever et al., 2019b; Zhang et al., 2019). The sfo complexity of these methods varies depending on the VR scheme, with the best guarantee being of order $\mathcal{O}(\epsilon^{-2})$ (Zhang et al., 2019; Yurtsever et al., 2019b). For a thorough comparison of the complexities, we refer the reader to Section 6 of (Yurtsever et al., 2019b).

3. Preliminaries

Notation. We use $\|\cdot\|$ to express the Euclidean norm and $\langle\cdot,\ \cdot\rangle$ to denote the corresponding inner product. The distance between a point x and a set $\mathcal{X}$ is defined as $\operatorname{dist}(x,\mathcal{X}) := \inf_{y\in\mathcal{X}}\|y-x\|$. The indicator function of a set $\mathcal{X}$ is given by $\delta_{\mathcal{X}}(x) = 0,$ if $x\in\mathcal{X}$, and $\delta_{\mathcal{X}}(x) = +\infty$ otherwise. We denote by $\mathcal{D}_{\mathcal{X}} := \max_{(x,y)\in\mathcal{X}\times\mathcal{X}}\|x-y\|$ the diameter of a compact set $\mathcal{X}$.

For the probabilistic setting, we denote by ξ an element of our sample space and by $P(\xi)$ its probability measure. Unless stated otherwise, expectations will be taken with respect to ξ. We use $[n]$ to denote $\{1,2,\dots n\}$.

Given a function $f:\mathbb{R}^d\to\mathbb{R}$ and $L>0$, we say that f is L-smooth if ∇f is Lipschitz continuous, which is defined as $\|\nabla f(x)-\nabla f(y)\|\le L\|x-y\|, \forall x,y\in\mathbb{R}^d$.

Following the same setup as in (Fercoq et al., 2019), the space of random variables used in this work is

$$\mathcal{H} = \left\{y(\xi)_\xi\in\mathbb{R}^m \mid \xi\in\mathbb{R}^n,\ \mathbb{E}\left[\|y(\xi)_\xi\|^2\right] < +\infty\right\},$$

where the associated scalar product is given by $\langle x,\ z\rangle := \mathbb{E}\left[x(\xi)^T z(\xi)\right] = \int x(\xi)^T z(\xi) dP(\xi)$.

Smoothing. Nesterov (2005) proposes a technique for obtaining smooth approximations parametrized by β, of a non-smooth and convex function g. The resulting smoothed approximations take the following form:

$$g_\beta(x) = \max_y \langle y,\ x\rangle - g^*(y) - \frac{\beta}{2}\|y\|^2,$$

where $g^*(y) = \sup_z\langle z,\ y\rangle - g(z)$ is the Fenchel conjugate of g. Note that g_β is convex and $\frac{1}{\beta}$-smooth. The present work focuses on the case when $g(\cdot,\xi) = \delta_{b(\xi)}(\cdot)$. Smoothing the indicator function is studied in the context of proximal methods by Tran-Dinh et al. (2018); Fercoq et al. (2019) and for deterministic CGM by Yurtsever et al. (2018). Of particular note is that when $g(x) = \delta_{\mathcal{X}}(x)$, the smoothed function becomes $g_\beta(x) = \frac{1}{2\beta}\operatorname{dist}(x,\mathcal{X})^2$.

Optimality Conditions. We denote by x^* a solution to problem (1) and say that x is an ϵ-solution for (1) if it satisfies

$$\mathbb{E}\left[|f(x,\xi)-f(x^*)|\right]\le\epsilon,\quad \sqrt{\mathbb{E}\left[\operatorname{dist}(A(\xi)x,b(\xi))^2\right]}\le\epsilon. \tag{2}$$

Oracles. Our complexity results are given relative to the following oracles:

- **Stochastic first order oracle (sfo):** For a stochastic function $\mathbb{E}\left[f(\cdot,\xi)\right]$ with $\xi\sim P$, the sfo returns a pair $(f(x,\xi),\nabla f(x,\xi))$ where ξ is an i.i.d. sample from P (Nemirovsky & Yudin, 1983).
- **Incremental first order oracle (ifo):** For finite-sum problems, the ifo takes an index $i\in[n]$ and returns a pair $(f_i(x),\nabla f_i(x))$.
- **Linear minimization oracle (lmo):** The linear minimization oracle of set $\mathcal{X}$ is given by $\operatorname{lmo}_{\mathcal{X}}(y) = \arg\min_{x\in\mathcal{X}}\langle x,\ y\rangle$ and is assumed to be efficient to compute throughout this paper. This is the main projection-free oracle model for CGM-type methods.

4. Algorithms & Convergence

We now describe our proposed methods for solving (1), H-1SFW and H-SPIDER-FW, and provide their theoretical convergence guarantees.

4.1. Challenges and High-Level Ideas

Problem (1) can be rewritten equivalently as:

$$\min_{x\in\mathcal{X}} F(x) := \mathbb{E}\left[f(x,\xi)+\delta_{b(\xi)}(A(\xi)x)\right]. \tag{3}$$

Note that, in this form, our objective is non-smooth due to the indicator function. In order to leverage the conditional gradient framework, we *smooth* $\delta_{b(\xi)}(A(\xi)x)$ through the technique described in Section 3, thus obtaining a surrogate objective F_β. For notational simplicity, we refer to the smoothed stochastic indicator as:

$$g_\beta(A(\xi)x) = \frac{1}{2\beta}\operatorname{dist}(A(\xi)x,b(\xi))^2. \tag{4}$$

The minimization problem in terms of the smoothed objective thus becomes:

$$\min_{x\in\mathcal{X}} F_\beta(x) := \mathbb{E}\left[f(x,\xi)+g_\beta(A(\xi)x)\right], \tag{5}$$

with $\lim_{\beta\to0} F_\beta(x) = F(x)$. A natural idea is to optimize smooth approximations F_β which are progressively more accurate representations of F. To this end, we apply *conditional gradient* steps in conjunction with decreasing the smoothness parameter β, practically emulating a homotopy

transformation. As the iterations unfold our algorithms in fact approach the optimum of the original objective $F(x)$, as stated theoretically in Sections 4.3.2 and 4.4.2.

However, the aforementioned idea faces a technical challenge: decreasing the smoothing parameter β impacts the variance of the stochastic gradients $\nabla_x g_\beta(A(\xi)x)$, which increases proportionally. This issue has previously been signaled in the work of (Fercoq et al., 2019), where the authors address a similar setting using stochastic proximal gradient steps. Here, the problem is further aggravated by the use of lmo calls over $\mathcal{X}$, as it is well-known that CGMs are sensitive to non-vanishing gradient noise (Mokhtari et al., 2018).

Our solution is to simply perform VR on the stochastic gradients and theoretically establish a rate for $\beta \to 0$ in order to counteract the exploding variance. Precisely, we show how two different VR schemes can be successfully used within the homotopy framework:

- H-1SFW uses one stochastic sample to update a gradient estimator at every iteration, following the technique introduced in (Mokhtari et al., 2018). Depending on computational resources, the single-sample model can be extended to a fixed batch size with the same convergence guarantees.
- H-SPIDER-FW uses stochastic mini-batches of increasing size to compute the gradient estimator, using the technique proposed in (Fang et al., 2018).

The theoretical results characterizing our algorithms are presented in sections refsec:h1sfw and 4.4. First, we state the rate at which the β-dependent stochastic gradient noise vanishes under each VR scheme in lemmas 4.1 and 4.2. The main convergence theorems 4.1 and 4.2 then describe the performance of our algorithms in terms of the quantity $\mathbb{E}\left[S_{\beta_k}(x_k,\xi)\right] := \mathbb{E}\left[F_{\beta_k}(x_k,\xi) - f(x^*)\right]$, called the *smoothed gap*. Finally, in corollaries 4.1 and 4.2 we translate the aforementioned results into guarantees over the objective residual and constraint feasibility. All proofs are deferred to the appendix due to lack of space.

4.2. Technical Assumptions

Assumption 4.1. *The stochastic functions $f(\cdot,\xi)$ are convex and L_f-smooth. This further implies that $f(x)$ is L_f-smooth.*

Assumption 4.2. *The stochastic gradients $\nabla f(x,\xi)$ are unbiased and have a uniform variance bound σ_f^2. Formally,*

$$\mathbb{E}\left[\nabla f(x,\xi)\right] = \nabla f(x)$$
$$\mathbb{E}\left[\left\|\nabla f(x,\xi) - \nabla f(x)\right\|^2\right] \leq \sigma_f^2 < +\infty. \tag{6}$$

Algorithm 1 H-1SFW

Input: $x_1 \in \mathcal{X}, \beta_0 > 0, P(\xi)$
for $k = 1, 2, \ldots,$ **do**
  Set ρ_k, β_k and γ_k; sample $\xi_k \sim P(\xi)$
  $d_k = (1-\rho_k)d_{k-1} + \rho_k \nabla_x F_{\beta_k}(x_k, \xi_k)$
  $w_k = \text{lmo}_{\mathcal{X}}(d_k)$
  $x_{k+1} = x_k + \gamma_k(w_k - x_k)$.
end for

Assumption 4.3. *The domain $\mathcal{X}$ is convex and compact, with diameter $\mathcal{D}_{\mathcal{X}}$.*

Assumption 4.4. *Slater's condition holds for problem (3). Specifically, letting $G : \mathcal{H} \to \mathbb{R} \cup \{\infty\}$, $G(Ax) := \mathbb{E}\left[\delta_{b(\xi)}(A(\xi)x)\right]$, with the linear operator $A : \mathbb{R}^d \to \mathcal{H}$ defined as $(Ax)(\xi) := A(\xi)x,\ \forall x$, we require that*

$$0 \in \text{sri}\left(\text{dom}(G) - A\,\text{dom}(f)\right),$$

where sri *is the strong relative interior of the set (Bauschke et al.).*

Assumption 4.5. *The spectral norm of the stochastic linear operator $A(\xi)$ is uniformly bounded by a constant L_A:*

$$L_A := \sup_{\xi} \|A(\xi)\|^2 < +\infty.$$

This assumption is also made in (Fercoq et al., 2019).

4.3. H(omotopy)-1SFW

We now describe our first algorithm which relies on the VR scheme proposed in (Mokhtari et al., 2018), and whose advantage lies in a simple update rule and single-loop structure.

4.3.1. Gradient Estimator Model

We denote the gradient estimator by d_k, and remark that it is biased with respect to the true gradient $\nabla F_\beta(x_k)$ and exhibits a vanishing variance. This scheme achieves VR while conveniently considering only one stochastic constraint at a time. The estimator update rule is given by

$$d_k = (1-\rho_k)d_{k-1} + \rho_k \nabla F_{\beta_k}(x_k, \xi_k),$$

where $\nabla F_{\beta_k}(x_k,\xi_k) = \nabla f(x_k,\xi_k) + \nabla g_{\beta_k}(A(\xi_k)X_k)$, and ρ_k is a decaying convex combination parameter. The proposed method is provided via pseudocode in Algorithm 1.

4.3.2. Convergence Results

Before stating the results, we remark that Lemma 4.1 is the counterpart of Lemma 1 in (Mokhtari et al., 2018) and its

proof follows a similar route, up to bounding β-dependent quantities. It is worth noting that in our case, handling the stochastic linear inclusion constraints results in a rate surcharge factor of $\mathcal{O}\left(k^{1/3}\right)$.

Lemma 4.1. *Let* $\rho_k = \frac{3}{(k+5)^{2/3}}$, $\gamma_k = \frac{2}{k+1}$, $\beta_k = \frac{\beta_0}{(k+1)^{1/6}}$, $\beta_0 > 0$ *in Algorithm 1. Then, for all k,*

$$\mathbb{E}\left[\|\nabla F_{\beta_k}(x_k) - d_k\|^2\right] \leq \frac{C_1}{(k+5)^{1/3}},$$

where $C_1 = \max\left\{6^{1/3}\|\nabla F_{\beta_0}(x_0) - d_0\|^2, \right.$
$\left. 2\left[18\sigma_f^2 + 112L_f^2\mathcal{D}_{\mathcal{X}}^2 + \frac{522L_A^2\mathcal{D}_{\mathcal{X}}^2}{\beta_0^2}\right]\right\}$

Theorem 4.1. *Consider Algorithm 1 with parameters* $\rho_k = \frac{3}{(k+5)^{2/3}}$, $\gamma_k = \frac{2}{k+1}$, $\beta_k = \frac{\beta_0}{(k+1)^{1/6}}$, $\beta_0 > 0$ *(identical to Lemma 4.1). Then, for all k,*

$$\mathbb{E}\left[S_{\beta_k}(x_{k+1})\right] \leq \frac{C_2}{k^{1/6}},$$

where $C_2 = \max\left\{S_0(x_1),\ b = 2\mathcal{D}_{\mathcal{X}}\sqrt{C_1} + 2\mathcal{D}_{\mathcal{X}}^2\left(L_f + \frac{L_A}{\beta_0}\right)\right\}$ *and* C_1 *is defined in Lemma 4.1.*

Corollary 4.1. *The expected convergence in terms of objective suboptimality and feasibility of Algorithm 1 is, respectively,*

$$\begin{aligned}\mathbb{E}\left[|f(x_k,\xi) - f(x^*)|\right] &\in \mathcal{O}\left(k^{-1/6}\right)\\ \sqrt{\mathbb{E}\left[\mathrm{dist}(A(\xi)x_k, b(\xi))^2\right]} &\in \mathcal{O}\left(k^{-1/6}\right).\end{aligned}$$

Consequently, the oracle complexity is $\#(sfo) \in \mathcal{O}\left(\epsilon^{-6}\right)$ *and* $\#(lmo) \in \mathcal{O}\left(\epsilon^{-6}\right)$.

4.4. H(omotopy)-SPIDER-FW

Our second algorithm presents a more complex VR scheme, which improves on the complexity of H-1SFW. The method relies on the SPIDER estimator originally proposed under the framework of Normalized Gradient Descent in (Fang et al., 2018) and further studied for CGMs in (Yurtsever et al., 2019b). Different from Section 4.3.2, the results that follow distinguish two scenarios: the first is customary to VR methods such as SVRG (Johnson & Zhang, 2013) or SARAH (Nguyen et al., 2017) and assumes a finite-sum form of f; the second, different from most other VR schemes, caters to objectives of the form $f(x) = \mathbb{E}\left[f(x,\xi)\right]$ where $\xi \sim P(\xi)$, and can handle a potentially infinite number of stochastic functions of (1).

Algorithm 2 H-SPIDER-FW

Input: $\bar{x}_1 \in \mathcal{X}, \beta_0 > 0, P(\xi)$
for $t = 1, 2, \ldots, T$ **do**
  $x_{t,1} = \bar{x}_t$
  Compute $\gamma_{t,1}, \beta_{t,1}, K_t$; sample $\xi_{\mathcal{Q}_t} \overset{\text{i.i.d}}{\sim} P(\xi)$
  $v_{t,1} = \tilde{\nabla} F_{\beta_{t,1}}(x_{t,1}, \xi_{\mathcal{Q}_t})$
  $w_{t,1} \in \mathrm{lmo}_{\mathcal{X}}(v_{t,1})$
  $x_{t,2} = x_{t,1} + \gamma_{t,1}(w_{t,1} - x_{t,1})$
  for $k = 2, \ldots, K_t$ **do**
    Compute $\gamma_{t,k}, \beta_{t,k}$; sample $\xi_{\mathcal{S}_{t,k}} \overset{\text{i.i.d}}{\sim} P(\xi)$
    $v_{t,k} = v_{t,k-1} - \tilde{\nabla} F_{\beta_{t,k-1}}(x_{t,k-1}, \xi_{\mathcal{S}_{t,k}})$
      $+ \tilde{\nabla} F_{\beta_{t,k}}(x_{t,k}, \xi_{\mathcal{S}_{t,k}})$
    $w_{t,k} \in \mathrm{lmo}_{\mathcal{X}}(v_{t,k})$
    $x_{t,k+1} = x_{t,k} + \gamma_{t,k}(w_{t,k} - x_{t,k})$
  end for
  Set $\bar{x}_{t+1} = x_{t,K_t+1}$
end for

4.4.1. Gradient Estimator Model

We denote the SPIDER gradient estimator by $v_{t,k}$, and remark that it is also biased relative to $\nabla F_{\beta_k}(x_k)$ and exhibits a vanishing variance. This scheme achieves VR through the use of increasing-size mini-batches. The estimator update rule is given by

$$\begin{aligned}v_{t,k} = v_{t,k-1} &- \tilde{\nabla} F_{\beta_{t,k-1}}(x_{t,k-1}, \xi_{\mathcal{S}_{t,k}})\\ &+ \tilde{\nabla} F_{\beta_{t,k}}(x_{t,k}, \xi_{\mathcal{S}_{t,k}}),\end{aligned} \tag{7}$$

where $\tilde{\nabla} F_{\beta_{t,k}}(x_{t,k}, \xi_{\mathcal{S}_{t,k}}) = \tilde{\nabla} f(x_k, \xi_{\mathcal{S}_{t,k}}) + \tilde{\nabla} g_{\beta_{t,k}}(A(\xi_{\mathcal{S}_{t,k}})x_{t,k})$ defines the averaged gradient over a mini-batch of size $|\mathcal{S}_{t,k}|$.

The double indexing used in (7) hints at the double-loop structure of the algorithm, a format similar to most VR-based methods. The method is structured similarly to SPIDER-FW from (Yurtsever et al., 2019b), and proceeds in two steps: the outer loop computes an accurate gradient estimator and sets the batch size for the inner iterations. The inner-loop then iteratively 'refreshes' this gradient according to (7) and performs homotopy steps on β using a theoretically-determined schedule. The proposed method is provided via pseudocode in Algorithm 2.

4.4.2. Convergence Results

Again, we remark that Lemma 4.2 is the counterpart of Lemma 4, Appendix C in (Yurtsever et al., 2019b). However in this case, our proof takes a different, more tedious route, as the latter result does not accommodate homotopy steps.

In comparison, the bound we obtain depends linearly on the total iteration count, whereas the lemma of (Yurtsever et al., 2019b) depends only on the outer loop counter K_t.

Lemma 4.2 (Estimator variance for finite-sum problems)**.** *Consider Algorithm 2, and let ξ be finitely sampled from set $[n]$, $\xi_{\mathcal{Q}_t} = [n]$ and $\xi_{\mathcal{S}_{t,k}}$, such that $|\mathcal{S}_{t,k}| = K_t = 2^{t-1}$. Also, let $\gamma_{t,k} = \frac{2}{K_t+k}$, $\beta_{t,k} = \frac{\beta_0}{\sqrt{K_t+k}}$, $\beta_0 > 0$. Then, for a fixed t and for all $k \leq K_t$,*

$$\mathbb{E}\left[\left\|\nabla F_{\beta_{t,k}}(x_{t,k}) - v_{t,k}\right\|^2\right] \leq \frac{C_1}{K_t + k},$$

where $C_1 = 2\mathcal{D}_{\mathcal{X}}^2\left(8L_f^2 + \frac{98L_A^2}{\beta_0^2}\right)$.

Lemma 4.3 (Estimator variance for general expectation problems)**.** *Consider Algorithm 2 and let $\xi \sim P(\xi)$ and $\xi_{\mathcal{Q}_t}$ such that $|\mathcal{Q}_t| = \lceil \frac{2K_t}{\beta_{t,1}^2} \rceil$. Also, let $\xi_{\mathcal{S}_{t,k}}$, such that $|\mathcal{S}_{t,k}| = K_t = 2^{t-1}$, $\gamma_{t,k} = \frac{2}{K_t+k}$, $\beta_{t,k} = \frac{\beta_0}{\sqrt{K_t+k}}$, $\beta_0 > 0$. Then, for a fixed t and for all $k \leq K_t$,*

$$\mathbb{E}\left[\left\|\nabla F_{\beta_{t,k}}(x_{t,k}) - v_{t,k}\right\|^2\right] \leq \frac{C_2}{K_t + k},$$

where $C_2 = 16L_f^2\mathcal{D}_{\mathcal{X}}^2 + 2L_A^2\mathcal{D}_{\mathcal{X}}^2\left(\frac{98}{\beta_0^2} + 1\right) + 2\beta_0^2\sigma_f^2$.

Theorem 4.2. *Consider Algorithm 2 with parameters $\gamma_{t,k} = \frac{2}{K_t+k}$, $\beta_{t,k} = \frac{\beta 0}{\sqrt{K_t+k}}$, $\beta_0 > 0$, and $\xi_{\mathcal{S}_{t,k}}$, such that $|\mathcal{S}_{t,k}| = K_t = 2^{t-1}$. Then,*

- *For ξ be finitely sampled from set $[n]$, $\xi_{\mathcal{Q}_t} = [n]$ and $\forall t \in \mathbb{N}$, $1 \leq k \leq 2^{t-1}$,*

$$\mathbb{E}\left[S_{\beta_{t,k}}(x_{t,k+1})\right] \leq \frac{C_3}{\sqrt{K_t + k + 1}},$$

$$\text{where } C_3 = \max\left\{S_{\beta_{1,0}}(x_{1,1}),\right.$$
$$\left. 2\mathcal{D}_{\mathcal{X}}^2 L_f + 2\mathcal{D}_{\mathcal{X}}^2\sqrt{16L_f^2 + \frac{196L_A^2}{\beta_0^2}} + \frac{2\mathcal{D}_{\mathcal{X}}^2 L_A}{\beta_0}\right\};$$

- *For $\xi \sim P(\xi)$, $\xi_{\mathcal{Q}_t}$ such that $|\mathcal{Q}_t| = \lceil \frac{2K_t}{\beta_{t,1}^2} \rceil$ and $\forall t \in \mathbb{N}$, $1 \leq k \leq 2^{t-1}$,*

$$\mathbb{E}\left[S_{\beta_{t,k}}(x_{t,k+1})\right] \leq \frac{C_4}{\sqrt{K_t + k + 1}},$$

$$\text{where } C_4 = \max\left\{S_{\beta_{1,0}}(x_{1,1}),\ 2\mathcal{D}_{\mathcal{X}}^2 L_f + \frac{2\mathcal{D}_{\mathcal{X}}^2 L_A}{\beta_0}\right.$$
$$\left. + 2\mathcal{D}_{\mathcal{X}}\sqrt{16L_f^2\mathcal{D}_{\mathcal{X}}^2 + 2L_A^2\mathcal{D}_{\mathcal{X}}^2\left(\frac{98}{\beta_0^2} + 1\right) + 2\beta_0^2\sigma_f^2}\right\}.$$

Corollary 4.2. *The expected convergence in terms of objective suboptimality and feasibility of Algorithm 2 is, respectively,*

$$\mathbb{E}\left[|f(x_{t,k}) - f(x^*)|\right] \in \mathcal{O}\left((K_t + k)^{-1/2}\right)$$
$$\sqrt{\mathbb{E}\left[\operatorname{dist}(A(\xi)x_{t,k}, b(\xi))^2\right]} \in \mathcal{O}\left((K_t + k)^{-1/2}\right)$$

for both the finite-sum and the general expectation setting, up to constants. Consequently, the oracle complexities are given by $\#(ifo) \in \mathcal{O}\left(n\log_2(\epsilon^{-2}) + \epsilon^{-4}\right)$ and $\#(lmo) \in \mathcal{O}\left(\epsilon^{-2}\right)$ for the finite-sum setting, and by $\#(sfo) \in \mathcal{O}\left(\epsilon^{-4}\right)$ and $\#(lmo) \in \mathcal{O}\left(\epsilon^{-2}\right)$ for the more general expectation setting.

4.5. Discussion

Rate Degradation in the Absence of Projection Oracles. Compared to proximal methods for solving (1), our algorithms require $\mathcal{O}(\epsilon^{-2})$ times more sfo calls to reach an ϵ-solution. This is well-known for CG-based methods: for instance, solving a fully deterministic version of (1) using the Augmented Lagrangian framework has a gradient complexity of $\mathcal{O}(\epsilon^{-1})$ (Xu, 2017), whereas the best known complexity for CG-based algorithms is $\mathcal{O}(\epsilon^{-2})$ (Yurtsever et al., 2018).

Comparison with SHCGM (Locatello et al., 2019). The state-of-the-art for solving (1) is the half-stochastic method SHCGM (Locatello et al., 2019), in which stochasticity is restricted to the objective function f, while the constraints are processed deterministically. This algorithm attains an $\mathcal{O}(\epsilon^{-3})$ sfo complexity and an $\mathcal{O}(\epsilon^{-3})$ lmo complexity, by resorting to the same VR scheme as H-1SFW applied only to $f(x, \xi)$. Since SHCGM handles the constraints deterministically, it does not face the challenge of exploding variance as $\beta \to 0$.

Our analysis shows that handling the β-dependence of the gradient noise comes at the price of H-1SFW being $\mathcal{O}(\epsilon^{-3})$ times more expensive in terms of both oracles. In contrast, owing to a more powerful variance-reduction scheme, H-SPIDER-FW attains only an $\mathcal{O}(\epsilon)$-times worse sfo complexity, while improving by an $\mathcal{O}(\epsilon)$ factor in terms of the lmo complexity. Given that an lmo call is generally more expensive than that of an sfo, we have in fact *improved* the complexity over the state-of-the-art, while being the first to process linear constraints stochastically. Moreover, we note that the lmo complexity of H-SPIDER-FW is on the same order as its fully deterministic counterpart, the HCGM (Yurtsever et al., 2018).

The Role of VR. The choice of VR technique dictates the worst-case convergence guarantees of our methods, a fact which is apparent from the discrepancy between the variance

bounds of Lemmas 4.1 and 4.2- 4.3, respectively: $\mathcal{O}(k^{-1/3})$ for d_k vs. $\mathcal{O}(k^{-1})$ for $v_{t,k}$. This signals the existence of a trade-off: a more intricate way of handling stochastic penalty-type constraints can ensure the better convergence guarantees of H-SPIDER-FW, while a simpler VR scheme comes at the cost of the rather pessimistic ones of H-1SFW. Fortunately, as shown in the Section 5, the simple H-1SFW greatly outperforms its worst-case guarantees.

5. Numerical Experiments

For demonstrating the empirical efficiency of our algorithms, we apply them to three problem instances: synthetically-generated SDPs, the K-means clustering SDP relaxation and the Sparsest Cut-associated SDP.

Evaluation Metrics: Our experiments subscribe to a finite-sum template, where we define $f(x) := \sum_{i=1}^{n_1} f_i(x)$ and $g_\beta(Ax) = \sum_{i=1}^{n_2} g_{i,\beta}(A_i^T x)$. The objective convergence is recorded as $|f(x)-f^\star|$, with $f^\star := f(x^*)$. Due to imperfect feasibility, the value of $f(x)$ can overshoot $f^\star$, since the constrained optimum is not the global one. This usually appears as the increase of $|f(x) - f^\star|$ immediately after a significant drop when the quantity $f(x) - f^\star$ becomes negative; then the decreasing trend restarts, as the objective and constraints re-balance. Such a phenomenon is common for homotopy-based methods, see for instance (Yurtsever et al., 2018). Lastly, the feasibility is recorded as $\|Ax - b\|$.

Baseline: To the best of our knowledge, the HCGM (Yurtsever et al., 2018) and the SHCGM (Locatello et al., 2019) are the only algorithms which tackle SDPs under the conditional gradient framework. The latter represents the empirical state-of-the-art and we choose it as the baseline for our experiments.

5.1. Synthetic SDP Problems

This proof-of concept experiment aims to show the performance of our fully stochastic methods, given a fixed problem dimension and an increasing set of constraints. We consider the synthetic SDP:

$$\begin{aligned} \min_{\substack{X\in\mathbb{S}_+^d \\ \mathrm{tr}(X)\le \frac{1}{d}}} \quad & \langle C, X\rangle \\ \text{subject to} \quad & \mathrm{tr}(A_i X) = b_i, i = 1\dots n \end{aligned}$$

where the entries of A_i and C are generated from $\mathcal{U}(0,1)$, and $b_i = \langle A_i, X^*\rangle$ for a fixed X^*. We perform uniform sampling on the pairs (A_i, b_i) for computing their stochastic gradients in our algorithms. We fix the dimension to be $d = 20$ and vary the size of constraints with $n = 5\mathtt{e}2$ and $5\mathtt{e}3$.

For a fair comparison, we sweep the parameter β_0 for the three algorithms in the range $[1\mathtt{e}{-7}, 1\mathtt{e}1]$. We settle for $1\mathtt{e}{-7}$, $1\mathtt{e}{-7}$ and $1\mathtt{e}{-5}$ for SHCGM, H-1SFW and H-SPIDER-FW, respectively. For H-1SFW and SHCGM, we choose the batch size to be 1% of the data.

Figure 1 illustrates the outcome of the experiments, where we observe a clear improvement of the stochastic algorithms over the baseline with a stable margin throughout the test cases.

Interestingly, H-1SFW exhibits strong empirical performance on the synthetic data, much better than its theoretical worst-case bound. A possible explanation is that the entries of C and A_i are generated from a "benign" distribution and *concentrate* around its mean (Ledoux, 2001). In such scenarios, even a small subset of constraints allows for effective variance reduction. For comparison, we provide an additional set of results for synthetic SDPs generated from a less well-behaved distribution in Appendix A.2. Nevertheless, we observe the same good performance of H-1SFW even with real data, in the next sections.

Regarding H-SPIDER-FW, we observe that the suboptimality and feasibility decrease at the rate $k^{-\frac{1}{2}}$ and $k^{-\frac{3}{4}}$, respectively, which is better than the worst-case bounds in Theorem 4.2.

5.2. The K-means Clustering Relaxation

We consider the unsupervised learning task of partitioning d data points into k clusters. We adopt the SDP formulation in (Peng & Wei, 2007), which amounts to solving:

$$\begin{aligned} \min_{X\in\mathcal{X}} \quad & \langle C, X\rangle \\ \text{subject to} \quad & X\vec{1} = \vec{1}, \\ & X_{i,j} \ge 0, \;\; 1 \le i, j \le d. \end{aligned} \tag{8}$$

Here, $C \in \mathbb{R}^{d\times d}$ is the Euclidean distance matrix of the d data points, $\mathcal{X} = \{X \in \mathbb{R}^{d\times d} : X \succeq 0,\ \mathrm{tr}(X) \le k\}$, $\vec{1}$ is the all 1's vector. Notice that the number of linear constraints in (8) is $\mathcal{O}(d^2)$.

In order to compare against existing work, we adopt the MNIST dataset ($k = 10$) (LeCun & Cortes, 2010) with $d = 10^3$ samples and perform data preprocessing as in (Mixon et al., 2016). The very same setup appeared in several works (Mixon et al., 2016; Yurtsever et al., 2018; Locatello et al., 2019), with SHCGM (Locatello et al., 2019) showing the best practical performance.

We perform parameter sweeping on $\beta_0 \in [1\mathtt{e}{-7}, 1\mathtt{e}2]$ for H-1SFW and H-SPIDER-FW, and settle for $5\mathtt{e}{-2}$ and $6\mathtt{e}0$, respectively. For SHCGM, we adopt the same hyperparameter as in (Locatello et al., 2019). The batchsize for H-1SFW and SHCGM is set to 5%.

The comparison of our algorithms against SHCGM is reported in Figure 2. H-1SFW and H-SPIDER-FW converge

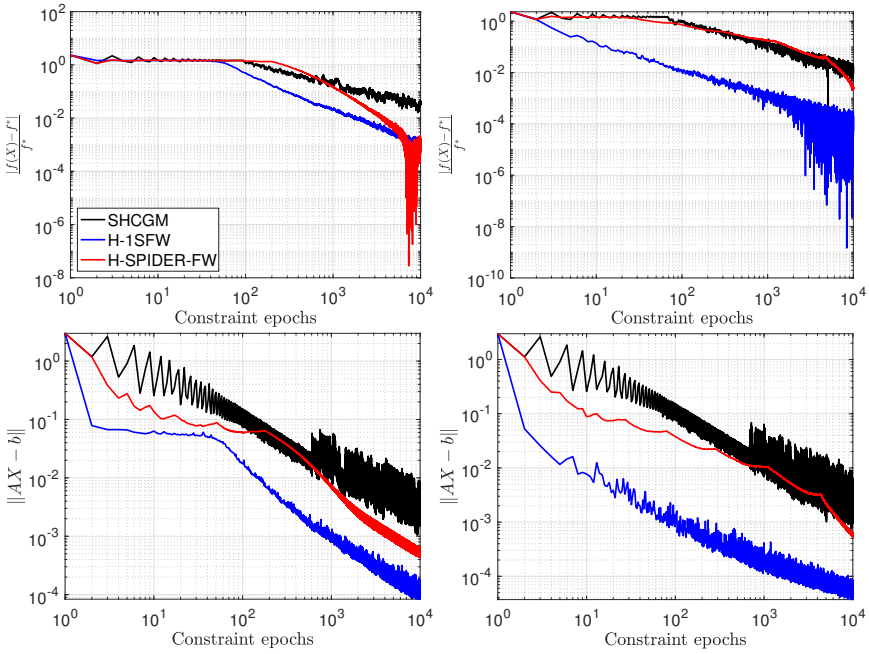

Figure 1. Synthetic SDPs, with each column showing the convergence in objective suboptimality (top) and in feasibility (bottom) for a specific problem. The left hand-side column corresponds to a problem with `5e2` constraints, while the right hand-side one to a problem with `5e3` constraints.

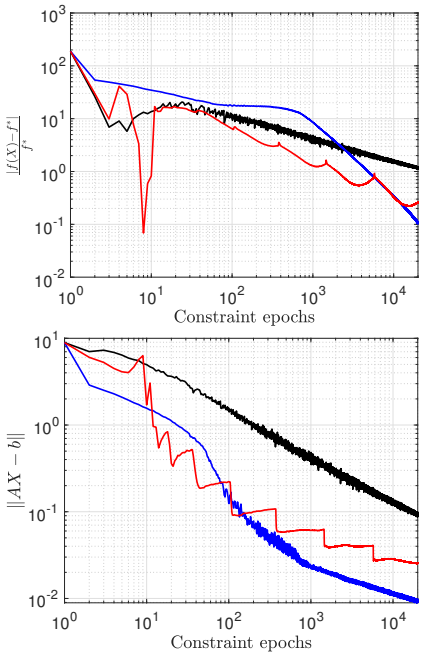

Figure 2. The K-means SDP relaxation, with convergence in objective suboptimality (top) and in feasibility (bottom).

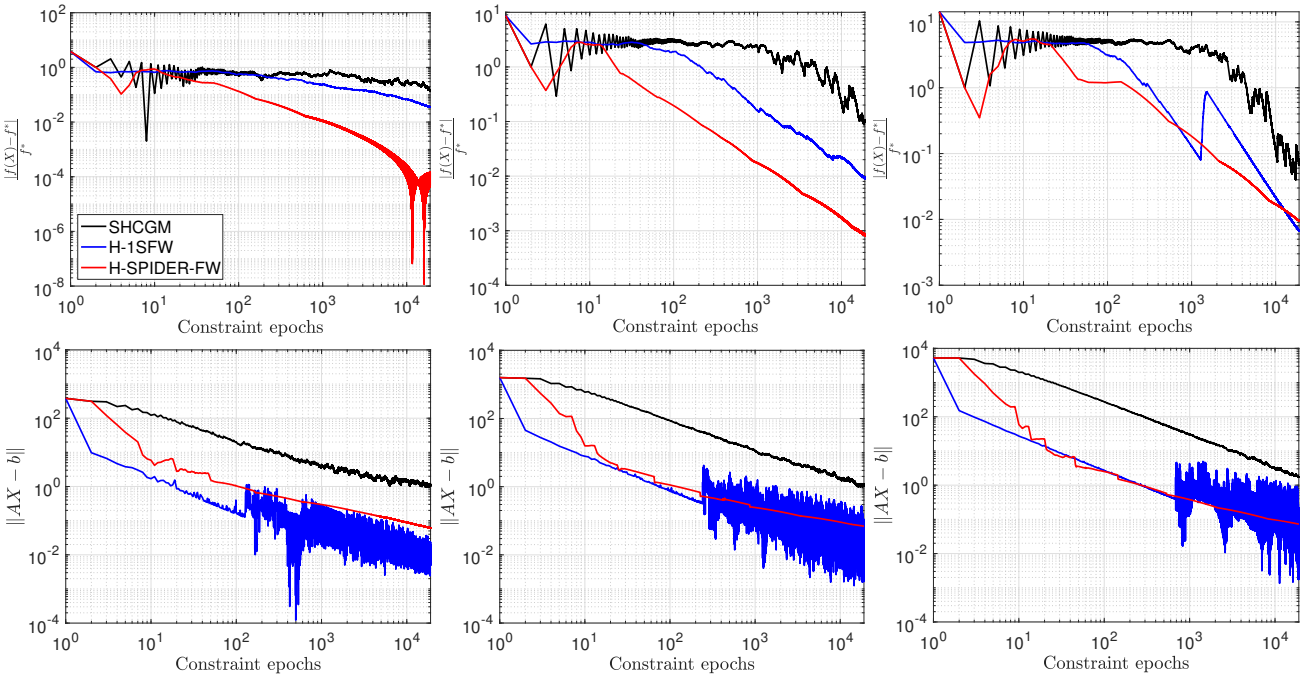

Figure 3. The Sparsest Cut-associated SDP relaxation, where each column shows the convergence in objective suboptimality (top) and feasibility (bottom) for a specific problem. From left to right, the results correspond to graphs *mammalia-primate-association-13*, *insecta-ant-colony1-day37* and *insecta-ant-colony4-day10*, sorted by increasing size.

Table 1. Details of the Network Repository (Rossi & Ahmed, 2015) graphs used in the experiments.

Graph name	$\|V\|$	$\|E\|$	Avg. node degree	Max. node degree	USC SDP dimension	USC SDP # constraints
mammalia-primate-association-13	25	181	14	19	$X \in \mathbb{R}^{25\times 25}$	$\sim$ 6.90e3
insecta-ant-colony1-day37	55	1k	42	53	$X \in \mathbb{R}^{55\times 55}$	$\sim$ 7.87e4
insecta-ant-colony4-day10	102	4k	79	99	$X \in \mathbb{R}^{102\times 102}$	$\sim$ 5.15e5

at a comparable rate, with both clearly overtaking the baseline with regards to objective suboptimality and feasibility convergence.

5.3. Computing an ℓ_2^2 Embedding for the Uniform Sparsest Cut Problem

The Uniform Sparsest Cut problem (USC) aims to find a bipartition $(S, \bar{S})$ of the nodes of a graph $G = (V, E)$, $|V| = d$, which minimizes the quantity

$$\frac{E(S, \bar{S})}{|S||\bar{S}|},$$

where $E(S, \bar{S})$ is the number of edges connecting S and $\bar{S}$. This problem is of broad interest, with applications in areas such as VLSI layout design, topological design of communication networks and image segmentation, to name a few. Relevant to machine learning, it appears as a subproblem in hierarchical clustering algorithms (Dasgupta, 2016; Chatziafratis et al., 2018).

Computing such a bipartition is NP-hard and intense research has gone into designing efficient approximation algorithms for this problem. In the seminal work of Arora et al. (2009) an $\mathcal{O}\left(\sqrt{\log d}\right)$ approximation algorithm is proposed for solving USC, which relies on finding a *well-spread* ℓ_2^2 geometric representation of G where each node $i \in V$ is mapped to a vector v_i in $\mathbb{R}^d$. In this experimental section we focus on solving the SDP that computes this geometric embedding, as its high number of triangle inequality constraints ($\mathcal{O}\left(d^3\right)$) makes it a suitable candidate for our framework. The canonical formulation of the SDP is given below (for the original formulation, see Appendix A.3).

$$\begin{aligned} \min_{X\in\mathcal{X}} \quad & \langle L, X\rangle \\ \text{subject to} \quad & d\,\mathrm{Tr}(X) - \mathrm{Tr}(\mathbf{1}_{d\times d}X) = \frac{d^2}{2} \\ & X_{i,j} + X_{j,k} - X_{i,k} - X_{j,j} \le 0, \quad \forall\, i,j,k \in V \end{aligned}$$

Here, L represents the Laplacian of G, $\mathcal{X} = \{X \in \mathbb{R}^{d\times d} : X \succeq 0,\ \mathrm{tr}(X) \le d\}$ and $X_{i,j} = \langle v_i,\ v_j\rangle$ gives the geometric embedding of the nodes. We run our algorithms on three graphs of different sizes from the Network Repository dataset (Rossi & Ahmed, 2015), whose details are summarized in Table 1. Note the cubic dependence of the number of constraints relative to the number of nodes. We perform parameter sweeping on $\beta_0 \in [\texttt{1e-5}, \texttt{1e5}]$ using the smallest graph, *mammalia-primate-association-13*, and keep the same parameters for all the experiments. The values of β_0 for SHCGM, H-1SFW and H-SPIDER-FW are `1e2`, `1e-2` and `1e1` respectively, and the batch size for both H-1SFW and SHCGM is set to 5%.

Figure 3 depicts the outcomes of the experiments, with both our algorithms consistently outperforming SHCGM and H-SPIDER-FW attaining the fastest convergence. A possible explanation is that, given the much larger number of constraints relative to the problem dimension ($\mathcal{O}\left(n^3\right)$ v.s $\mathcal{O}\left(n^2\right)$), H-SPIDER-FW's increasing mini-batches readily reach an adequate balance between feasibility enforcement and objective minimization.

Acknowledgements

The authors are grateful to Mehmet Fatih Sahin and Alp Yurtsever for the helpful discussions throughout the development of this paper.

This work was partially supported by the Swiss National Science Foundation (SNSF) under grant number 200021_178865 / 1; the Army Research Office under Grant Number W911NF-19-1-0404; the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement no 725594 - time-data).

References

Allen-Zhu, Z. and Yuan, Y. Improved svrg for non-strongly-convex or sum-of-non-convex objectives. In *International conference on machine learning*, pp. 1080–1089, 2016.

Arora, S., Hazan, E., and Kale, S. Fast algorithms for approximate semidefinite programming using the multiplicative weights update method. In *46th Annual IEEE Symposium on Foundations of Computer Science (FOCS'05)*, pp. 339–348. IEEE, 2005.

Arora, S., Rao, S., and Vazirani, U. Expander flows, geometric embeddings and graph partitioning. *Journal of the ACM (JACM)*, 56(2):5, 2009.

Balasubramanian, K. and Ghadimi, S. Zeroth-order (non)-convex stochastic optimization via conditional gradient and gradient updates. In *Advances in Neural Information Processing Systems*, pp. 3455–3464, 2018.

Bauschke, H. H., Combettes, P. L., et al. *Convex analysis and monotone operator theory in Hilbert spaces*, volume 408. Springer.

Burer, S. and Monteiro, R. D. Local minima and convergence in low-rank semidefinite programming. *Mathematical Programming*, 103(3):427–444, 2005.

Chatziafratis, V., Niazadeh, R., and Charikar, M. Hierarchical clustering with structural constraints. *arXiv preprint arXiv:1805.09476*, 2018.

Clarkson, K. L. Coresets, sparse greedy approximation, and the frank-wolfe algorithm. *ACM Transactions on Algorithms (TALG)*, 6(4):63, 2010.

Dasgupta, S. A cost function for similarity-based hierarchical clustering. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pp. 118–127, 2016.

Defazio, A., Bach, F., and Lacoste-Julien, S. Saga: A fast incremental gradient method with support for non-strongly convex composite objectives. In *Advances in neural information processing systems*, pp. 1646–1654, 2014.

Fang, C., Li, C. J., Lin, Z., and Zhang, T. Spider: Near-optimal non-convex optimization via stochastic path-integrated differential estimator. In *Advances in Neural Information Processing Systems*, pp. 689–699, 2018.

Fercoq, O., Alacaoglu, A., Necoara, I., and Cevher, V. Almost surely constrained convex optimization. *arXiv preprint arXiv:1902.00126*, 2019.

Frank, M. and Wolfe, P. An algorithm for quadratic programming. *Naval Research Logistics Quarterly*, 3: 95–110, 1956. doi: 10.1002/nav.3800030109. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/nav.3800030109.

Garber, D. *Projection-free Algorithms for Convex Optimization and Online Learning*. PhD thesis, Technion-Israel Institute of Technology, Faculty of Industrial and , 2016.

Garber, D. and Hazan, E. Sublinear time algorithms for approximate semidefinite programming. *Mathematical Programming*, 158(1-2):329–361, 2016.

Gidel, G., Pedregosa, F., and Lacoste-Julien, S. Frank-wolfe splitting via augmented lagrangian method. In *International Conference on Artificial Intelligence and Statistics*, pp. 1456–1465, 2018.

Hazan, E. Sparse approximate solutions to semidefinite programs. In *Latin American symposium on theoretical informatics*, pp. 306–316. Springer, 2008.

Hazan, E. and Luo, H. Variance-reduced and projection-free stochastic optimization. In *International Conference on Machine Learning*, pp. 1263–1271, 2016.

Hazan, E. E. and Kale, S. Projection-free online learning. In *29th International Conference on Machine Learning, ICML 2012*, pp. 521–528, 2012.

Huang, Q., Chen, Y., and Guibas, L. Scalable semidefinite relaxation for maximum a posterior estimation. In *International Conference on Machine Learning*, pp. 64–72, 2014.

Iyengar, G., Phillips, D. J., and Stein, C. Feasible and accurate algorithms for covering semidefinite programs. In *Scandinavian Workshop on Algorithm Theory*, pp. 150–162. Springer, 2010.

Jaggi, M. Revisiting Frank-Wolfe: Projection-free sparse convex optimization. In Dasgupta, S. and McAllester, D. (eds.), *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pp. 427–435, Atlanta, Georgia, USA, 17–19 Jun 2013. PMLR. URL http://proceedings.mlr.press/v28/jaggi13.html.

Johnson, R. and Zhang, T. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in neural information processing systems*, pp. 315–323, 2013.

Lan, G. The complexity of large-scale convex programming under a linear optimization oracle. *arXiv preprint arXiv:1309.5550*, 2013.

Lan, G. and Zhou, Y. Conditional gradient sliding for convex optimization. *SIAM Journal on Optimization*, 26(2):1379–1409, 2016.

LeCun, Y. and Cortes, C. MNIST handwritten digit database. 2010. URL http://yann.lecun.com/exdb/mnist/.

Ledoux, M. *The concentration of measure phenomenon*. Number 89. American Mathematical Soc., 2001.

Liu, Y.-F., Liu, X., and Ma, S. On the nonergodic convergence rate of an inexact augmented lagrangian framework

for composite convex programming. *Mathematics of Operations Research*, 44(2):632–650, 2019.

Locatello, F., Yurtsever, A., Fercoq, O., and Cevher, V. Stochastic conditional gradient method for composite convex minimization. *arXiv preprint arXiv:1901.10348*, 2019.

Mahdavi, M., Zhang, L., and Jin, R. Mixed optimization for smooth functions. In *Advances in neural information processing systems*, pp. 674–682, 2013.

Mixon, D. G., Villar, S., and Ward, R. Clustering subgaussian mixtures by semidefinite programming. *arXiv preprint arXiv:1602.06612*, 2016.

Mokhtari, A., Hassani, H., and Karbasi, A. Stochastic conditional gradient methods: From convex minimization to submodular maximization. *arXiv preprint arXiv:1804.09554*, 2018.

Nemirovsky, A. S. and Yudin, D. B. Problem complexity and method efficiency in optimization. 1983.

Nesterov, Y. Smooth minimization of non-smooth functions. *Mathematical programming*, 103(1):127–152, 2005.

Nguyen, L. M., Liu, J., Scheinberg, K., and Takáč, M. Sarah: A novel method for machine learning problems using stochastic recursive gradient. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 2613–2621. JMLR. org, 2017.

Patrascu, A. and Necoara, I. Nonasymptotic convergence of stochastic proximal point methods for constrained convex optimization. *Journal of Machine Learning Research*, 18: 198–1, 2017.

Peng, J. and Wei, Y. Approximating k-means-type clustering via semidefinite programming. *SIAM journal on optimization*, 18(1):186–205, 2007.

Rossi, R. A. and Ahmed, N. K. The network data repository with interactive graph analytics and visualization. In *AAAI*, 2015. URL http://networkrepository.com.

Roux, N. L., Schmidt, M., and Bach, F. R. A stochastic gradient method with an exponential convergence _rate for finite training sets. In *Advances in neural information processing systems*, pp. 2663–2671, 2012.

Schmidt, M., Le Roux, N., and Bach, F. Minimizing finite sums with the stochastic average gradient. *Mathematical Programming*, 162(1-2):83–112, 2017.

Silveti-Falls, A., Molinari, C., and Fadili, J. Generalized conditional gradient with augmented lagrangian for composite minimization. *arXiv preprint arXiv:1901.01287*, 2019.

Tran-Dinh, Q., Fercoq, O., and Cevher, V. A smooth primal-dual optimization framework for nonsmooth composite convex minimization. *SIAM Journal on Optimization*, 28 (1):96–134, 2018.

Weinberger, K. Q. and Saul, L. K. Distance metric learning for large margin nearest neighbor classification. *Journal of Machine Learning Research*, 10(2), 2009.

Xiao, L. and Zhang, T. A proximal stochastic gradient method with progressive variance reduction. *SIAM Journal on Optimization*, 24(4):2057–2075, 2014.

Xu, Y. Accelerated first-order primal-dual proximal methods for linearly constrained composite convex programming. *SIAM Journal on Optimization*, 27(3):1459–1484, 2017.

Xu, Y. Primal-dual stochastic gradient method for convex programs with many functional constraints. *arXiv preprint arXiv:1802.02724*, 2018.

Yang, L., Sun, D., and Toh, K.-C. Sdpnal+: a majorized semismooth newton-cg augmented lagrangian method for semidefinite programming with nonnegative constraints. *Mathematical Programming Computation*, 7(3):331–366, 2015.

Yurtsever, A., Fercoq, O., Locatello, F., and Cevher, V. A conditional gradient framework for composite convex minimization with applications to semidefinite programming. In *35th International Conference on Machine Learning (ICML)*, pp. 5727–5736. PMLR, 2018.

Yurtsever, A., Fercoq, O., and Cevher, V. A conditional-gradient-based augmented lagrangian framework. In *International Conference on Machine Learning*, pp. 7272–7281, 2019a.

Yurtsever, A., Sra, S., and Cevher, V. Conditional gradient methods via stochastic path-integrated differential estimator. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 7282–7291, Long Beach, California, USA, 09–15 Jun 2019b. PMLR. URL http://proceedings.mlr.press/v97/yurtsever19b.html.

Zhang, M., Shen, Z., Mokhtari, A., Hassani, H., and Karbasi, A. One sample stochastic frank-wolfe. *arXiv preprint arXiv:1910.04322*, 2019.

Appendix

A. Additional Experiment Information

In this section we provide some omitted experiment details.

A.1. Experiment Setup

The experiments presented in this paper were implemented in MATLAB R2019b and executed on a 2,9 GHz 6-Core Intel Core i9 CPU with 32 GB RAM. For retrieving the values of $f(x^*)$ we used the code of (Mixon et al., 2016) which relies on SDPNAL+ (Yang et al., 2015) for the clustering experiments, and CVX for the Sparsest Cut ones. The code is included in the supplemental material.

A.2. Additional Results for Synthetic SDPs

The setup for these experiments is the same as that of Section 5.1, but with a different distribution for generating A_i and C. Specifically, we use the heavy-tailed Stable distribution with parameters $(\alpha = 1.5, \beta = 0, \gamma = 10, \delta = 0)$. We sweep β_0 for all three algorithms in the range $[\texttt{1e-7}, \texttt{1e-1}]$ and settle for $\texttt{1e-5}$, $\texttt{1e-7}$, $\texttt{1e-6}$ for SHCGM, H-1SFW and H-SPIDER-FW, respectively. The results are depicted in Figure 4.

We observe that, given this more difficult distribution, all methods are comparable in terms of convergence speed for both objective suboptimality and feasibility, with H-SPIDER-FW having an edge over the other two.

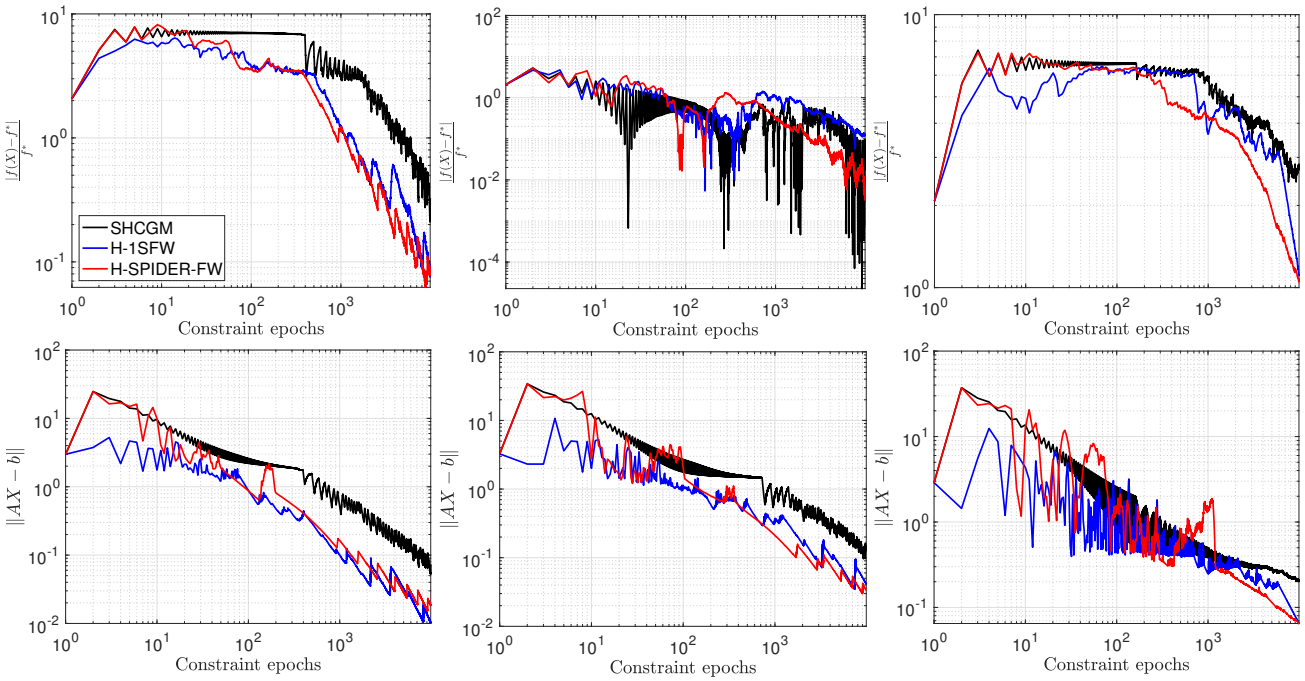

Figure 4. Synthetic SDPs, with each column showing the convergence in objective suboptimality (top) and in feasibility (bottom) for a specific problem. From left to right, the columns depict the results for problems with $\texttt{5e2}$, $\texttt{1e3}$ and $\texttt{5e3}$ constraints.

A.3. The Uniform Sparsest Cut SDP

The left column of Table 2 provides the original SDP formulation of (Arora et al., 2009) for finding the ℓ_2^2 embedding of nodes $i \in V$; the right column contains the corresponding canonical formulation. In our experiments we use the latter formulation to which we add the trace constraint $\mathrm{tr}(X) \leq d$. This additional constraint does not change the optimal objective (Iyengar et al., 2010).

Table 2. SDP formulations for retrieving the ℓ_2^2 embedding of graph nodes.

Original SDP	**Canonical SDP**
minimize $\frac{1}{d^2}\sum_{(i,j)\in E}\lvert v_i - v_j\rvert^2$	**minimize** $\mathrm{Tr}(LX)$
subject to $\sum_{\substack{i,j\in V\\ i\neq j}}\lvert v_i - v_j\rvert^2 = d^2$	**subject to** $d\mathrm{Tr}(X) - \mathrm{Tr}(\mathbf{1}_{d\times d}X) = \frac{d^2}{2}$
$\lvert v_i - v_j\rvert^2 + \lvert v_j - v_k\rvert^2 \geq \lvert v_i - v_k\rvert^2 \quad \forall\, i,j,k \in V$	$X_{i,j} + X_{j,k} - X_{i,k} - X_{j,j} \leq 0 \quad \forall\, i,j,k \in V$

B. Omitted Proofs

B.1. Preliminaries

We begin by introducing some new notation used throughout the proofs and state some simple technical observations:

1. **Notation**
   - $L_A := \sup_\xi \|A(\xi)\|^2$;
   - $f(x) := \mathbb{E}\left[f(x,\xi)\right]$.
   - From above it follows that Assumption 4.2 can be rewritten as: $\mathbb{E}\left[\nabla f(x,\xi)\right] = \nabla f(x)$ and $\mathbb{E}\left[\|\nabla f(x,\xi) - \nabla f(x)\|^2\right] \leq \sigma_f^2 < +\infty$;
   - $g(A(\xi)x) := \delta_{\{b(\xi)\}}(A(\xi)x)$;
   - $g_\beta(A(\xi)x) := \frac{1}{2\beta}\mathrm{dist}(A(\xi)x, b(\xi))^2 = \frac{1}{2\beta}\|A(\xi)x - \Pi_{b(\xi)}(A(\xi)x)\|^2$, where $\Pi_{b(\xi)}(A(\xi)x)) = \underset{y\in b(\xi)}{\arg\min}\|A(\xi)x - y\|^2$; also, g_β is $\frac{1}{\beta}$-smooth
   - $G_\beta(Ax) := \mathbb{E}\left[g_\beta(A(\xi)x)\right]$, $\nabla G_\beta(Ax) := \mathbb{E}\left[\nabla g_\beta(A(\xi)x)\right]$, where $A : \mathbb{R}^d \to \mathcal{H}$ is a linear operator such that $(Ax)\xi = A(\xi)x$ and $G_\beta : \mathcal{H} \to \mathbb{R}\cup\{\infty\}$.
   - $F_{\beta_k}(x,\xi) := f(x,\xi) + g_{\beta_k}(x,\xi)$, $\nabla F_{\beta_k}(x,\xi) := \nabla f(x,\xi) + \nabla g_{\beta_k}(x,\xi)$
   - We annotate averaged stochastic quantities with the symbol $\sim$. For example, the averaged stochastic gradient of the constraints is expressed as $\tilde{\nabla}_x g_\beta(A(\xi)x)$;
   - The optimal value of the dual problem at $A(\xi)x$ is denoted as $\lambda_\beta^*(A(\xi)x) := \frac{1}{\beta}(A(\xi)x - \Pi_{b(\xi)}(A(\xi)x))$;
   - The smoothed gap is defined as $S_\beta(x) := F_\beta(x) - f(x^*)$.
2. **Technical observations**

a. From the definition of G_β:

$$\begin{aligned}\nabla_x G_\beta(Ax) &= \mathbb{E}\left[\nabla_x g_\beta(A(\xi)x)\right] \\ &= \mathbb{E}\left[A^T(\xi)\nabla g_\beta(A(\xi)x)\right] \\ &= \mathbb{E}\left[\frac{1}{\beta}A^T(\xi)\left(A(\xi)x - \Pi_{b(\xi)}(A(\xi)x)\right)\right];\end{aligned}$$

b. Form smoothness of G_β, iterate update rule and non-expansiveness of projections:

$$\begin{aligned}&\|\nabla G_\beta(Ax_{k+1}) - \nabla G_\beta(Ax_k)\|^2 \\ &\quad= \left\|\frac{1}{\beta}\mathbb{E}\left[A^T(\xi)\left(A(\xi)x_{k+1} - \Pi_{b(\xi)}(A(\xi)x_{k+1})\right) - A^T(\xi)\left(A(\xi)x_k - \Pi_{b(\xi)}(A(\xi)x_k)\right)\right]\right\|^2 \\ &\quad\le \frac{1}{\beta^2}\mathbb{E}\left[\left\|A^T(\xi)A(\xi)\left(x_{k+1} - x_k\right) + A^T(\xi)\left(\Pi_{b(\xi)}(A(\xi)x_k) - \Pi_{b(\xi)}(A(\xi)x_{k+1})\right)\right\|^2\right] \\ &\quad\le \frac{1}{\beta^2}\mathbb{E}\left[2\left\|A^T(\xi)A(\xi)\left(x_{k+1} - x_k\right)\right\|^2 + 2\left\|A^T(\xi)\left(\Pi_{b(\xi)}(A(\xi)x_k) - \Pi_{b(\xi)}(A(\xi)x_{k+1})\right)\right\|^2\right] \\ &\quad\le \frac{2\gamma_k^2 L_A^2 \mathcal{D}_{\mathcal{X}}^2}{\beta^2} + \frac{2}{\beta^2}\mathbb{E}\left[\|A(\xi)\|^2\left\|\Pi_{b(\xi)}(A(\xi)x_k) - \Pi_{b(\xi)}(A(\xi)x_{k+1})\right\|^2\right] \\ &\quad\le \frac{2\gamma_k^2 L_A^2 \mathcal{D}_{\mathcal{X}}^2}{\beta^2} + \frac{2}{\beta^2}\mathbb{E}\left[\|A(\xi)\|^2\|A(\xi)x_k - A(\xi)x_{k+1}\|^2\right] \\ &\quad\le \frac{4\gamma_k^2 L_A^2 \mathcal{D}_{\mathcal{X}}^2}{\beta^2};\end{aligned}$$

c. Variance of $g_\beta(A(\xi)x, \xi)$:

$$\begin{aligned}\mathbb{E}\left[\|\nabla g_\beta(A(\xi)x,\xi) - \nabla G_\beta(Ax)\|^2\right] &= \mathbb{E}\left[\|\nabla g_\beta(A(\xi)x,\xi)\|^2 - \|\nabla G_\beta(Ax)\|^2\right] \\ &\le \mathbb{E}\left[\|\nabla g_\beta(A(\xi)x,\xi)\|^2\right] \\ &\le \frac{1}{\beta^2}\mathbb{E}\left[\|A(\xi)\|^2\left\|A(\xi)x - \Pi_{b(\xi)}(A(\xi)x)\right\|^2\right] \\ &\le \frac{1}{\beta^2}\mathbb{E}\left[\|A(\xi)\|^2\|A(\xi)x - A(\xi)x^*\|^2\right] \\ &\le \frac{L_A^2 D_{\mathcal{X}}^2}{\beta^2},\end{aligned}$$

where we used the definition of G_β and $\left\|A(\xi)x - \Pi_{b(\xi)}(A(\xi)x)\right\|^2 \le \|A(\xi)x - A(\xi)x^*\|^2$

d. Smoothness constant of $g_\beta(A(\xi)x)$ and $F_\beta(x,\xi)$:

$$\begin{aligned}\|\nabla g_\beta(A(\xi)x) - \nabla g_\beta(A(\xi)y)\| &= \|\frac{A^T(\xi)}{2\beta}\left(A(\xi)x - \Pi_{b(\xi)}(A(\xi)x)\right) - \frac{A^T(\xi)}{2\beta}\left(A(\xi)y - \Pi_{b(\xi)}(A(\xi)y)\right)\| \\ &\le \frac{L_A}{2\beta}\|x-y\| + \frac{\|A(\xi)\|}{2\beta}\|\Pi_{b(\xi)}(A(\xi)y) - \Pi_{b(\xi)}(A(\xi)x)\| \\ &\le \frac{L_A}{2\beta}\|x-y\| + \frac{\|A(\xi)\|}{2\beta}\|A(\xi)y - A(\xi)x\| \\ &\le \frac{L_A}{\beta}\|x-y\|\end{aligned}$$

This implies that $F_\beta(x,\xi)$ is $(L_f + \frac{L_A}{\beta})$-smooth.

e. Properties of g_β (results from **Lemma 10** in (Tran-Dinh et al., 2018)):

i. $g(z_1) \geq g_\beta(z_2) + \langle \nabla g_\beta(z_2),\ z_1 - z_2 \rangle + \frac{\beta}{2}\|\lambda^*_\beta(z_2)\|^2$

ii. $g_{\beta_k}(A(\xi)x_k) \leq g_{\beta_{k-1}}(A(\xi)x_k) + \frac{\beta_{k-1}-\beta_k}{2}\|\lambda^*_{\beta_k}(A(\xi)x_k)\|^2$

Secondly, we restate **Lemma 3.1** from (Fercoq et al., 2019) for completeness, as we rely on it for translating the convergence rates from the smoothed gap onto objective suboptimality and feasibility.

Lemma B.1 (Restatement of **Lemma 3.1** from (Fercoq et al., 2019))**.**
Let (x^*, λ^*) *be a saddle point of* $\mathcal{L}(x,\lambda) := f(x) + \int \langle A(\xi)x, \lambda(\xi)\rangle - \mathrm{supp}_{b(\xi)}(\lambda(\xi))\mu(d\xi)$, *where* $\mathrm{supp}_{\mathcal{X}}(x) := \sup_{y\in\mathcal{X}}\langle y,\ x\rangle$. *Then the following holds:*

1. $S_\beta(x) \geq -\frac{\beta}{2}\|\lambda^*\|^2$
2. $F(x) - F(x^*) \geq -\frac{1}{4\beta}\int dist(A(\xi)x, b(\xi))^2 dP(\xi) - \beta\|\lambda^*\|^2$
3. $F(x) - F(x^*) \leq S_\beta(x)$
4. $\int dist(A(\xi)x, \chi(\xi))^2 dP(\xi) \leq 4\beta^2\|\lambda^*\|^2 + 4\beta S_\beta(x)$

Finally, we adapt **Lemma 17** in (Mokhtari et al., 2018) for use in our convergence proofs, and provide the proof below.

Lemma B.2 (Adaptation of **Lemma 17** in (Mokhtari et al., 2018))**.**
Let $0 < \alpha \leq 1$, $1 \leq \beta \leq 2$, $b \geq 0$, $c > 1$, $t_0 \geq 0$. *Let* ϕ_k *be a sequence of real numbers satisfying*

$$\phi_k \leq (1 - \frac{c}{(k+k_0)^\alpha})\phi_{k-1} + \frac{b}{(k+k_0)^\beta}. \tag{9}$$

Then, the sequence ϕ_k *converges to zero at the rate*

$$\phi_k \leq \frac{Q}{(k+1+k_0)^{\beta-\alpha}}, \tag{10}$$

when $\alpha = 1$, $1 < \beta \leq 2$, *or* $\alpha = \frac{2}{3}$, $\beta = 1$, *where* $Q = \max(\phi_0(k_0+1)^{\beta-\alpha}, b/(c-1))$.

Proof We use induction. By the definition of Q, $\phi_0 \leq Q/((k_0+1)^{\beta-\alpha})$, so the base step holds. Now assume it holds for k and check for $k+1$. To ease the notation let $y = k+1+k_0$. When $\alpha = 1$,

$$\phi_{k+1} \leq \left(1 - \frac{c}{y}\right)\frac{Q}{y^{\beta-1}} + \frac{b}{y^\beta} = \left(1 - \frac{c}{y}\right)\frac{Q}{y^{\beta-1}} + \frac{(c-1)Q}{y^\beta} = \frac{Q}{y^{\beta-1}} - \frac{Q}{y^\beta} \leq \frac{Q}{(y+1)^{\beta-1}},$$

where the last step follows since $1 \leq \beta \leq 2$, i.e. $\frac{y-1}{y^\beta} \leq \frac{1}{(y+1)^{\beta-1}} \iff \frac{(y-1)(y+1)^\beta}{(y+1)y^\beta} \leq 1$ and $\frac{(y-1)(y+1)^\beta}{(y+1)y^\beta} \leq \frac{(y-1)(y+1)^2}{(y+1)y^2} \leq 1$, since $\beta \leq 2$.

For general α, β, we get $\frac{1}{y^{\beta-\alpha}} - \frac{1}{y^\beta} \leq \frac{1}{(y+1)^{\beta-\alpha}} \iff \frac{y^\alpha - 1}{y^\beta} \leq \frac{(y+1)^\alpha}{(y+1)^\beta}$. If $\alpha = 2/3, \beta = 1$, then $\frac{y^{2/3}-1}{y} \leq \frac{(y+1)^{2/3}}{(y+1)} \iff \frac{(y^{2/3}-1)(y+1)^{1/3}}{y} \leq 1 \iff \frac{(y^{2/3}-1)^3(y+1)}{y^3} \leq 1 \iff \frac{(y^2-3y^{4/3}+3y^{2/3}-1)(y+1)}{y^3} \leq 1 \iff \frac{(y^3+y^2-3y^{7/3}-3y^{4/3}+3y^{5/3}+3y^{2/3}-y-1}{y^3} \leq 1$ which holds for $y \geq 1$. □

B.2. Analysis of H-1SFW

This section provides the omitted proofs of Section 4.3.2 in the main text. We start with a supporting lemma, needed for the proof of Lemma 4.1.

Lemma B.3. *Let* $d_k = (1-\rho_k)d_{k-1} + \rho_k \nabla F_{\beta_k}(x_k, \xi_k)$, $\rho_k \in [0,1]$. *Then, for all* k,

$$\mathbb{E}_k\left[\|\nabla F_{\beta_k}(x_k) - d_k\|^2\right] \leq (1-\frac{\rho_k}{2})\|\nabla F_{\beta_{k-1}}(x_{k-1}) - d_{k-1}\|^2 + 2\rho_k^2\left(\sigma_f^2 + \frac{L_A^2 D_{\mathcal{X}}^2}{\beta_k^2}\right) + \frac{2}{\rho_k}\left[2L_f^2\gamma_{k-1}^2\mathcal{D}_{\mathcal{X}}^2 + 2L_A^2\mathcal{D}_{\mathcal{X}}^2\left[\left(\frac{1}{\beta_k} - \frac{1}{\beta_{k-1}}\right)^2 + \frac{4\gamma_{k-1}}{\beta_{k-1}}\left|\frac{1}{\beta_k} - \frac{1}{\beta_{k-1}}\right| + \frac{4\gamma_{k-1}^2}{\beta_{k-1}^2}\right]\right], \tag{11}$$

where $\mathbb{E}_k[\cdot] = \mathbb{E}[\cdot|\mathcal{F}_k]$ *and* $\mathcal{F}_k$ *is a* σ*-algebra measuring all sources of randomness up to step* k.

Proof We use the definition $d_k = (1-\rho_k)d_{k-1} + \rho_k \nabla F_{\beta_k}(x_k, \xi_k)$ to write the difference

$$\begin{aligned}
&\|\nabla F_{\beta_k}(x_k) - d_k\|^2 \\
&\quad= \|\nabla F_{\beta_k}(x_k) - (1-\rho_k)d_{k-1} - \rho_k \nabla F_{\beta_k}(x_k, \xi_k)\|^2 \\
&\quad= \|\nabla F_{\beta_k}(x_k) + (1-\rho_k)\nabla F_{\beta_{k-1}}(x_{k-1}) - (1-\rho_k)\nabla F_{\beta_{k-1}}(x_{k-1}) - (1-\rho_k)d_{k-1} - \rho_k \nabla F_{\beta_k}(x_k, \xi_k)\|^2 \\
&\quad= \|\rho_k(\nabla F_{\beta_k}(x_k) - \nabla F_{\beta_k}(x_k, \xi_k)) + (1-\rho_k)(\nabla F_{\beta_k}(x_k) - \nabla F_{\beta_{k-1}}(x_{k-1})) \\
&\quad+ (1-\rho_k)(\nabla F_{\beta_{k-1}}(x_{k-1}) - d_{k-1})\|^2 \\
&\quad= \rho_k^2\|\nabla F_{\beta_k}(x_k) - \nabla F_{\beta_k}(x_k, \xi_k)\|^2 + (1-\rho_k)^2\|\nabla F_{\beta_k}(x_k) - \nabla F_{\beta_{k-1}}(x_{k-1})\|^2 \\
&\quad+ (1-\rho_k)^2\|\nabla F_{\beta_{k-1}}(x_{k-1}) - d_{k-1}\|^2 \\
&\quad+ 2\rho_k(1-\rho_k)\langle \nabla F_{\beta_k}(x_k) - \nabla F_{\beta_k}(x_k, \xi_k), \nabla F_{\beta_k}(x_k) - \nabla F_{\beta_{k-1}}(x_{k-1})\rangle \\
&\quad+ 2\rho_k(1-\rho_k)\langle \nabla F_{\beta_k}(x_k) - \nabla F_{\beta_k}(x_k, \xi_k), \nabla F_{\beta_{k-1}}(x_{k-1}) - d_{k-1}\rangle \\
&\quad+ 2(1-\rho_k)^2\langle \nabla F_{\beta_k}(x_k) - \nabla F_{\beta_{k-1}}(x_{k-1}), \nabla F_{\beta_{k-1}}(x_{k-1}) - d_{k-1}\rangle
\end{aligned}$$

We remark that $\mathbb{E}_k\left[\nabla F_{\beta_k}(x_k, \xi_k)\right] = \nabla F_{\beta_k}(x_k)$ so that first two linear terms are 0. We now take expectations conditioned on $\mathcal{F}_k$,

$$\begin{aligned}
&\mathbb{E}_k\left[\|\nabla F_{\beta_k}(x_k) - d_k\|^2\right] \\
&\quad= \rho_k^2\mathbb{E}_k\left[\|\nabla F_{\beta_k}(x_k) - \nabla F_{\beta_k}(x_k, \xi_k)\|^2\right] + (1-\rho_k)^2\|\nabla F_{\beta_k}(x_k) - \nabla F_{\beta_{k-1}}(x_{k-1})\|^2 \\
&\quad+ (1-\rho_k)^2\|\nabla F_{\beta_{k-1}}(x_{k-1}) - d_{k-1}\|^2 + 2(1-\rho_k)^2\langle \nabla F_{\beta_k}(x_k) - \nabla F_{\beta_{k-1}}(x_{k-1}), \nabla F_{\beta_{k-1}}(x_{k-1}) - d_{k-1}\rangle
\end{aligned} \tag{12}$$

Invoking the variance bound from Technical Observation c. from Section B.1, we have:

$$\mathbb{E}_k\left[\|\nabla F_{\beta_k}(x_k) - \nabla F_{\beta_k}(x_k, \xi_k)\|^2\right] \leq 2\mathbb{E}_k\left[\|\nabla f(x_k) - \nabla f(x_k, \xi_k)\|^2\right] + 2\mathbb{E}_k\left[\|G_{\beta_k}(Ax_k) - \nabla g_{\beta_k}(A(\xi)x_k, \xi_k)\|^2\right] \tag{13}$$

$$\leq 2\left(\sigma_f^2 + \frac{L_A^2 D_{\mathcal{X}}^2}{\beta_k^2}\right) \tag{14}$$

For the linear term, we use Young's inequality for some $\sigma_k > 0$ to get

$$2(1-\rho_k)^2\langle \nabla F_{\beta_k}(x_k) - \nabla F_{\beta_{k-1}}(x_{k-1}), \nabla F_{\beta_{k-1}}(x_{k-1}) - d_{k-1}\rangle \\ \leq (1-\rho_k)^2\sigma_k\|\nabla F_{\beta_{k-1}}(x_{k-1}) - d_{k-1}\|^2 + (1-\rho_k)^2(1/\sigma_k)\|\nabla F_{\beta_k}(x_k) - \nabla F_{\beta_{k-1}}(x_{k-1})\|^2 \quad (15)$$

For the $\|\nabla F_{\beta_k}(x_k) - \nabla F_{\beta_{k-1}}(x_{k-1})\|^2$ term, we use the iterate update rule and Technical Observation b. to get:

$$\begin{aligned}
&\|\nabla F_{\beta_k}(x_k) - \nabla F_{\beta_{k-1}}(x_{k-1})\|^2 \\
&\quad = \|\nabla f(x_k) - \nabla f(x_{k-1}) + \nabla G_{\beta_k}(Ax_k) - \nabla G_{\beta_{k-1}}(Ax_{k-1})\|^2 \\
&\quad \leq 2\|\nabla f(x_k) - \nabla f(x_{k-1})\|^2 + 2\|\nabla G_{\beta_k}(Ax_k) - \nabla G_{\beta_{k-1}}(Ax_k) + \nabla G_{\beta_{k-1}}(Ax_k) - \nabla G_{\beta_{k-1}}(Ax_{k-1})\|^2 \\
&\quad \leq 2L_f^2\|x_k - x_{k-1}\|^2 + 2\|\nabla G_{\beta_k}(Ax_k) - \nabla G_{\beta_{k-1}}(Ax_k)\|^2 + 2\|\nabla G_{\beta_{k-1}}(Ax_k) - \nabla G_{\beta_{k-1}}(Ax_{k-1})\|^2 \\
&\qquad\qquad + 4\|\nabla G_{\beta_k}(Ax_k) - \nabla G_{\beta_{k-1}}(Ax_k)\|\|\nabla G_{\beta_{k-1}}(Ax_k) - \nabla G_{\beta_{k-1}}(Ax_{k-1})\| \\
&\quad \leq 2L_f^2\gamma_{k-1}^2\mathcal{D}_{\mathcal{X}}^2 + 2L_A^2\mathcal{D}_{\mathcal{X}}^2\left(\frac{1}{\beta_k} - \frac{1}{\beta_{k-1}}\right)^2 + \frac{8\gamma_{k-1}^2L_A^2\mathcal{D}_{\mathcal{X}}^2}{\beta_{k-1}^2} + 8\gamma_{k-1}\left|\frac{1}{\beta_k} - \frac{1}{\beta_{k-1}}\right|\frac{L_A^2\mathcal{D}_{\mathcal{X}}^2}{\beta_{k-1}} \qquad (16)
\end{aligned}$$

Putting everything back into (12):

$$\begin{aligned}
&\mathbb{E}_k\left[\|\nabla F_{\beta_k}(x_k) - d_k\|^2\right] \\
&\quad = \rho_k^2\mathbb{E}_k\left[\|\nabla F_{\beta_k}(x_k) - \nabla F_{\beta_k}(x_k,\xi_k)\|^2\right] + (1-\rho_k)^2(1+\sigma_k^{-1})\|\nabla F_{\beta_k}(x_k) - \nabla F_{\beta_{k-1}}(x_{k-1})\|^2 \\
&\qquad\qquad + (1-\rho_k)^2(1+\sigma_k)\|\nabla F_{\beta_{k-1}}(x_{k-1}) - d_{k-1}\|^2 \\
&\quad \leq (1-\rho_k)^2(1+\sigma_k)\|\nabla F_{\beta_{k-1}}(x_{k-1}) - d_{k-1}\|^2 + 2\rho_k^2\left(\sigma_f^2 + \frac{L_A^2\mathcal{D}_{\mathcal{X}}^2}{\beta_k^2}\right) \\
&\qquad + (1-\rho_k)^2(1+\sigma_k^{-1})\left[2L_f^2\gamma_{k-1}^2\mathcal{D}_{\mathcal{X}}^2 + 2L_A^2\mathcal{D}_{\mathcal{X}}^2\left[\left(\frac{1}{\beta_k} - \frac{1}{\beta_{k-1}}\right)^2 + \frac{4\gamma_{k-1}}{\beta_{k-1}}\left|\frac{1}{\beta_k} - \frac{1}{\beta_{k-1}}\right| + \frac{4\gamma_{k-1}^2}{\beta_{k-1}^2}\right]\right]. \qquad (17)
\end{aligned}$$

Using the facts that $\rho_k \leq 1$, $(1-\rho_k)^2 \leq (1-\rho_k)$, $(1-\rho_k)(1+\frac{\rho_k}{2}) \leq (1-\rho_k/2)$, $(1-\rho_k)(1+\frac{2}{\rho_k}) \leq \frac{2}{\rho_k}$ and setting $\sigma_k := \frac{\rho_k}{2}$, we get:

$$\begin{aligned}
\mathbb{E}_k\left[\|\nabla F_{\beta_k}(x_k) - d_k\|^2\right] &\leq (1-\frac{\rho_k}{2})\|\nabla F_{\beta_{k-1}}(x_{k-1}) - d_{k-1}\|^2 + 2\rho_k^2\left(\sigma_f^2 + \frac{L_A^2D_{\mathcal{X}}^2}{\beta_k^2}\right) \\
&\quad + \frac{2}{\rho_k}\left[2L_f^2\gamma_{k-1}^2\mathcal{D}_{\mathcal{X}}^2 + 2L_A^2\mathcal{D}_{\mathcal{X}}^2\left[\left(\frac{1}{\beta_k} - \frac{1}{\beta_{k-1}}\right)^2 + \frac{4\gamma_{k-1}}{\beta_{k-1}}\left|\frac{1}{\beta_k} - \frac{1}{\beta_{k-1}}\right| + \frac{4\gamma_{k-1}^2}{\beta_{k-1}^2}\right]\right] \quad \square
\end{aligned}$$

Lemma 4.1. *Let* $\rho_k = \frac{3}{(k+5)^{2/3}}$, $\gamma_k = \frac{2}{k+1}$, $\beta_k = \frac{\beta_0}{(k+1)^{1/6}}$, $\beta_0 > 0$ *in Algorithm 1. Then, for all* k,

$$\mathbb{E}\left[\|\nabla F_{\beta_k}(x_k) - d_k\|^2\right] \leq \frac{C_1}{(k+5)^{1/3}}, \qquad (18)$$

where $C_1 = \max\left(6^{1/3}\|\nabla F_{\beta_0}(x_0) - d_0\|^2, 2\left[18\sigma_f^2 + 112L_f^2\mathcal{D}_{\mathcal{X}}^2 + \frac{522L_A^2\mathcal{D}_{\mathcal{X}}^2}{\beta_0^2}\right]\right)$.

Proof

We apply the expectation with respect to the whole history to (11) and estimate the rate of $\left|\frac{1}{\beta_k}-\frac{1}{\beta_{k-1}}\right|$:

$$\mathbb{E}\left[\|\nabla F_{\beta_k}(x_k)-d_k\|^2\right] \leq (1-\frac{\rho_k}{2})\mathbb{E}\left[\|\nabla F_{\beta_{k-1}}(x_{k-1})-d_{k-1}\|^2\right] + 2\rho_k^2\left(\sigma_f^2+\frac{L_A^2\mathcal{D}_{\mathcal{X}}^2}{\beta_k^2}\right) + \frac{2}{\rho_k}\left[2L_f^2\gamma_{k-1}^2\mathcal{D}_{\mathcal{X}}^2 + 2L_A^2\mathcal{D}_{\mathcal{X}}^2\left[\left(\frac{1}{\beta_k}-\frac{1}{\beta_{k-1}}\right)^2+\frac{4\gamma_{k-1}}{\beta_{k-1}}\left|\frac{1}{\beta_k}-\frac{1}{\beta_{k-1}}\right|+\frac{4\gamma_{k-1}^2}{\beta_{k-1}^2}\right]\right]$$

$$\begin{aligned}0\leq \frac{1}{\beta_k}-\frac{1}{\beta_{k-1}} &= \frac{(k+1)^{1/6}-(k)^{1/6}}{\beta_0}\\ &= \frac{1}{\beta_0\left[(k+1)^{5/6}+(k+1)^{4/6}k^{1/6}+(k+1)^{3/6}k^{2/6}+(k+1)^{2/6}k^{3/6}+(k+1)^{1/6}k^{4/6}+k^{5/6}\right]}\\ &\leq \frac{1}{6\beta_0 k^{5/6}}\end{aligned}$$

Replacing the parameter rates we further get:

$$\begin{aligned}&\mathbb{E}\left[\|\nabla F_{\beta_k}(x_k)-d_k\|^2\right]\\ &\leq \left(1-\frac{3}{2(k+5)^{2/3}}\right)\mathbb{E}\left[\|\nabla F_{\beta_{k-1}}(x_{k-1})-d_{k-1}\|^2\right]+\frac{18}{(k+5)^{4/3}}\left(\sigma_f^2+\frac{L_A^2\mathcal{D}_{\mathcal{X}}^2(k+1)^{2/6}}{\beta_0^2}\right)\\ &\qquad+\frac{2(k+5)^{2/3}}{3}\left[\frac{8L_f^2\mathcal{D}_{\mathcal{X}}^2}{k^2}+\frac{2L_A^2\mathcal{D}_{\mathcal{X}}^2}{\beta_0^2}\left(\frac{1}{36k^{10/6}}+\frac{4}{3k^{10/6}}+\frac{16}{k^{10/6}}\right)\right]\\ &\leq \left(1-\frac{3}{2(k+5)^{2/3}}\right)\mathbb{E}\left[\|\nabla F_{\beta_{k-1}}(x_{k-1})-d_{k-1}\|^2\right]+\frac{18\sigma_f^2}{k+5}+\frac{18L_A^2\mathcal{D}_{\mathcal{X}}^2}{\beta_0^2(k+5)}+\frac{2(k+5)^{2/3}}{3k^{10/6}}\left(8L_f^2\mathcal{D}_{\mathcal{X}}^2+\frac{36L_A^2\mathcal{D}_{\mathcal{X}}^2}{\beta_0^2}\right)\\ &\leq \left(1-\frac{3}{2(k+5)^{2/3}}\right)\mathbb{E}\left[\|\nabla F_{\beta_{k-1}}(x_{k-1})-d_{k-1}\|^2\right]+\frac{18\sigma_f^2}{k+5}+\frac{18L_A^2\mathcal{D}_{\mathcal{X}}^2}{\beta_0^2(k+5)}+\frac{14}{k+5}\left(8L_f^2\mathcal{D}_{\mathcal{X}}^2+\frac{36L_A^2\mathcal{D}_{\mathcal{X}}^2}{\beta_0^2}\right) \qquad (19)\\ &= \left(1-\frac{3}{2(k+5)^{2/3}}\right)\mathbb{E}\left[\|\nabla F_{\beta_{k-1}}(x_{k-1})-d_{k-1}\|^2\right]+\frac{1}{k+5}\left(18\sigma_f^2+112L_f^2\mathcal{D}_{\mathcal{X}}^2+\frac{522L_A^2\mathcal{D}_{\mathcal{X}}^2}{\beta_0^2}\right)\end{aligned}$$

where line (19) follows form the fact that

$$\frac{(k+5)^{2/3}}{k^{10/6}}=\frac{(k+5)^{4/6}}{k^{10/6}}\frac{(k+5)^{6/6}}{(k+5)^{6/6}}=\left(1+\frac{5}{k}\right)^{4/6+6/6}\frac{1}{(k+5)^{6/6}}=\left(1+\frac{5}{k}\right)^{5/3}\frac{1}{k+5}<\frac{6^{5/3}}{k+5}<\frac{21}{k+5}$$

We can now invoke Lemma B.3 for $b=18\sigma_f^2+112L_f^2\mathcal{D}_{\mathcal{X}}^2+\frac{522L_A^2\mathcal{D}_{\mathcal{X}}^2}{\beta_0^2}$ and $c=\frac{3}{2}$, $\alpha=\frac{2}{3}$ and $\beta=1$, $k_0=5$ to conclude the result. □

Theorem 4.1. *Consider Algorithm 1 with parameters* $\rho_k=\frac{3}{(k+5)^{2/3}}$, $\gamma_k=\frac{2}{k+1}$, $\beta_k=\frac{\beta_0}{(k+1)^{1/6}}$, $\beta_0>0$ *(the same as Lemma 4.1). Then, for all k,*

$$\mathbb{E}\left[S_{\beta_k}(x_{k+1})\right]\leq\frac{C_2}{k^{1/6}}, \qquad (20)$$

where $C_2 = \max\left\{S_0(x_1),\ b = 2\mathcal{D}_{\mathcal{X}}\sqrt{C_1} + 2\mathcal{D}_{\mathcal{X}}^2\left(L_f + \frac{L_A}{\beta_0}\right)\right\}$*, and* C_1 *is defined in Lemma 4.1.*

Proof

We essentially follow the steps for proving **Theorem 9** of (Locatello et al., 2019), modified to suit our setting. Using Technical Observation d. and the definition of $\mathcal{D}_{\mathcal{X}}$:

$$\begin{aligned}
F_{\beta_k}(x_{k+1}) &= \mathbb{E}_{k+1}\left[F_{\beta_k}(x_{k+1},\xi)\right] \\
&\leq \mathbb{E}_{k+1}\left[F_{\beta_k}(x_k,\xi) + \langle \nabla F_{\beta_k}(x_k,\xi), x_{k+1}-x_k\rangle + \frac{1}{2}\left(L_f + \frac{L_A}{\beta_k}\right)\|x_{k+1}-x_k\|^2\right] \\
&\leq F_{\beta_k}(x_k) + \gamma_k\langle \nabla F_{\beta_k}(x_k), w_k - x_k\rangle + \frac{\gamma_k^2}{2}\left(L_f + \frac{L_A}{\beta_k}\right)\mathcal{D}_{\mathcal{X}}^2
\end{aligned} \tag{21}$$

We treat the term $\langle \nabla F_{\beta_k}(x_k), w_k - x_k\rangle$ separately, using the fact that $w_k \in \arg\min_x \langle d_k, y\rangle$ and the definition of $\mathcal{D}_{\mathcal{X}}$:

$$\begin{aligned}
\langle \nabla F_{\beta_k}(x_k), w_k - x_k\rangle &= \langle \nabla F_{\beta_k}(x_k) - d_k, w_k - x_k\rangle + \langle d_k, w_k - x_k\rangle \\
&= \langle \nabla F_{\beta_k}(x_k) - d_k, w_k - x^*\rangle + \langle \nabla F_{\beta_k}(x_k) - d_k, x^* - x_k\rangle + \langle d_k, w_k - x_k\rangle \\
&\leq \langle \nabla F_{\beta_k}(x_k) - d_k, w_k - x^*\rangle + \langle \nabla F_{\beta_k}(x_k) - d_k, x^* - x_k\rangle + \langle d_k, x^* - x_k\rangle \\
&= \langle \nabla F_{\beta_k}(x_k) - d_k, w_k - x^*\rangle + \langle \nabla F_{\beta_k}(x_k), x^* - x_k\rangle \\
&\leq \|\nabla F_{\beta_k}(x_k) - d_k\|\|w_k - x^*\| + \langle \nabla F_{\beta_k}(x_k), x^* - x_k\rangle \\
&\leq \|\nabla F_{\beta_k}(x_k) - d_k\|\mathcal{D}_{\mathcal{X}} + \langle \nabla F_{\beta_k}(x_k), x^* - x_k\rangle \\
&= \|\nabla F_{\beta_k}(x_k) - d_k\|\mathcal{D}_{\mathcal{X}} + \langle \nabla f(x_k) + \nabla_x G_{\beta_k}(Ax_k), x^* - x_k\rangle
\end{aligned} \tag{22}$$

Using Technical Observation 2(e.)i we observe that

$$\begin{aligned}
\langle \nabla_x G_{\beta_k}(Ax_k), x^* - x_k\rangle &= \mathbb{E}_k\left[\langle \nabla_x g_{\beta_k}(A(\xi)x_k), x^* - x_k\rangle\right] \\
&= \mathbb{E}_k\left[\langle \nabla g_{\beta_k}(A(\xi)x_k), A(\xi)x^* - A(\xi)x_k\rangle\right] \\
&\leq \mathbb{E}_k\left[g(A(\xi)x^*) - g_{\beta_k}(A(\xi)x_k) - \frac{\beta_k}{2}\|\lambda^*_{\beta_k}(A(\xi)x_k)\|^2\right] \\
&= G(Ax^*) - G_{\beta_k}(Ax_k) - \frac{\beta_k}{2}\mathbb{E}_k\left[\|\lambda^*_{\beta_k}(A(\xi)x_k)\|^2\right]
\end{aligned}$$

Using the above and the convexity of f, we obtain:

$$\langle \nabla F_{\beta_k}(x_k), w_k - x_k\rangle \leq \|\nabla F_{\beta_k}(x_k) - d_k\|\mathcal{D}_{\mathcal{X}} + f(x^*) + G(Ax^*) \underbrace{-f(x_k) - G_{\beta_k}(Ax_k)}_{=-F_{\beta_k}(x_k)} - \frac{\beta_k}{2}\mathbb{E}_k\left[\|\lambda^*_{\beta_k}(A(\xi)x_k)\|^2\right]$$

Substituting everything back into Equation (21) and noting that $G(Ax^*) = 0$:

$$F_{\beta_k}(x_{k+1}) \leq (1-\gamma_k)F_{\beta_k}(x_k) + \gamma_k\|\nabla F_{\beta_k}(x_k) - d_k\|\mathcal{D}_{\mathcal{X}} + \gamma_k f(x^*) - \frac{\gamma_k\beta_k}{2}\mathbb{E}_k\left[\|\lambda^*_{\beta_k}(A(\xi)x_k)\|^2\right] + \frac{\gamma_k^2}{2}\left(L_f + \frac{L_A}{\beta_k}\right)\mathcal{D}_{\mathcal{X}}^2.$$

Using Technical Observation 2(e.)ii we observe that

$$F_{\beta_k}(x_k) = \mathbb{E}_k\left[f(x_k,\xi) + g_{\beta_k}(A(\xi)x_k)\right]$$

$$\leq \mathbb{E}_k\left[f(x_k,\xi) + g_{\beta_{k-1}}(A(\xi)x_k) + \frac{\beta_{k-1}-\beta_k}{2}\|\lambda^*_{\beta_k}(A(\xi)x_k)\|^2\right]$$
$$= F_{\beta_{k-1}}(x_k) + \mathbb{E}_k\left[\frac{\beta_{k-1}-\beta_k}{2}\|\lambda^*_{\beta_k}(A(\xi)x_k)\|^2\right]$$

Substituting the above, we obtain:

$$\begin{aligned} F_{\beta_k}(x_{k+1}) &\leq (1-\gamma_k)F_{\beta_{k-1}}(x_k) + \gamma_k\|\nabla F_{\beta_k}(x_k) - d_k\|\mathcal{D}_{\mathcal{X}} + \gamma_k f(x^*) \\ &\quad + \frac{(1-\gamma_k)(\beta_{k-1}-\beta_k)-\gamma_k\beta_k}{2}\mathbb{E}_k\left[\|\lambda^*_{\beta_k}(A(\xi)x_k)\|^2\right] + \frac{\gamma_k^2}{2}\left(L_f + \frac{L_A}{\beta_k}\right)\mathcal{D}_{\mathcal{X}}^2 \\ &\leq (1-\gamma_k)F_{\beta_{k-1}}(x_k) + \gamma_k\|\nabla F_{\beta_k}(x_k) - d_k\|\mathcal{D}_{\mathcal{X}} + \gamma_k f(x^*) + \frac{\gamma_k^2}{2}\left(L_f + \frac{L_A}{\beta_k}\right)\mathcal{D}_{\mathcal{X}}^2, \end{aligned} \tag{23}$$

where the last line comes from the fact that $(1-\gamma_k)(\beta_{k-1}-\beta_k)-\gamma_k\beta_k < 0$:

$$\begin{aligned} (1-\gamma_k)(\beta_{k-1}-\beta_k)-\gamma_k\beta_k = \beta_{k-1}-\beta_k-\gamma_k\beta_{k-1} &= \frac{\beta_0}{k^{1/6}} - \frac{\beta_0}{(k+1)^{1/6}} - \frac{2\beta_0}{(k+1)k^{1/6}} \\ &= \frac{\beta_0}{k^{1/6}}\left(1 - \frac{k^{1/6}}{(k+1)^{1/6}} - \frac{2}{k+1}\right) \\ &= \frac{\beta_0}{k^{1/6}}\left(\frac{k-1}{k+1} - \frac{k^{1/6}}{(k+1)^{1/6}}\right) \\ &< \frac{\beta_0}{k^{1/6}}\left(\underbrace{\frac{k}{k+1}}_{\in(0,1)} - \frac{k^{1/6}}{(k+1)^{1/6}}\right) \\ &< 0. \end{aligned}$$

Starting from Equation (23) and subtracting $f(x^*)$ from both sides, noting the definition of $S_{\beta_k}(x) := F_\beta(x) - f(x^*)$ and taking the expectation on both sides:

$$\mathbb{E}\left[S_{\beta_k}(x_{k+1})\right] \leq (1-\gamma_k)\mathbb{E}\left[S_{\beta_{k-1}}(x_k)\right] + \frac{\gamma_k^2}{2}D_{\mathcal{X}}^2\left(L_f + \frac{L_A}{\beta_k}\right) + \gamma_k\mathbb{E}\left[\|\nabla F_{\beta_k}(x_k) - d_k\|\right]\mathcal{D}_{\mathcal{X}}. \tag{24}$$

Replacing the parameter rates for the second term, we bound by

$$\begin{aligned} \frac{\gamma_k^2}{2}\mathcal{D}_{\mathcal{X}}^2(L_f + \frac{L_A}{\beta_k}) &= \frac{2\mathcal{D}_{\mathcal{X}}^2 L_f}{k^2} + \frac{2\mathcal{D}_{\mathcal{X}}^2 L_A}{\beta_0 k^{11/6}} \\ &\leq \frac{2\mathcal{D}_{\mathcal{X}}^2}{k^{7/6}}\left(L_f + \frac{L_A}{\beta_0}\right) \end{aligned}$$

For the last term we use the parameter rates and Lemma 4.1 together with Jensen's inequality $\mathbb{E}\left[\|\nabla F_{\beta_k}(x_k) - d_k\|\right] = \sqrt{\mathbb{E}\left[\|\nabla F_{\beta_k}(x_k) - d_k\|\right]^2} \leq \sqrt{\mathbb{E}\left[\|\nabla F_{\beta_k}(x_k) - d_k\|^2\right]}$ to get

$$\gamma_k\mathcal{D}_{\mathcal{X}}\mathbb{E}\left[\|\nabla F_{\beta_k}(x_k) - d_k\|\right] = \frac{2\mathcal{D}_{\mathcal{X}}}{k+1}\frac{\sqrt{C_1}}{(k+5)^{1/6}}$$

$$\leq \frac{2\mathcal{D}_{\mathcal{X}}\sqrt{C_1}}{k^{7/6}},$$

Substituting the above into (24), we get

$$\mathbb{E}\left[S_{\beta_k}(x_{k+1})\right] \leq \left(1-\frac{2}{k}\right)\mathbb{E}\left[S_{\beta_{k-1}}(x_k)\right] + \frac{2\mathcal{D}_{\mathcal{X}}\sqrt{C_1} + 2\mathcal{D}_{\mathcal{X}}^2\left(L_f + \frac{L_A}{\beta_0}\right)}{k^{7/6}}.$$

Finally, we use Lemma B.2 with $\alpha = 1$, $\beta = 7/6$, $c = 2$, $b = 2\mathcal{D}_{\mathcal{X}}\sqrt{C_1} + 2\mathcal{D}_{\mathcal{X}}^2\left(L_f + \frac{L_A}{\beta_0}\right)$ to arrive at the statement. □

Corollary 4.1. *The expected convergence in terms of objective suboptimality and feasibility is, respectively*

$$\mathbb{E}\left[|f(x_k,\xi) - f(x^*)|\right] \in \mathcal{O}\left(k^{-1/6}\right), \quad \sqrt{\mathbb{E}\left[\operatorname{dist}(A(\xi)x_k, b(\xi))^2\right]} \in \mathcal{O}\left(k^{-1/6}\right).$$

Consequently, the oracle complexity of Algorithm 1 is $\#(sfo) \in \mathcal{O}\left(\epsilon^{-6}\right)$ and $\#(lmo) \in \mathcal{O}\left(\epsilon^{-6}\right)$.

Proof The stated result comes from applying Lemma B.1 in conjunction with the convergence smoothed-gap rate obtained in Theorem 4.1. Considering that at every iteration we take one stochastic sample and compute one lmo, along with the $\mathcal{O}\left(k^{-1/6}\right)$ convergence rate, we obtain the stated oracle complexities. □

B.3. Analysis of H-SPIDER-FW

This section provides the omitted proofs of Section 4.4.2 in the main text. We start with a supporting lemma, needed for the proof of Lemma 4.2 and Lemma 4.3.

Lemma B.4. *Let $v_{t,k} = v_{t,k-1} - \tilde{\nabla}F_{\beta_{t,k-1}}(x_{t,k-1}, \xi_{\mathcal{S}_{t,k}}) + \tilde{\nabla}F_{\beta_{t,k}}(x_{t,k}, \xi_{\mathcal{S}_{t,k}})$, with $|\mathcal{S}_{t,k}| = K_t = 2^{t-1}$ and $v_{t,1} = \tilde{\nabla}F_{\beta_{t,1}}(x_{t,1}, \xi_{\mathcal{Q}_t})$. Also, let $\gamma_{t,k} = \frac{2}{K_t+k}$, $\beta_{t,k} = \frac{\beta_0}{\sqrt{K_t+k}}$. Then, for a fixed t and for all $k \leq K_t$,*

$$\mathbb{E}_{t,1}\left[\left\|\nabla F_{\beta_{t,k}}(x_{t,k}) - v_{t,k}\right\|^2\right] \leq \frac{2\mathcal{D}_{\mathcal{X}}^2}{K_t+k}\left(8L_f^2 + \frac{98L_A^2}{\beta_0^2}\right) + \mathbb{E}_{t,1}\left[\left\|\nabla F_{\beta_1}(x_1) - v_1\right\|^2\right] \tag{25}$$

Proof

$$\begin{aligned}
\left\|\nabla F_{\beta_{t,k}}(x_{t,k}) - v_{t,k}\right\|^2 &= \left\|\nabla F_{\beta_{t,k}}(x_{t,k}) - v_{t,k-1} - \tilde{\nabla}F_{\beta_{t,k}}(x_{t,k}, \xi_{\mathcal{S}_{t,k}}) + \tilde{\nabla}F_{\beta_{t,k-1}}(x_{t,k-1}, \xi_{\mathcal{S}_{t,k}})\right\|^2 \\
&= \left\|\nabla F_{\beta_{t,k}}(x_{t,k}) - \nabla F_{\beta_{t,k-1}}(x_{t,k-1}) + \nabla F_{\beta_{t,k-1}}(x_{t,k-1}) - v_{t,k-1}\right. \\
&\qquad \left. - \tilde{\nabla}F_{\beta_{t,k}}(x_{t,k}, \xi_{\mathcal{S}_{t,k}}) + \tilde{\nabla}F_{\beta_{t,k-1}}(x_{t,k-1}, \xi_{\mathcal{S}_{t,k}})\right\|^2 \\
&= \left\|\nabla F_{\beta_{t,k-1}}(x_{t,k-1}) - v_{t,k-1}\right\|^2 \\
&\quad + \left\|\nabla F_{\beta_{t,k}}(x_{t,k}) - \nabla F_{\beta_{t,k-1}}(x_{t,k-1}) - \tilde{\nabla}F_{\beta_{t,k}}(x_{t,k}, \xi_{\mathcal{S}_{t,k}}) + \tilde{\nabla}F_{\beta_{t,k-1}}(x_{t,k-1}, \xi_{\mathcal{S}_{t,k}})\right\|^2 \\
&\quad + 2\langle \nabla F_{\beta k-1}(x_{t,k-1}) - v_{t,k-1},\ \nabla F_{\beta_{t,k}}(x_{t,k}) - \nabla F_{\beta k-1}(x_{t,k-1}) \\
&\quad - \tilde{\nabla}F_{\beta_{t,k}}(x_{t,k}, \xi_{\mathcal{S}_{t,k}}) + \tilde{\nabla}F_{\beta_{t,k-1}}(x_{t,k-1}, \xi_{\mathcal{S}_{t,k}})\rangle
\end{aligned}$$

We now take the expectation on both sides $\mathbb{E}_{t,k}[X] = \mathbb{E}[X|\mathcal{F}_{t,k}]$ conditioned on all randomness up to step (t,k) (i.e. the expectations are taken solely with regards to $\xi_{\mathcal{S}_{t,k}}$).

$$\begin{aligned}
&\mathbb{E}_{t,k}\left[\|\nabla F_{\beta_{t,k}}(x_{t,k}) - v_{t,k}\|^2\right] \\
&= \|\nabla F_{\beta_{t,k-1}}(x_{t,k-1}) - v_{t,k-1}\|^2 + \mathbb{E}_{t,k}\left[\|\nabla F_{\beta_{t,k}}(x_{t,k}) - \nabla F_{\beta_{t,k-1}}(x_{t,k-1}) - \tilde{\nabla} F_{\beta_{t,k}}(x_{t,k},\xi_{\mathcal{S}_{t,k}}) + \tilde{\nabla} F_{\beta_{t,k-1}}(x_{t,k-1},\xi_{\mathcal{S}_{t,k}})\|^2\right] \\
&\quad + 2\langle \nabla F_{\beta_{t,k-1}}(x_{t,k-1}) - v_{t,k-1},\ \underbrace{\mathbb{E}_{t,k}\left[\nabla F_{\beta_{t,k}}(x_{t,k}) - \nabla F_{\beta_{t,k-1}}(x_{t,k-1}) - \tilde{\nabla} F_{\beta_{t,k}}(x_{t,k},\xi_{\mathcal{S}_{t,k}}) + \tilde{\nabla} F_{\beta_{t,k-1}}(x_{t,k-1},\xi_{\mathcal{S}_{t,k}})\right]}_{=0,\ \text{since}\ \nabla F_\beta(x) = \mathbb{E}\left[\tilde{\nabla} F(x,\xi_{\mathcal{S}_{t,k}})\right]}\rangle \\
&= \|\nabla F_{\beta_{t,k-1}}(x_{t,k-1}) - v_{t,k-1}\|^2 + \underbrace{\mathbb{E}_{t,k}\left[\|\nabla F_{\beta_{t,k}}(x_{t,k}) - \nabla F_{\beta_{t,k-1}}(x_{t,k-1}) - \tilde{\nabla} F_{\beta_{t,k}}(x_{t,k},\xi_{\mathcal{S}_{t,k}}) + \tilde{\nabla} F_{\beta_{t,k-1}}(x_{t,k-1},\xi_{\mathcal{S}_{t,k}})\|^2\right]}_{=T}
\end{aligned} \tag{26}$$

We now bound T:

$$\begin{aligned}
T &= \mathbb{E}_{t,k}\left[\|\nabla F_{\beta_{t,k}}(x_{t,k}) - \nabla F_{\beta_{t,k-1}}(x_{t,k-1}) - \tilde{\nabla} F_{\beta_{t,k}}(x_{t,k},\xi_{\mathcal{S}_{t,k}}) + \tilde{\nabla} F_{\beta_{t,k-1}}(x_{t,k-1},\xi_{\mathcal{S}_{t,k}})\|^2\right] \\
&= \mathbb{E}_{t,k}\left[\|\frac{1}{K_t}\sum_{i=1}^{K_t} \nabla F_{\beta_{t,k}}(x_{t,k}) - \nabla F_{\beta_{t,k-1}}(x_{t,k-1}) - \nabla F_{\beta_{t,k}}(x_{t,k},\xi_i) + \nabla F_{\beta_{t,k-1}}(x_{t,k-1},\xi_i)\|^2\right] \qquad (27) \\
&= \frac{1}{K_t^2}\mathbb{E}_{t,k}\left[\sum_{i=1}^{K_t}\|\nabla F_{\beta_{t,k}}(x_{t,k}) - \nabla F_{\beta_{t,k-1}}(x_{t,k-1}) - \nabla F_{\beta_{t,k}}(x_{t,k},\xi_i) + \nabla F_{\beta_{t,k-1}}(x_{t,k-1},\xi_i)\|^2\right] \\
&\quad + \frac{2}{K_t^2}\mathbb{E}_{t,k}\Big[\sum_{\substack{i,j<K_t \\ i<j}} \langle \nabla F_{\beta_{t,k}}(x_{t,k}) - \nabla F_{\beta_{t,k-1}}(x_{t,k-1}) - \nabla F_{\beta_{t,k}}(x_{t,k},\xi_i) + \nabla F_{\beta_{t,k-1}}(x_{t,k-1},\xi_i), \\
&\qquad\qquad \nabla F_{\beta_{t,k}}(x_{t,k}) - \nabla F_{\beta_{t,k-1}}(x_{t,k-1}) - \nabla F_{\beta_{t,k}}(x_{t,k},\xi_j) + \nabla F_{\beta_{t,k-1}}(x_{t,k-1},\xi_j)\rangle\Big] \qquad (28) \\
&= \frac{1}{K_t^2}\sum_{i=1}^{K_t}\mathbb{E}_{t,k}\left[\|\nabla F_{\beta_{t,k}}(x_{t,k}) - \nabla F_{\beta_{t,k-1}}(x_{t,k-1}) - \nabla F_{\beta_{t,k}}(x_{t,k},\xi_i) + \nabla F_{\beta_{t,k-1}}(x_{t,k-1},\xi_i)\|^2\right] \qquad (29) \\
&= \frac{K_t}{K_t^2}\mathbb{E}_{t,k}\left[\|\nabla F_{\beta_{t,k}}(x_{t,k}) - \nabla F_{\beta_{t,k-1}}(x_{t,k-1}) - \nabla F_{\beta_{t,k}}(x_{t,k},\xi) + \nabla F_{\beta_{t,k-1}}(x_{t,k-1},\xi)\|^2\right] \\
&= \frac{1}{K_t}\mathbb{E}_{t,k}\|\nabla f(x_{t,k}) - \nabla f(x_{t,k-1}) - \nabla f(x_{t,k},\xi) + \nabla f(x_{t,k-1},\xi) \\
&\qquad + \nabla G_{\beta_{t,k}}(Ax_{t,k}) - \nabla G_{\beta_{t,k-1}}(Ax_{t,k-1}) - \nabla g_{\beta_{t,k}}(A(\xi)x_{t,k}) + \nabla g_{\beta_{t,k-1}}(A(\xi)x_{t,k-1})\|^2 \\
&\le \underbrace{\frac{2}{K_t}\mathbb{E}_{t,k}\|\nabla f(x_{t,k}) - \nabla f(x_{t,k-1}) - \nabla f(x_{t,k},\xi) + \nabla f(x_{t,k-1},\xi)\|^2}_{=T_1} \\
&\quad + \underbrace{\frac{2}{K_t}\mathbb{E}_{t,k}\|\nabla G_{\beta_{t,k}}(Ax_{t,k}) - \nabla G_{\beta_{t,k-1}}(Ax_{t,k-1}) - \nabla g_{\beta_{t,k}}(A(\xi)x_{t,k}) + \nabla g_{\beta_{t,k-1}}(A(\xi)x_{t,k-1})\|^2}_{=T_2}
\end{aligned}$$

Line (27) comes from the use of an averaged gradient with batch size K_t. Line (28) comes from applying the square norm to the inner sum, and linearity of expectation. Line (29) comes from plugging the expectation inside the inner product as allowed by the independence of the samples $A(\xi_i)$ and $A(\xi_j)$ (if $X \perp Y$, then $\mathbb{E}_{t,k}[XY] = \mathbb{E}_{t,k}[X]\,\mathbb{E}_{t,k}[Y]$). This results in each term being zero, due to stochastic gradient unbiasedness.

We evaluate the terms T_1 and T_2 separately:

$$
\begin{aligned}
T_2 &= \frac{2}{K_t}\mathbb{E}_{t,k}\|\nabla G_{\beta_{t,k}}(Ax_{t,k}) - \nabla G_{\beta_{t,k-1}}(Ax_{t,k-1}) - \nabla g_{\beta_{t,k}}(A(\xi)x_{t,k}) + \nabla g_{\beta_{t,k-1}}(A(\xi)x_{t,k-1})\|^2 \\
&= \frac{2}{K_t}\mathbb{E}_{t,k}\big[\|\nabla G_{\beta_{t,k}}(Ax_{t,k}) - \nabla G_{\beta_{t,k-1}}(Ax_{t,k-1})\|^2 \\
&\qquad - 2\langle \nabla G_{\beta_{t,k}}(Ax_{t,k}) - \nabla G_{\beta_{t,k-1}}(Ax_{t,k-1}),\ \nabla g_{\beta_{t,k}}(A(\xi)x_{t,k}) - \nabla g_{\beta_{t,k-1}}(A(\xi)x_{t,k-1})\rangle \\
&\qquad + \|\nabla g_{\beta_{t,k}}(A(\xi)x_{t,k}) - \nabla g_{\beta_{t,k-1}}(A(\xi)x_{t,k-1})\|^2\big] \\
&= \frac{2}{K_t}\big(\|\nabla G_{\beta_{t,k}}(Ax_{t,k}) - \nabla G_{\beta_{t,k-1}}(Ax_{t,k-1})\|^2 \\
&\qquad - 2\langle \nabla G_{\beta_{t,k}}(Ax_{t,k}) - \nabla G_{\beta_{t,k-1}}(Ax_{t,k-1}),\ \mathbb{E}_{t,k}\left[\nabla g_{\beta_{t,k}}(A(\xi)x_{t,k}) - \nabla g_{\beta_{t,k-1}}(A(\xi)x_{t,k-1})\right]\rangle \\
&\qquad + \mathbb{E}_{t,k}\left[\|\nabla g_{\beta_{t,k}}(A(\xi)x_{t,k}) - \nabla g_{\beta_{t,k-1}}(A(\xi)x_{t,k-1})\|^2\right]\big) \\
&= \frac{2}{K_t}\big(\mathbb{E}_{t,k}\left[\|\nabla g_{\beta_{t,k}}(A(\xi)x_{t,k}) - \nabla g_{\beta_{t,k-1}}(A(\xi)x_{t,k-1})\|^2\right] - \|\nabla G_{\beta_{t,k}}(Ax_{t,k}) - \nabla G_{\beta_{t,k-1}}(Ax_{t,k-1})\|^2\big) \\
&\le \frac{2}{K_t}\mathbb{E}_{t,k}\left[\|\nabla g_{\beta_{t,k}}(A(\xi)x_{t,k}) - \nabla g_{\beta_{t,k}}(A(\xi)x_{t,k-1}) + \nabla g_{\beta_{t,k}}(A(\xi)x_{t,k-1}) - \nabla g_{\beta_{t,k-1}}(A(\xi)x_{t,k-1})\|^2\right] \\
&= \frac{2}{K_t}\mathbb{E}_{t,k}\Big[\|\frac{1}{\beta_{t,k}}A^T(\xi)A(\xi)\left(x_{t,k} - x_{t,k-1}\right) + \frac{1}{\beta_{t,k}}A^T(\xi)\left[\Pi_{b(\xi)}\left(A(\xi)x_{t,k-1}\right) - \Pi_{b(\xi)}\left(A(\xi)x_{t,k}\right)\right] \\
&\qquad\qquad + \left(\frac{1}{\beta_{t,k}} - \frac{1}{\beta_{t,k-1}}\right)A^T(\xi)\left[A(\xi)x_{t,k-1} - \Pi_{b(\xi)}\left(A(\xi)x_{t,k-1}\right)\right]\|^2\Big] \\
&\le \frac{2}{K_t}\mathbb{E}_{t,k}\Big[\frac{3L_A^2}{\beta_{t,k}^2}\|x_{t,k} - x_{t,k-1}\|^2 + \frac{3L_A}{\beta_{t,k}^2}\|\Pi_{b(\xi)}\left(A(\xi)x_{t,k-1}\right) - \Pi_{b(\xi)}\left(A(\xi)x_{t,k}\right)\|^2 \\
&\qquad\qquad + 3L_A\left(\frac{1}{\beta_{t,k}} - \frac{1}{\beta_{t,k-1}}\right)^2\|A(\xi)x_{t,k-1} - \Pi_{b(\xi)}\left(A(\xi)x_{t,k-1}\right)\|^2\Big] \\
&\le \frac{2}{K_t}\mathbb{E}_{t,k}\Big[\frac{3L_A^2}{\beta_{t,k}^2}\|x_{t,k} - x_{t,k-1}\|^2 + \frac{3L_A^2}{\beta_{t,k}^2}\|x_{t,k-1} - x_{t,k}\|^2 + 3L_A\left(\frac{1}{\beta_{t,k}} - \frac{1}{\beta_{t,k-1}}\right)^2\|A(\xi)x_{t,k-1} - A(\xi)x^*\|^2\Big] \qquad (30) \\
&\le \frac{2}{K_t}\Big[\frac{6L_A^2\gamma_{t,k-1}^2\mathcal{D}_{\mathcal{X}}^2}{\beta_{t,k}^2} + 3L_A^2\mathcal{D}_{\mathcal{X}}^2\left(\frac{1}{\beta_{t,k}} - \frac{1}{\beta_{t,k-1}}\right)^2\Big] \qquad (31) \\
&\le \frac{2L_A^2\mathcal{D}_{\mathcal{X}}^2}{\beta_0^2K_t(K_t+k-1)}\Big[\frac{24(K_t+k)}{(K_t+k-1)} + \frac{3}{4}\Big] \qquad (32) \\
&\le \frac{98L_A^2\mathcal{D}_{\mathcal{X}}^2}{\beta_0^2K_t(K_t+k-1)} \qquad (33)
\end{aligned}
$$

where line (30) comes from the non-expansiveness of projections and $\|A(\xi)x_{t,k-1} - \Pi_{b(\xi)}\left(A(\xi)x_{t,k-1}\right)\| \le \|A(\xi)x_{t,k-1} - y\|$, $\forall y \in b(\xi)$, and line (31) comes from the iterate update rule and the definition of $\mathcal{D}_{\mathcal{X}}$. Line (32) comes from replacing the parameter rates and the fact that:

$$
0 \le \frac{1}{\beta_{t,k}} - \frac{1}{\beta_{t,k-1}} = \frac{1}{\beta_0}\left(\sqrt{K_t+k} - \sqrt{K_t+k-1}\right)
$$

$$= \frac{1}{\beta_0}\left(\frac{1}{\sqrt{K_t+k}+\sqrt{K_t+k-1}}\right)$$
$$\leq \frac{1}{2\beta_0\sqrt{K_t+k-1}}$$

Now we evaluate T_1 and use the fact that $\nabla f(x,\xi)$ are L_f-Lipschitz:

$$\begin{aligned}
T_1 &= \frac{2}{K_t}\mathbb{E}_{t,k}\left[\|\nabla f(x_{t,k}) - \nabla f(x_{t,k-1}) - \nabla f(x_{t,k},\xi) + \nabla f(x_{t,k-1},\xi)\|^2\right] \\
&= \frac{2}{K_t}\Bigg(\|\nabla f(x_{t,k}) - \nabla f(x_{t,k-1})\|^2 + \mathbb{E}_{t,k}\left[\|\nabla f(x_{t,k},\xi) - \nabla f(x_{t,k-1},\xi)\|^2\right] \\
&\qquad\qquad - 2\langle \nabla f(x_{t,k}) - \nabla f(x_{t,k-1}), \mathbb{E}_{t,k}\left[\nabla f(x_{t,k},\xi) - \nabla f(x_{t,k-1},\xi)\right]\rangle\Bigg) \\
&\leq \frac{2}{K_t}\mathbb{E}_{t,k}\left[\|\nabla f(x_{t,k},\xi) - \nabla f(x_{t,k-1},\xi)\|^2\right] \\
&\leq \frac{2L_f^2}{K_t}\|x_{t,k} - x_{t,k-1}\|^2 \\
&\leq \frac{2L_f^2\gamma_{t,k-1}^2\mathcal{D}_{\mathcal{X}}^2}{K_t} \\
&= \frac{8L_f^2\mathcal{D}_{\mathcal{X}}^2}{K_t(K_t+k-1)^2}
\end{aligned} \tag{34}$$

Plugging in (34) and (33) into the expression of T, we get that

$$\begin{aligned}
T &\leq \frac{8L_f^2\mathcal{D}_{\mathcal{X}}^2}{K_t(K_t+k-1)^2} + \frac{98L_A^2\mathcal{D}_{\mathcal{X}}^2}{\beta_0^2 K_t(K_t+k-1)} \\
&\leq \frac{\mathcal{D}_{\mathcal{X}}^2\left(8L_f^2 + \frac{98L_A^2}{\beta_0^2}\right)}{K_t(K_t+k-1)}
\end{aligned} \tag{35}$$

Now we telescope the sum in Equation (26) and get

$$\begin{aligned}
\mathbb{E}_{t,1}\left[\|\nabla F_{\beta_{t,k}}(x_{t,k}) - v_{t,k}\|^2\right] &= \mathbb{E}_{t,1}\left[\mathbb{E}_{t,2}\left[\ldots\mathbb{E}_{t,k}\left[\|\nabla F_{\beta_{t,k}}(x_{t,k}) - v_{t,k}\|^2\right]\right]\right] \\
&\leq \frac{\mathcal{D}_{\mathcal{X}}^2}{K_t}\left(8L_f^2 + \frac{98L_A^2}{\beta_0^2}\right)\sum_{i=2}^{k}\frac{1}{K_t+i-1} + \mathbb{E}_{t,1}\left[\|\nabla F_{\beta_{t,1}}(x_{t,1}) - v_{t,1}\|^2\right] \\
&\leq \frac{\mathcal{D}_{\mathcal{X}}^2}{K_t}\left(8L_f^2 + \frac{98L_A^2}{\beta_0^2}\right)\sum_{i=2}^{k}\frac{1}{\frac{K_t+k}{2}} + \mathbb{E}_{t,1}\left[\|\nabla F_{\beta_{t,1}}(x_{t,1}) - v_{t,1}\|^2\right] \qquad (36) \\
&= \frac{2\mathcal{D}_{\mathcal{X}}^2}{K_t}\left(8L_f^2 + \frac{98L_A^2}{\beta_0^2}\right)\frac{k-1}{K_t+k} + \mathbb{E}_{t,1}\left[\|\nabla F_{\beta_{t,1}}(x_{t,1}) - v_{t,1}\|^2\right] \\
&\leq \frac{2\mathcal{D}_{\mathcal{X}}^2}{K_t+k}\left(8L_f^2 + \frac{98L_A^2}{\beta_0^2}\right) + \mathbb{E}_{t,1}\left[\|\nabla F_{\beta_{t,1}}(x_{t,1}) - v_{t,1}\|^2\right] \qquad (37)
\end{aligned}$$

where line (36) comes from the fact that

$$2 \leq k \leq 2^{t-1} = K_t \implies 2^{t-2} + 1 \leq \frac{K_t + k}{2} \leq 2^{t-1} \textbf{ and}$$
$$2 \leq i \leq k \leq 2^{t-1} \implies 2^{t-1} + 1 \leq K_t + i - 1 \leq 2^t - 1$$

and line (37) comes from $k - 1 \leq K_t$. □

Lemma 4.2 (Estimator variance for finite-sum problems)**.** *Consider Algorithm 2, and let ξ be finitely sampled from set $[n]$, $\xi_{\mathcal{Q}_t} = [n]$ and $\xi_{\mathcal{S}_{t,k}}$, such that $|\mathcal{S}_{t,k}| = K_t = 2^{t-1}$. Also, let $\gamma_{t,k} = \frac{2}{K_t+k}$, $\beta_{t,k} = \frac{\beta_0}{\sqrt{K_t+k}}$. Then, for a fixed t and for all $k \leq K_t$,*

$$\mathbb{E}\left[\left\|\nabla F_{\beta_{t,k}}(x_{t,k}) - v_{t,k}\right\|^2\right] \leq \frac{C_1}{K_t + k}, \tag{38}$$

where $C_1 = 2\mathcal{D}_{\mathcal{X}}^2 \left(8L_f^2 + \frac{98L_A^2}{\beta_0^2}\right)$.

Proof

The result directly follows from the fact that we take a full gradient in the outer loop ($\xi_{\mathcal{Q}_t} = [n]$), thus zeroing out the term $\mathbb{E}_{t,1}\left[\left\|\nabla F_{\beta_{t,1}}(x_{t,1}) - v_{t,1}\right\|^2\right]$ of Lemma B.4. Taking the full expectation on both sides gives us the stated result. □

Lemma 4.3 (Estimator variance for generic problems)**.** *Consider Algorithm 2 and let $\xi \sim P(\xi)$ and $\xi_{\mathcal{Q}_t}$ such that $|\mathcal{Q}_t| = \lceil \frac{2K_t}{\beta_{t,1}^2} \rceil$. Also, let $\xi_{\mathcal{S}_{t,k}}$, such that $|\mathcal{S}_{t,k}| = K_t = 2^{t-1}$, $\gamma_{t,k} = \frac{2}{K_t+k}$, $\beta_{t,k} = \frac{\beta_0}{\sqrt{K_t+k}}$. Then, for a fixed t and for all $k \leq K_t$,*

$$\mathbb{E}\left[\left\|\nabla F_{\beta_{t,k}}(x_{t,k}) - v_{t,k}\right\|^2\right] \leq \frac{C_2}{K_t + k}, \tag{39}$$

where $C_2 = 16L_f^2\mathcal{D}_{\mathcal{X}}^2 + 2L_A^2\mathcal{D}_{\mathcal{X}}^2\left(\frac{98}{\beta_0^2} + 1\right) + 2\beta_0^2\sigma_f^2$.

Proof

From the use of averaged gradient and Technical Observation c.:

$$\begin{aligned}
\mathbb{E}_{t,1}\left[\left\|\nabla F_{\beta_{t,1}}(x_{t,1}) - v_{t,1}\right\|^2\right] &\leq \frac{1}{|\mathcal{Q}_t|}\mathbb{E}_{t,1}\left[\left\|\nabla f(x_{t,1}) - \nabla f(x_{t,1},\xi) + \nabla G_{\beta_{t,1}}(Ax_{t,1}) - \nabla g_{\beta_{t,1}}(A(\xi)x_{t,1})\right\|^2\right] \\
&\leq \frac{1}{|\mathcal{Q}_t|}\left(2\mathbb{E}_{t,1}\left[\left\|\nabla f(x_{t,1}) - \nabla f(x_{t,1},\xi)\right\|^2\right] + 2\mathbb{E}_{t,1}\left[\left\|\nabla G_{\beta_{t,1}}(Ax_{t,1}) - \nabla g_{\beta_{t,1}}(A(\xi)x_{t,1})\right\|^2\right)\right] \\
&\leq \frac{\beta_{t,1}^2}{2K_t}\left(2\sigma_f^2 + \frac{2L_A^2\mathcal{D}_{\mathcal{X}}^2}{\beta_{t,1}^2}\right) \\
&\leq \frac{\beta_0^2}{2K_t(K_t+1)}\left(2\sigma_f^2 + \frac{2L_A^2\mathcal{D}_{\mathcal{X}}^2(K_t+1)}{\beta_0^2}\right) \\
&\leq \frac{\beta_0^2\sigma_f^2}{K_t^2} + \frac{L_A^2\mathcal{D}_{\mathcal{X}}^2}{K_t} \\
&\leq \frac{1}{K_t+k}\left(2\beta_0^2\sigma_f^2 + 2L_A^2\mathcal{D}_{\mathcal{X}}^2\right)
\end{aligned}$$

Where we have used that $2K_t \geq K_t + k$ and $K_t^2 \geq K_t = \frac{2K_t}{2} \geq \frac{K_t+k}{2}$, $\forall K_t \in \mathbb{N}, K_t \geq 1, \forall k \leq K_t$. Replacing in (37), we obtain the desired result. □

Theorem 4.2. *Consider Algorithm 2 with parameters* $\gamma_{t,k} = \frac{2}{K_t+k}$, $\beta_{t,k} = \frac{\beta 0}{\sqrt{K_t+k}}$ *and* $\xi_{\mathcal{S}_{t,k}}$, *such that* $|\mathcal{S}_{t,k}| = K_t = 2^{t-1}$. *Then,*

- *For* ξ *be finitely sampled from set* $[n]$ *and* $\xi_{\mathcal{Q}_t} = [n]$,

$$\mathbb{E}\left[S_{\beta_{t,k}}(x_{t,k+1})\right] \leq \frac{C_3}{\sqrt{K_t+k+1}}, \ \ \forall t \in \mathbb{N},\, 1 \leq k \leq 2^{t-1}$$

where $C_3 = \max\left\{S_{\beta_{1,0}}(x_{1,1}),\ 2\mathcal{D}_{\mathcal{X}}^2 L_f + 2\mathcal{D}_{\mathcal{X}}^2\sqrt{16L_f^2 + \frac{196L_A^2}{\beta_0^2}} + \frac{2\mathcal{D}_{\mathcal{X}}^2 L_A}{\beta_0}\right\}$;

- *For* $\xi \sim P(\xi)$ *and* $\xi_{\mathcal{Q}_t}$ *such that* $|\mathcal{Q}_t| = \lceil \frac{2K_t}{\beta_{t,1}^2} \rceil$,

$$\mathbb{E}\left[S_{\beta_{t,k}}(x_{t,k+1})\right] \leq \frac{C_4}{\sqrt{K_t+k+1}}, \ \ \forall t \in \mathbb{N},\, 1 \leq k \leq 2^{t-1}$$

where $C_4 = \max\left\{S_{\beta_{1,0}}(x_{1,1}),\ 2\mathcal{D}_{\mathcal{X}}^2 L_f + \frac{2\mathcal{D}_{\mathcal{X}}^2 L_A}{\beta_0} + 2\mathcal{D}_{\mathcal{X}}\sqrt{16L_f^2\mathcal{D}_{\mathcal{X}}^2 + 2L_A^2\mathcal{D}_{\mathcal{X}}^2\left(\frac{98}{\beta_0^2}+1\right) + 2\beta_0^2\sigma_f^2}\right\}$.

Proof

The proof has two steps, coming from the nested loop structure of Algorithm 2. We first determine the recursion for $S_{\beta_{t,k}}(x_{t,k+1})$ for all the iterates of the inner loop (constant t) and then show that the recursion holds at the 'edges' i.e., when going from $t-1$ to t.

1. Convergence Recursion

1.1 Recursion of $S_{\beta_{t,k}}$ for Constant t (Inner Loop)

Using Technical Observation d., the definition of $\mathcal{D}_{\mathcal{X}}$ and the optimality of $w_{t,k}$:

$$\begin{aligned}
F_{\beta_{t,k}}(x_{k+1}) &= \mathbb{E}_{t,k}\left[F_{\beta_{t,k}}(x_{t,k},\xi)\right] \\
&\leq \mathbb{E}_{t,k}\left[F_{\beta_{t,k}}(x_{t,k},\xi) + \langle \nabla F_{\beta_{t,k}}(x_{t,k},\xi),\, x_{t,k+1}-x_{t,k}\rangle + \frac{L_f + \frac{L_A}{\beta_{t,k}}}{2}\|x_{t,k+1}-x_{t,k}\|^2\right] \\
&\leq F_{\beta_{t,k}}(x_{t,k}) + \gamma_{t,k}\langle \nabla F_{\beta_{t,k}}(x_{t,k}),\, w_{t,k}-x_{t,k}\rangle + \frac{\gamma_{t,k}^2(L_f+\frac{L_A}{\beta_{t,k}})}{2}\|w_{t,k}-x_{t,k}\|^2 \\
&\leq F_{\beta_{t,k}}(x_{t,k}) + \gamma_{t,k}\langle \nabla F_{\beta_{t,k}}(x_{t,k}),\, w_{t,k}-x_{t,k}\rangle + \frac{\mathcal{D}_{\mathcal{X}}^2\gamma_{t,k}^2}{2}(L_f+\frac{L_A}{\beta_{t,k}}) \\
&\leq F_{\beta_{t,k}}(x_{t,k}) + \gamma_{t,k}\left(\langle \nabla F_{\beta_{t,k}}(x_{t,k}) - v_{t,k},\, w_{t,k}-x_{t,k}\rangle + \langle v_{t,k},\, x^*-x_{t,k}\rangle\right) + \frac{\mathcal{D}_{\mathcal{X}}^2\gamma_{t,k}^2}{2}(L_f+\frac{L_A}{\beta_{t,k}})
\end{aligned} \tag{40}$$

We process the second term above separately, using the convexity of f, Technical Observation 2(e.)i and noting that $v_{t,k-1} - v_{t,k} = \tilde{\nabla}F_{\beta_{t,k-1}}(x_{t,k-1},\xi_{\mathcal{S}_{t,k}}) - \tilde{\nabla}F_{\beta_{t,k}}(x_{t,k},\xi_{\mathcal{S}_{t,k}})$:

$$\begin{aligned}
&\langle \nabla F_{\beta_{t,k}}(x_{t,k}) - v_{t,k},\, w_{t,k}-x_{t,k}\rangle + \langle v_{t,k},\, x^*-x_{t,k}\rangle \\
&\quad= \langle \nabla F_{\beta_{t,k}}(x_{t,k}) - v_{t,k},\, w_{t,k}-x^*\rangle + \langle \nabla F_{\beta_{t,k}}(x_{t,k}) - v_{t,k},\, x^*-x_{t,k}\rangle \\
&\qquad\qquad + \langle v_{t,k-1} - \tilde{\nabla}F_{\beta_{t,k-1}}(x_{t,k-1},\xi_{\mathcal{S}_{t,k}}) + \tilde{\nabla}F_{\beta_k}(x_{t,k},\xi_{\mathcal{S}_{t,k}}),\, x^*-x_{t,k}\rangle \\
&\quad= \langle \nabla F_{\beta_{t,k}}(x_{t,k}) - v_{t,k},\, w_{t,k}-x^*\rangle + \langle \nabla F_{\beta_{t,k}}(x_{t,k}) - v_{t,k} + v_{t,k-1} - \tilde{\nabla}F_{\beta_{t,k-1}}(x_{t,k-1},\xi_{\mathcal{S}_{t,k}}),\, x^*-x_{t,k}\rangle
\end{aligned}$$

$$
\begin{aligned}
&\quad + \langle \tilde{\nabla} f(x_{t,k}, \xi_{\mathcal{S}_{t,k}}),\ x^* - x_{t,k} \rangle + \langle A^T(\xi_{\mathcal{S}_{t,k}}) \tilde{\nabla} g_{\beta_{t,k}}(A(\xi_{\mathcal{S}_{t,k}}) x_{t,k}),\ x^* - x_{t,k} \rangle \\
&\leq \langle \nabla F_{\beta_{t,k}}(x_{t,k}) - v_{t,k},\ w_{t,k} - x^* \rangle + \langle \nabla F_{\beta_{t,k}}(x_{t,k}) - \tilde{\nabla} F_{\beta_{t,k}}(x_{t,k}, \xi_{\mathcal{S}_{t,k}}),\ x_{t,k} - x^* \rangle \\
&\qquad + \tilde{f}(x^*, \xi_{\mathcal{S}_{t,k}}) - \tilde{f}(x_{t,k}, \xi_{\mathcal{S}_{t,k}}) + \langle \tilde{\nabla} g_{\beta_{t,k}}(A(\xi_{\mathcal{S}_{t,k}}) x_{t,k}),\ A(\xi_{\mathcal{S}_{t,k}}) x^* - A^T(\xi_{\mathcal{S}_{t,k}}) x_{t,k} \rangle \\
&\leq \langle \nabla F_{\beta_{t,k}}(x_{t,k}) - v_{t,k},\ w_{t,k} - x^* \rangle + \langle \nabla F_{\beta_{t,k}}(x_{t,k}) - \tilde{\nabla} F_{\beta_{t,k}}(x_{t,k}, \xi_{\mathcal{S}_{t,k}}),\ x_{t,k} - x^* \rangle \\
&\qquad + \tilde{f}(x^*, \xi_{\mathcal{S}_{t,k}}) + \underbrace{\tilde{g}(A(\xi_{\mathcal{S}_{t,k}}) x^*)}_{=0 \text{ a.s.}} \underbrace{- \tilde{f}(x_{t,k}, \xi_{\mathcal{S}_{t,k}}) - \tilde{g}_{\beta_{t,k}}(A(\xi_{\mathcal{S}_{t,k}}) x_{t,k})}_{= -\tilde{F}_{\beta_{t,k}}(x_{t,k}, \xi_{\mathcal{S}_{t,k}})} - \frac{\beta_{t,k}}{2} \|\lambda^*_{\beta_{t,k}}(\widetilde{A(\xi_{\mathcal{S}_{t,k}})} x_{t,k})\|^2
\end{aligned}
\tag{41}
$$

We can now resume Equation (40) by plugging in the inequality in (41), subtracting $f(x^*)$ from both sides, and taking the conditional expectation $\mathbb{E}_{t,k}[X] = \mathbb{E}[X|\mathcal{F}_{t,k}]$.

$$
\begin{aligned}
&\mathbb{E}_{t,k}\left[F_{\beta_{t,k}}(x_{k+1}) - f(x^*)\right] \\
&\leq \mathbb{E}_{t,k}\left[F_{\beta_{t,k}}(x_{t,k}) + \gamma_{t,k}\left(\langle \nabla F_{\beta_{t,k}}(x_{t,k}) - v_{t,k},\ w_{t,k} - x_{t,k} \rangle + \langle v_{t,k},\ x^* - x_{t,k} \rangle\right) + \frac{\mathcal{D}_{\mathcal{X}}^2 \gamma_{t,k}^2}{2}\left(L_f + \frac{L_A}{\beta_{t,k}}\right)\right] - f(x^*) \\
&\leq F_{\beta_{t,k}}(x_{t,k}) + \gamma_{t,k}\Bigg(\langle \nabla F_{\beta_{t,k}}(x_{t,k}) - v_{t,k},\ w_{t,k} - x^* \rangle + \underbrace{\mathbb{E}_{t,k}\left[\langle \nabla F_{\beta_{t,k}}(x_{t,k}) - \tilde{\nabla} F_{\beta_{t,k}}(x_{t,k}, \xi_{\mathcal{S}_{t,k}}),\ x_{t,k} - x^* \rangle\right]}_{=0,\ \text{unbiasedness}} \\
&\quad + \mathbb{E}_{t,k}\left[\tilde{f}(x^*, \xi_{\mathcal{S}_{t,k}}) - \tilde{F}_{\beta_{t,k}}(x_{t,k}, \xi_{\mathcal{S}_{t,k}}) - \frac{\beta_{t,k}}{2}\|\lambda^*_{\beta_{t,k}}(\widetilde{A(\xi_{\mathcal{S}_{t,k}})} x_{t,k})\|^2\right]\Bigg) + \frac{\mathcal{D}_{\mathcal{X}}^2 \gamma_{t,k}^2}{2}\left(L_f + \frac{L_A}{\beta_{t,k}}\right) - f(x^*) \\
&\leq F_{\beta_{t,k}}(x_{t,k}) + \gamma_{t,k}\Bigg(\langle \nabla F_{\beta_{t,k}}(x_{t,k}) - v_{t,k},\ w_{t,k} - x^* \rangle + f(x^*) - F_{\beta_{t,k}}(x_{t,k}) - \mathbb{E}_{t,k}\left[\frac{\beta_{t,k}}{2}\|\lambda^*_{\beta_{t,k}}(\widetilde{A(\xi_{\mathcal{S}_{t,k}})} x_{t,k})\|^2\right]\Bigg) \\
&\qquad + \frac{\mathcal{D}_{\mathcal{X}}^2 \gamma_{t,k}^2}{2}\left(L_f + \frac{L_A}{\beta_{t,k}}\right) - f(x^*) \\
&= (1 - \gamma_{t,k})(F_{\beta_{t,k}}(x_{t,k}) - f(x^*)) + \gamma_{t,k}\langle \nabla F_{\beta_{t,k}}(x_{t,k}) - v_{t,k},\ w_{t,k} - x^* \rangle - \frac{\gamma_{t,k}\beta_{t,k}}{2}\mathbb{E}_{t,k}\left[\|\lambda^*_{\beta_{t,k}}(\widetilde{A(\xi_{\mathcal{S}_{t,k}})} x_{t,k})\|^2\right] \\
&\qquad + \frac{\mathcal{D}_{\mathcal{X}}^2 \gamma_{t,k}^2}{2}\left(L_f + \frac{L_A}{\beta_{t,k}}\right)
\end{aligned}
$$

Using Technical Observation 2(e.)ii we observe that

$$
\begin{aligned}
F_{\beta_{t,k}}(x_{t,k}) &= \mathbb{E}_{t,k}\left[\tilde{f}(x_{t,k}, \xi_{\mathcal{S}_{t,k}}) + \tilde{g}_{\beta_{t,k}}(A(\xi_{\mathcal{S}_{t,k}}) x_{t,k})\right] \\
&\leq \mathbb{E}_{t,k}\left[\tilde{f}(x_{t,k}, \xi_{\mathcal{S}_{t,k}}) + \tilde{g}_{\beta_{t,k-1}}(A(\xi_{\mathcal{S}_{t,k}}) x_{t,k}) + \frac{\beta_{t,k-1} - \beta_{t,k}}{2}\|\lambda^*_{\beta_{t,k}}(\widetilde{A(\xi_{\mathcal{S}_{t,k}})} x_{t,k})\|^2\right] \\
&= F_{\beta_{t,k-1}}(x_{t,k}) + \mathbb{E}_{t,k}\left[\frac{\beta_{t,k-1} - \beta_{t,k}}{2}\|\lambda^*_{\beta_{t,k}}(\widetilde{A(\xi_{\mathcal{S}_{t,k}})} x_{t,k})\|^2\right].
\end{aligned}
$$

Using the above and the definition of $\mathcal{D}_{\mathcal{X}}$, we continue the inequality as:

$$\mathbb{E}_{t,k}\left[F_{\beta_{t,k}}(x_{k+1}) - f(x^*)\right] \leq (1-\gamma_{t,k})(F_{\beta_{t,k-1}}(x_{t,k}) - f(x^*)) + \gamma_{t,k}\mathcal{D}_{\mathcal{X}}\|\nabla F_{\beta_{t,k}}(x_{t,k}) - v_{t,k}\| + \frac{(1-\gamma_{t,k})(\beta_{t,k-1}-\beta_{t,k}) - \gamma_{t,k}\beta_{t,k}}{2}\mathbb{E}_{t,k}\left[\|\lambda^*_{\beta_{t,k}}(\widetilde{A(\xi_{\mathcal{S}_{t,k}})}x_{t,k})\|^2\right] + \frac{\mathcal{D}^2_{\mathcal{X}}\gamma^2_{t,k}}{2}\left(L_f + \frac{L_A}{\beta_{t,k}}\right) \tag{42}$$

Using the stated parameter rates, we notice that $(1-\gamma_{t,k})(\beta_{t,k-1}-\beta_{t,k}) - \gamma_{t,k}\beta_{t,k} < 0$, as follows:

$$\begin{aligned}
&\left(1-\frac{2}{K_t+k}\right)\left(\frac{\beta_0}{\sqrt{K_t+k-1}} - \frac{\beta_0}{\sqrt{K_t+k}}\right) - \frac{2\beta_0}{(K_t+k)\sqrt{K_t+k}} \\
&= \frac{\beta_0}{\sqrt{K_t+k-1}} - \frac{\beta_0}{\sqrt{K_t+k}} - \frac{2\beta_0}{(K_t+k)\sqrt{K_t+k-1}} \\
&= \beta_0\frac{K_t+k-\sqrt{K_t+k}\sqrt{K_t+k-1}-2}{(K_t+k)\sqrt{K_t+k-1}} \\
&= \beta_0\frac{(K_t+k-1) - 2\sqrt{\frac{K_t+k}{4}}\sqrt{K_t+k-1} + \frac{K_t+k}{4} - \frac{K_t+k}{4} - 1}{(K_t+k)\sqrt{K_t+k-1}} \\
&= \beta_0\frac{(\sqrt{K_t+k-1} - \frac{\sqrt{K_t+k}}{2})^2 - \frac{K_t+k}{4} - 1}{(K_t+k)\sqrt{K_t+k-1}} \\
&= \beta_0\frac{(\sqrt{K_t+k-1} - \frac{\sqrt{K_t+k}}{2} - \frac{\sqrt{K_t+k}}{2})(\sqrt{K_t+k-1} - \frac{\sqrt{K_t+k}}{2} + \frac{\sqrt{K_t+k}}{2}) - 1}{(K_t+k)\sqrt{K_t+k-1}} \\
&= \beta_0\frac{\overbrace{(\sqrt{K_t+k-1} - \sqrt{K_t+k})}^{<0}\sqrt{K_t+k-1} - 1}{(K_t+k)\sqrt{K_t+k-1}} \\
&< 0
\end{aligned} \tag{43}$$

Finally, noting the definition of $S_{\beta_{t,k}}(x_{t,k+1})$ and taking full expectation on both sides, we arrive at:

$$\mathbb{E}\left[S_{\beta_{t,k}}(x_{t,k+1})\right] \leq (1-\gamma_{t,k})\mathbb{E}\left[S_{\beta_{t,k-1}}(x_{t,k})\right] + \gamma_{t,k}\mathcal{D}_{\mathcal{X}}\mathbb{E}\left[\|\nabla F_{\beta_{t,k}}(x_{t,k}) - v_{t,k}\|\right] + \frac{\mathcal{D}^2_{\mathcal{X}}\gamma^2_{t,k}}{2}\left(L_f + \frac{L_A}{\beta_{t,k}}\right) \tag{44}$$

1.2 Recursion of $S_{\beta_{t,k}}$ at the 'Edges'

We now want to show that the same recursion holds when going for $S_{\beta_{t,1}}(x_{t,2})$ and $S_{\beta_{t-1,K_{t-1}}}(x_{t-1,K_{t-1}+1})$. We follow similar steps as in the previous section (which we shorten this time for conciseness). Using smoothness and the fact that from Algorithm 2 we have $x_{t,1} = x_{t-1,K_{t-1}+1}$:

$$F_{\beta_{t,1}}(x_{t,2}) \leq F_{\beta_{t,1}}(x_{t,1}) + \gamma_{t,1}\langle\nabla F_{\beta_{t,1}}(x_{t,1}),\ w_{t,1} - x_{t,1}\rangle + \frac{\mathcal{D}^2_{\mathcal{X}}\gamma^2_{t,1}}{2}\left(L_f + \frac{L_A}{\beta_{t,1}}\right) \tag{45}$$

Since $v_{t,1} = \nabla F_{\beta_{t,1}}(x_{t,1})$ and $w_{t,1} = \text{lmo}_{\mathcal{X}}(v_{t,1})$, we have that $\langle\nabla F_{\beta_{t,1}}(x_{t,1}),\ w_{t,1} - x_{t,1}\rangle \leq \langle\nabla F_{\beta_{t,1}}(x_{t,1}),\ x^* - x_{t,1}\rangle$. Further using the definition of F_β, the convexity of f and Technical Observation 2(e.)i, we have:

$$\langle\nabla F_{\beta_{t,1}}(x_{t,1}),\ w_{t,1} - x_{t,1}\rangle \leq \langle\nabla F_{\beta_{t,1}}(x_{t,1}),\ x^* - x_{t,1}\rangle$$

$$
\begin{aligned}
&= \langle \nabla f(x_{t,1}) + \nabla_x G_{\beta_{t,1}}(Ax_{t,1}),\ x^* - x_{t,1} \rangle \\
&\leq f(x^*) - f(x_{t,1}) + \mathbb{E}_{t,1}\left[\langle \tilde{\nabla}_x g_{\beta_{t,1}}(A(\xi_{\mathcal{Q}_t})x_{t,1}),\ x^* - x_{t,1} \rangle \right] \\
&\leq f(x^*) - f(x_{t,1}) + \mathbb{E}_{t,1}\left[\underbrace{\tilde{g}(A(\xi_{\mathcal{Q}_t})x^*)}_{=0 \text{ a.s.}} - \tilde{g}_{\beta_{t,1}}(A(\xi_{\mathcal{Q}_t})x_{t,1}) - \frac{\beta_{t,1}}{2}\|\lambda^*_{\beta_{t,1}}(\widetilde{A(\xi_{\mathcal{Q}_t})}x_{t,1})\|^2 \right] \\
&\leq f(x^*) \underbrace{- f(x_{t,1}) - G_{\beta_{t,1}}(Ax_{t,1})}_{=-F_{\beta_{t,1}}(x_t,1)} - \frac{\beta_{t,1}}{2}\mathbb{E}_{t,1}\left[\|\lambda^*_{\beta_{t,1}}(\widetilde{A(\xi_{\mathcal{Q}_t})}x_{t,1})\|^2 \right] \qquad (46)
\end{aligned}
$$

Another remark is that we can still make the transition from $F_{\beta_{t,1}}(x_{t,1})$ to $F_{\beta_{t-1,K_{t-1}}}(x_{t,1})$ using Technical Observation 2(e.)ii, since the β's are 'continuous' at the edge: $\beta_{t-1,K_{t-1}} = \frac{\beta_0}{\sqrt{K_{t-1}+K_{t-1}}} = \frac{\beta_0}{\sqrt{K_t}}$ and $\beta_{t,1} = \frac{\beta_0}{\sqrt{K_t+1}}$. We thus have:

$$
\begin{aligned}
F_{\beta_{t,1}}(x_{t,1}) &= \mathbb{E}_{t,1}\left[\tilde{f}(x_{t,1}, \xi_{\mathcal{Q}_t}) + \tilde{g}_{\beta_{t,1}}(A(\xi_{\mathcal{Q}_t})x_{t,1}) \right] \\
&\leq \mathbb{E}_{t,1}\left[\tilde{f}(x_{t,1}, \xi_{\mathcal{Q}_t}) + \tilde{g}_{\beta_{t-1,K_{t-1}}}(A(\xi_{\mathcal{Q}_t})x_{t,1}) + \frac{\beta_{t-1,K_{t-1}} - \beta_{t,1}}{2}\|\lambda^*_{\beta_{t,1}}(\widetilde{A(\xi_{\mathcal{Q}_t})}x_{t,1})\|^2 \right] \\
&= F_{\beta_{t-1,K_{t-1}}}(x_{t,1}) + \mathbb{E}_{t,1}\left[\frac{\beta_{t-1,K_{t-1}} - \beta_{t,1}}{2}\|\lambda^*_{\beta_{t,1}}(\widetilde{A(\xi_{\mathcal{Q}_t})}x_{t,1})\|^2 \right] \qquad (47)
\end{aligned}
$$

Inserting (47) and (46) into (45):

$$
\begin{aligned}
F_{\beta_{t,1}}(x_{t,2}) &\leq (1-\gamma_{t,1})F_{\beta_{t,1}}(x_{t,1}) + \gamma_{t,1}f(x^*) - \frac{\gamma_{t,1}\beta_{t,1}}{2}\mathbb{E}\left[\|\lambda^*_{\beta_{t,1}}(\widetilde{A(\xi_{\mathcal{Q}_t})}x_{t,1})\|^2 \right] + \frac{\mathcal{D}^2_{\mathcal{X}}\gamma^2_{t,1}}{2}\left(L_f + \frac{L_A}{\beta_{t,1}} \right) \\
&\leq (1-\gamma_{t,1})F_{\beta_{t-1,K_{t-1}}}(x_{t,1}) + \gamma_{t,1}f(x^*) + \underbrace{\frac{(1-\gamma_{t,1})(\beta_{t-1,K_{t-1}} - \beta_{t,1}) - \gamma_{t,1}\beta_{t,1}}{2}}_{<0, \text{ as before}}\mathbb{E}_{t,1}\left[\|\lambda^*_{\beta_{t,1}}(\widetilde{A(\xi_{\mathcal{Q}_t})}x_{t,1})\|^2 \right] \\
&\qquad + \frac{\mathcal{D}^2_{\mathcal{X}}\gamma^2_{t,1}}{2}\left(L_f + \frac{L_A}{\beta_{t,1}} \right)
\end{aligned}
$$

Finally, subtracting $f(x^*)$ from both sides and taking the expectation, we have:

$$
\mathbb{E}\left[S_{\beta_{t,1}}(x_{t,2}) \right] \leq (1-\gamma_{t,1})\mathbb{E}\left[S_{\beta_{t-1,K_{t-1}}}(x_{t,1}) \right] + \frac{\mathcal{D}^2_{\mathcal{X}}\gamma^2_{t,1}}{2}\left(L_f + \frac{L_A}{\beta_{t,1}} \right) \qquad (48)
$$

2. Convergence Rates for the Finite-Sum Case

For ease, we first cast the index pairs (t,k) to their corresponding global index counterparts (in a sense, we flatten the double loop structure). The variables indexed by (t,k) can be seen as equivalently indexed by $\boldsymbol{\kappa}(t,k) = K_t + k := 2^{t-1} + k$, $t \in \mathbb{N}$, $k \in \{1, \dots 2^{t-1}\}$.

The following properties hold for $\boldsymbol{\kappa}$:

- $\boldsymbol{\kappa}(t, k+1) = \boldsymbol{\kappa}(t,k) + 1$
- $\boldsymbol{\kappa}(t-1, K_{t-1}+1) = \boldsymbol{\kappa}(t-1, K_{t-1}) + 1 = \boldsymbol{\kappa}(t,1)$ (the 'increment-by-one' rule holds between the last iteration of epoch $t-1$ and the first iteration of epoch t)

In other words, $\boldsymbol{\kappa}(t,k)$ returns for iteration (t,k) its global index since the beginning of Algorithm 2.

We use this new indexing scheme and its properties to rewrite relations 44 and 48 into a single, global inequality. Note that here $\boldsymbol{\kappa}$ should be read as $\boldsymbol{\kappa}(t,k)$, for some given, arbitrary t,k.

$$\mathbb{E}\left[S_{\beta_{\boldsymbol{\kappa}}}(x_{\boldsymbol{\kappa}+1})\right] \leq (1-\gamma_{\boldsymbol{\kappa}})\mathbb{E}\left[S_{\beta_{\boldsymbol{\kappa}-1}}(x_{\boldsymbol{\kappa}})\right] + \gamma_{\boldsymbol{\kappa}}\mathcal{D}_{\mathcal{X}}\mathbb{E}\left[\|\nabla F_{\beta_{\boldsymbol{\kappa}}}(x_{\boldsymbol{\kappa}}) - v_{\boldsymbol{\kappa}}\|\right] + \frac{\mathcal{D}_{\mathcal{X}}^2\gamma_{\boldsymbol{\kappa}}^2}{2}\left(L_f + \frac{L_A}{\beta_{\boldsymbol{\kappa}}}\right) \tag{49}$$

Further replacing the parameter rates and the variance bound of Lemma 4.2 (subject to Jensen's inequality):

$$\begin{aligned}\mathbb{E}\left[S_{\beta_{\boldsymbol{\kappa}}}(x_{\boldsymbol{\kappa}+1})\right] &= \left(1-\frac{2}{\boldsymbol{\kappa}}\right)\mathbb{E}\left[S_{\beta_{\boldsymbol{\kappa}-1}}(x_{\boldsymbol{\kappa}})\right] + \frac{2\mathcal{D}_{\mathcal{X}}^2\sqrt{16L_f^2 + \frac{196L_A^2}{\beta_0^2}}}{\boldsymbol{\kappa}\sqrt{\boldsymbol{\kappa}}} + \frac{2\mathcal{D}_{\mathcal{X}}^2 L_f}{\boldsymbol{\kappa}^2} + \frac{2\mathcal{D}_{\mathcal{X}}^2 L_A}{\beta_0\boldsymbol{\kappa}\sqrt{\boldsymbol{\kappa}}} \\ &\leq \left(1-\frac{2}{\boldsymbol{\kappa}}\right)\mathbb{E}\left[S_{\beta_{\boldsymbol{\kappa}-1}}(x_{\boldsymbol{\kappa}})\right] + \frac{1}{\boldsymbol{\kappa}^{3/2}}\left(2\mathcal{D}_{\mathcal{X}}^2 L_f + 2\mathcal{D}_{\mathcal{X}}^2\sqrt{16L_f^2 + \frac{196L_A^2}{\beta_0^2}} + \frac{2\mathcal{D}_{\mathcal{X}}^2 L_A}{\beta_0}\right)\end{aligned}$$

We can now apply Lemma B.2, with $\alpha=1$, $\beta=3/2$, $b = 2\mathcal{D}_{\mathcal{X}}^2 L_f + 2\mathcal{D}_{\mathcal{X}}^2\sqrt{16L_f^2 + \frac{196L_A^2}{\beta_0^2}} + \frac{2\mathcal{D}_{\mathcal{X}}^2 L_A}{\beta_0}$, $c=2$, $k_0=0$ and $C_3 = \max\left\{S_{\beta_{1,0}}(x_{1,1}),\ 2\mathcal{D}_{\mathcal{X}}^2 L_f + 2\mathcal{D}_{\mathcal{X}}^2\sqrt{16L_f^2 + \frac{196L_A^2}{\beta_0^2}} + \frac{2\mathcal{D}_{\mathcal{X}}^2 L_A}{\beta_0}\right\}$ to get:

$$\mathbb{E}\left[F_{\beta_{\boldsymbol{\kappa}}}(x_{\boldsymbol{\kappa}+1}) - f(x^*)\right] \leq \frac{C_3}{\sqrt{\boldsymbol{\kappa}+1}}$$

$$\Updownarrow$$

$$\mathbb{E}\left[S_{\beta_{t,k}}(x_{t,k+1})\right] \leq \frac{C_3}{\sqrt{K_t+k+1}}$$

2. Convergence Rates for the General Expectation Case

Following the same steps for the general expectation case, we get:

$$\begin{aligned}\mathbb{E}\left[S_{\beta_{\boldsymbol{\kappa}}}(x_{\boldsymbol{\kappa}+1})\right] &= \left(1-\frac{2}{\boldsymbol{\kappa}}\right)\mathbb{E}\left[S_{\beta_{\boldsymbol{\kappa}-1}}(x_{\boldsymbol{\kappa}})\right] + \frac{2\mathcal{D}_{\mathcal{X}}\sqrt{16L_f^2\mathcal{D}_{\mathcal{X}}^2 + 2L_A^2\mathcal{D}_{\mathcal{X}}^2\left(\frac{98}{\beta_0^2}+1\right) + 2\beta_0^2\sigma_f^2}}{\boldsymbol{\kappa}\sqrt{\boldsymbol{\kappa}}} + \frac{2\mathcal{D}_{\mathcal{X}}^2 L_f}{\boldsymbol{\kappa}^2} + \frac{2\mathcal{D}_{\mathcal{X}}^2 L_A}{\beta_0\boldsymbol{\kappa}\sqrt{\boldsymbol{\kappa}}} \\ &\leq \left(1-\frac{2}{\boldsymbol{\kappa}}\right)\mathbb{E}\left[S_{\beta_{\boldsymbol{\kappa}-1}}(x_{\boldsymbol{\kappa}})\right] + \frac{1}{\boldsymbol{\kappa}^{3/2}}\left(2\mathcal{D}_{\mathcal{X}}^2 L_f + \frac{2\mathcal{D}_{\mathcal{X}}^2 L_A}{\beta_0} + 2\mathcal{D}_{\mathcal{X}}\sqrt{16L_f^2\mathcal{D}_{\mathcal{X}}^2 + 2L_A^2\mathcal{D}_{\mathcal{X}}^2\left(\frac{98}{\beta_0^2}+1\right) + 2\beta_0^2\sigma_f^2}\right)\end{aligned}$$

We can now apply Lemma B.2, with $b = 2\mathcal{D}_{\mathcal{X}}^2 L_f + \frac{2\mathcal{D}_{\mathcal{X}}^2 L_A}{\beta_0} + 2\mathcal{D}_{\mathcal{X}}\sqrt{16L_f^2\mathcal{D}_{\mathcal{X}}^2 + 2L_A^2\mathcal{D}_{\mathcal{X}}^2\left(\frac{98}{\beta_0^2}+1\right) + 2\beta_0^2\sigma_f^2}$, $c=2$, $\alpha=1$, $\beta=3/2$, and $C_4 = \max\left\{S_{\beta_{1,0}}(x_{1,1}),\ 2\mathcal{D}_{\mathcal{X}}^2 L_f + \frac{2\mathcal{D}_{\mathcal{X}}^2 L_A}{\beta_0} + 2\mathcal{D}_{\mathcal{X}}\sqrt{16L_f^2\mathcal{D}_{\mathcal{X}}^2 + 2L_A^2\mathcal{D}_{\mathcal{X}}^2\left(\frac{98}{\beta_0^2}+1\right) + 2\beta_0^2\sigma_f^2}\right\}$ to get

$$\mathbb{E}\left[S_{\beta_{t,k}}(x_{1,k+1})\right] \leq \frac{C_4}{\sqrt{K_t+k+1}} \quad \square$$

Corollary 4.2. *The expected convergence in terms of objective suboptimality and feasibility of Algorithm 2 is, respectively,*

$$\begin{aligned}\mathbb{E}\left[|f(x_{t,k}) - f(x^*)|\right] &\in \mathcal{O}\left((K_t+k)^{-1/2}\right)\\ \sqrt{\mathbb{E}\left[\operatorname{dist}(A(\xi)x_{t,k}, b(\xi))^2\right]} &\in \mathcal{O}\left((K_t+k)^{-1/2}\right)\end{aligned}$$

for both the finite-sum and the general expectation setting, up to constants. Consequently, the oracle complexity is given by $\#(ifo) \in \mathcal{O}\left(n\log_2(\epsilon^{-}2) + \epsilon^{-4}\right)$ and $\#(lmo) \in \mathcal{O}\left(\epsilon^{-2}\right)$ for the finite-sum setting, and by $\#(sfo) \in \mathcal{O}\left(\epsilon^{-4}\right)$ and $\#(lmo) \in \mathcal{O}\left(\epsilon^{-2}\right)$ for the more general expectation setting.

Proof A simple application of Lemma 3.1 in (Fercoq et al., 2019) for the previously derived convergence bounds of the smoothed gap, along with our chosen decrease rate for β yield the stated results.

For the oracle complexities, we choose a total number of outer loops T_ϵ in order to achieve a desired ϵ-accuracy.

$$\frac{1}{\sqrt{K_t+k}} \leq \epsilon \implies \frac{1}{\epsilon^2} \leq K_t + k \leq 2^t \implies T_\epsilon \geq \log_2\left(\epsilon^{-2}\right)$$

We can now state the corresponding complexity in terms of $\#(ifo)$ and $\#(lmo)$ for the finite-sum case of Algorithm 2:

$$\begin{aligned}\#(ifo) &= \sum_{t=1}^{T_\epsilon}\left(n + \sum_{k=2}^{K_t} K_t\right)\\ &= \sum_{t=1}^{T_\epsilon}\left(n + 2^{2(t-1)}\right)\\ &= nT_\epsilon + \mathcal{O}\left(2^{2T_\epsilon}\right) \in \mathcal{O}\left(\epsilon^{-4}\right)\end{aligned}$$

$$\#(lmo) = \sum_{t=1}^{T_\epsilon} K_t \leq 2K_{T_\epsilon} = 2^{T_\epsilon} \in \mathcal{O}\left(\epsilon^{-2}\right)$$

For the general expectation case, following the same steps, we get:

$$\begin{aligned}\#(sfo) &= \sum_{t=1}^{T_\epsilon}\left(|\mathcal{Q}_t| + \sum_{k=2}^{K_t} K_t\right)\\ &= \sum_{t=1}^{T_\epsilon}\left(\left\lceil\frac{2K_t}{\beta_{t,1}^2}\right\rceil + 2^{2(t-1)}\right)\\ &\leq \sum_{t=1}^{T_\epsilon}\left(\frac{2K_t}{\beta_{t,1}^2} + 1 + 2^{2(t-1)}\right)\\ &= \sum_{t=1}^{T_\epsilon}\left(\frac{2^t(2^{t-1}+1)}{\beta_0^2} + 1 + 2^{2(t-1)}\right)\\ &= \underbrace{\frac{1}{\beta_0^2}\sum_{t=1}^{T_\epsilon} 2^{2t-1}}_{\substack{\in\mathcal{O}\left(2^{2T_\epsilon}\right)\\ \equiv\mathcal{O}\left(\epsilon^{-4}\right)}} + \underbrace{\frac{1}{\beta_0^2}\sum_{t=1}^{T_\epsilon} 2^t}_{\substack{\in\mathcal{O}\left(2^{T_\epsilon}\right)\\ \equiv\mathcal{O}\left(\epsilon^{-2}\right)}} + \underbrace{T_\epsilon}_{\in\mathcal{O}(\log_2(\epsilon^{-2}))} + \underbrace{\sum_{t=1}^{T_\epsilon} 2^{2(t-1)}}_{\substack{\in\mathcal{O}\left(2^{2T_\epsilon}\right)\\ \equiv\mathcal{O}\left(\epsilon^{-4}\right)}}\\ &\in \mathcal{O}\left(\epsilon^{-4}\right)\end{aligned}$$

$$\#(lmo) = \sum_{t=1}^{T_\epsilon} K_t \leq 2K_{T_\epsilon} = 2^{T_\epsilon} \in \mathcal{O}\left(\epsilon^{-2}\right) \quad \square$$