A Unified Analysis of Stochastic Optimization Methods Using Jump System Theory and Quadratic Constraints

Hu, Bin; Seiler, Peter; Rantzer, Anders

Published in:
Proceedings of Machine Learning Research

2017

Document Version:
Publisher's PDF, also known as Version of record

Proceedings of Machine Learning Research vol 65:1–33, 2017

A Unified Analysis of Stochastic Optimization Methods Using Jump System Theory and Quadratic Constraints

Bin Hu BHU38@WISC.EDU
Wisconsin Institute for Discovery, University of Wisconsin, Madison, USA

Peter Seiler SEILE017@UMN.EDU
Department of Aerospace Engineering and Mechanics, University of Minnesota, Minneapolis, USA

Anders Rantzer RANTZER@CONTROL.LTH.SE
Department of Automatic Control, Lund University, Lund, Sweden

Abstract

We develop a simple routine unifying the analysis of several important recently-developed stochastic optimization methods including SAGA, Finito, and stochastic dual coordinate ascent (SDCA). First, we show an intrinsic connection between stochastic optimization methods and dynamic jump systems, and propose a general jump system model for stochastic optimization methods. Our proposed model recovers SAGA, SDCA, Finito, and SAG as special cases. Then we combine jump system theory with several simple quadratic inequalities to derive sufficient conditions for convergence rate certifications of the proposed jump system model under various assumptions (with or without individual convexity, etc). The derived conditions are linear matrix inequalities (LMIs) whose size roughly scale with the size of the training set. We make use of the symmetry in the stochastic optimization methods and reduce these LMIs to some equivalent small LMIs whose sizes are at most 3×3. We solve these small LMIs to provide analytical proofs of new convergence rates for SAGA, Finito and SDCA (with or without individual convexity). We also explain why our proposed LMI fails in analyzing SAG. We reveal a key difference between SAG and other methods, and briefly discuss how to extend our LMI analysis for SAG. An advantage of our approach is that the proposed analysis can be automated for a large class of stochastic methods under various assumptions (with or without individual convexity, etc).

Keywords: Empirical risk minimization, SAGA, Finito, SDCA, SAG, semidefinite programming, jump systems, quadratic constraints, control theory

1. Introduction

Convergence proofs for optimization methods are typically derived in a case-by-case manner. It is an important task to develop more unifying analysis which can be automatically generalized for complicated algorithms. The aim of this paper is to develop a unified analysis routine for a class of recently-developed stochastic optimization methods used in empirical risk minimization. Consider the following finite sum minimization

$$\underset{x\in\mathbb{R}^p}{\text{minimize}} \;\; g(x) := \frac{1}{n}\sum_{i=1}^{n} f_i(x) \tag{1}$$

where $g : \mathbb{R}^p \to \mathbb{R}$ is the objective function. The framework of (1) is useful for empirical risk minimization problems, e.g. ℓ_2-regularized logistic regression problems (Teo et al., 2007).

A widely-used approach for solving (1) is the stochastic gradient (SG) method (Robbins and Monro, 1951; Bottou and LeCun, 2004). However, the SG method only linearly converges to some tolerance of the optimum of (1) given a well-chosen constant stepsize. If a diminishing stepsize is used, the SG method will converge to the optimum but at a sublinear rate.

More recently, a class of new stochastic optimization methods have been proposed based on the idea of gradient aggregation. These methods converge linearly to the optimum point while preserving the iteration cost of the SG method. This family of gradient aggregation methods include SAG (Roux et al., 2012; Schmidt et al., 2013), SAGA (Defazio et al., 2014a), Finito (Defazio et al., 2014b), SDCA (Shalev-Shwartz and Zhang, 2013; Shalev-Shwartz, 2016) and SVRG (Johnson and Zhang, 2013). Existing linear rate bounds of SAG, SAGA, Finito, SDCA and SVRG are derived in a case-by-case manner. Moreover, the existing rate results for SAG, SAGA and Finito require the individual convexity of f_i. It is beneficial to develop a unified analysis framework which can be used to justify the existing rate results and obtain new rate bounds under various conditions (with or without individual convexity, etc).

Recently, semidefinite programs have been used to certify the performance of deterministic optimization methods (Drori and Teboulle, 2014; Kim and Fessler, 2016; Lessard et al., 2016; Nishihara et al., 2015; Taylor et al., 2017). Specifically, Lessard et al. (2016) provides a general analysis for deterministic first-order optimization methods (full gradient method, Nesterov's method, heavy ball method, etc) by adapting the integral quadratic constraint (IQC) framework (Megretski and Rantzer, 1997) from control theory. The key insight there is that the deterministic first-order methods can be viewed as interconnections of a linear time-invariant (LTI) dynamic system and a nonlinearity. Then quadratic inequalities can be used to characterize the nonlinearity and formulate LMI conditions.

In this paper, we present a unified analysis framework for a large class of stochastic optimization methods including SAGA, Finito and SDCA. Our approach here is inspired by the work of Lessard et al. (2016), and can be viewed as its stochastic extension. In our paper, the key insight is that many stochastic first-order methods can be viewed as an interconnection of a linear jump system and a static nonlinearity. Notice that a linear jump system is described by a linear state space model whose state matrices are functions of a jump parameter sampled from a given distribution. Since Lyapunov theory for jump systems has been well established in the controls field, we can incorporate quadratic constraints to obtain semidefinite programs for linear rate analysis of these stochastic optimization methods. Our main contributions are summarized as follows.

1. We present a unified jump system perspective on SAG, SAGA, Finito and SDCA. Specifically, we propose a general jump system model which governs the dynamics of a large family of stochastic methods including SAG, SAGA, Finito and SDCA.

2. We present a unified (and in some sense even automated) analysis framework for SAGA, Finito and SDCA using jump system perspectives and quadratic constraints. LMI conditions for a large class of stochastic methods under various conditions (with or without individual convexity, etc) are derived using one technique, and then solved to provide rate certificates.

3. We analytically solve the resultant LMIs to prove linear rate bounds for SAGA, Finito, and SDCA under different assumptions on g and f_i. Our results provide alternative proofs for many existing rate bounds. In addition, we prove that SAGA without individual convexity achieves an ϵ-optimal iteration complexity $\tilde{\mathcal{O}}\left((\frac{L^2}{m^2}+n)\log(\frac{1}{\epsilon})\right)$. We also prove Finito without individual convexity achieves an ϵ-optimal complexity of $\tilde{\mathcal{O}}\left(n\log(\frac{1}{\epsilon})\right)$ if $n \geq \frac{48L^2}{m^2}$.

4. Our quadratic constraint approach reveals a key difference between SAG and other methods. Specifically, SAGA, SDCA, and Finito only require simple quadratic inequalities used in this paper while SAG further requires more advanced quadratic inequalities to decode convexity. For this reason, the analysis of SAG is more involved, and our proposed LMI fails in analyzing SAG. We briefly sketch how to extend our LMI analysis for SAG. The extension requires incorporating more advanced quadratic inequalities into the LMI formulations.

The main advantage of our framework is its flexibility. The existing analysis for SAG, SAGA, Finito and SDCA is derived in a case-by-case manner. Our jump system framework provides a unified routine for analysis of such methods. Our analysis is highly repeatable and even "automated" in the sense that all LMI conditions are formulated using one technique and can be numerically solved to guide our analytical rate proof constructions. We emphasize that we view our LMI-based method as a complement rather than a replacement for existing proof techniques. One can always solve our proposed LMIs numerically and use the numerical results to narrow down possible Lyapunov function structures and useful function inequalities even before trying to construct proofs. This complements several existing proof techniques which more or less require guessing the required Lyapunov functions at the early stage of proof constructions. We will further explain this point after our main LMI condition is presented.

The rest of the paper is organized as follows. Section 2 introduces the notation and reviews the concepts of linear jump systems. In Section 3, we present a general jump system model which governs the dynamics of a large family of stochastic optimization methods including SAG, SAGA, Finito and SDCA. Section 4 presents a unified LMI analysis for the proposed jump system model. A unified LMI condition is derived using jump system theory and several function properties in the form of simple quadratic constraints. We apply the LMI condition and successfully prove various rate bounds for SAGA, SDCA, and Finito with or without individual convexity. We also explain why our proposed LMI fails in analyzing SAG. We reveal a key difference between SAG and other methods, and briefly discuss how to extend our LMI analysis for SAG. We present the main technical proofs in Section 5. Finally, we conclude with several future directions (Section 6).

2. Preliminaries

2.1. Notation and Background

The set of p-dimensional real vectors is denoted as $\mathbb{R}^p$. The $p \times p$ identity matrix and the $p \times p$ zero matrix are denoted as I_p and 0_p, respectively. The $n \times n$ identity matrix is denoted as I_n, and the $n \times n$ zero matrix is denoted as 0_n. Let e_i denote the n-dimensional vector whose entries are all 0 except the i-th entry which is 1. Let e denote the n-dimensional vector whose entries are all 1. Let $\tilde{0}$ denote the n-dimensional vector whose entries are all 0. For simplicity, 0 is occasionally used to denote a zero vector or a zero matrix when there is no confusion on the dimension. The Kronecker product of two matrices A and B is denoted by $A \otimes B$. Notice $(A \otimes B)^T = A^T \otimes B^T$ and $(A \otimes B)(C \otimes D) = (AC) \otimes (BD)$ when the matrices have compatible dimensions. When a matrix P is negative semidefinite (definite), we will use the notation $P \leq (<) 0$. When P is positive definite, we use the notation $P > 0$.

A continuously differentiable function $f : \mathbb{R}^p \to \mathbb{R}$ is L-smooth if for all $x, y \in \mathbb{R}^p$ we have $\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$. The continuously differentiable function f is said to be m-strongly convex if for all $x, y \in \mathbb{R}^p$ we have $f(x) \geq f(y) + \nabla f(y)^T(x - y) + \frac{m}{2}\|x - y\|^2$. Notice f is said

to be convex if f is 0-strongly convex. Let $\mathcal{F}(m, L)$ denote the set of continuously differentiable functions $f : \mathbb{R}^p \to \mathbb{R}$ that are L-smooth and m-strongly convex. Hence $\mathcal{F}(0, L)$ denotes the set of continuously differentiable convex functions that are L-smooth.

For any $f \in \mathcal{F}(m, L)$ with $m > 0$, there exist a unique $x^* \in \mathbb{R}^p$ such that $\nabla f(x^*) = 0$. In addition, the following inequality holds for any $x \in \mathbb{R}^p$ (Lessard et al., 2016, Proposition 5)

$$\begin{bmatrix} x - x^* \\ \nabla f(x) \end{bmatrix}^T \begin{bmatrix} -2mLI_p & (L+m)I_p \\ (L+m)I_p & -2I_p \end{bmatrix} \begin{bmatrix} x - x^* \\ \nabla f(x) \end{bmatrix} \geq 0 \tag{2}$$

However, a function satisfying the above inequality may not belong to $\mathcal{F}(m, L)$, and may not even be convex. The set of continuously differentiable functions satisfying (2) with some unique global minimum x^* is denoted as $\mathcal{S}(m, L)$. This class of functions has sector-bounded gradients, and includes $\mathcal{F}(m, L)$ as its subset. We emphasize that the functions in $\mathcal{S}(m, L)$ may not be convex.

A general assumption adopted in this paper is that $g \in \mathcal{S}(m, L)$ with $m > 0$. This is weaker than the assumption $g \in \mathcal{F}(m, L)$. Three sets of assumptions are typically used for f_i, i.e. $f_i \in \mathcal{F}(m, L)$, $f_i \in \mathcal{F}(0, L)$ or f_i being L-smooth. Given an arbitrary reference point x^* (the value of $\nabla f_i(x^*)$ may not be 0) and any $x \in \mathbb{R}^p$, the following inequality always holds

$$\begin{bmatrix} x - x^* \\ \nabla f_i(x) - \nabla f_i(x^*) \end{bmatrix}^T \begin{bmatrix} 2L\gamma I_p & (L-\gamma)I_p \\ (L-\gamma)I_p & -2I_p \end{bmatrix} \begin{bmatrix} x - x^* \\ \nabla f_i(x) - \nabla f_i(x^*) \end{bmatrix} \geq 0 \tag{3}$$

where γ is determined by the assumptions on f_i as follows

$$\gamma := \begin{cases} -m & \text{if } f_i \in \mathcal{F}(m, L) \\ 0 & \text{if } f_i \in \mathcal{F}(0, L) \\ L & \text{if } f_i \text{ is } L\text{-smooth} \end{cases}. \tag{4}$$

Notice (3) is just a summary of the definition of L-smoothness and the so-called co-coercivity condition (Lessard et al., 2016, Proposition 5).

Finally, the underlying probability space for the sampling index i_k is denoted as $(\Omega, \mathcal{F}, \mathbb{P})$. Let $\mathcal{F}_k$ be the σ-algebra generated by $(i_1, i_2, \ldots, i_k)$. Clearly, i_k is $\mathcal{F}_k$-adapted and we obtain a filtered probability space $(\Omega, \mathcal{F}, \{\mathcal{F}_k\}, \mathbb{P})$ which the stochastic method is defined on.

2.2. Stochastic Jump Systems

A linear jump system is described by the following set of recursive equations:

$$\begin{aligned} \xi^{k+1} &= A_{i_k}\xi^k + B_{i_k}w^k \\ v^k &= C_{i_k}\xi^k + D_{i_k}w^k. \end{aligned} \tag{5}$$

At each step k, the jump parameter i_k is a random variable taking value in a finite set $\mathcal{N} = \{1, \cdots, n\}$. In addition, $A_{i_k} : \mathcal{N} \to \mathbb{R}^{n_\xi \times n_\xi}$, $B_{i_k} : \mathcal{N} \to \mathbb{R}^{n_\xi \times n_w}$, $C_{i_k} : \mathcal{N} \to \mathbb{R}^{n_v \times n_\xi}$, and $D_{i_k} : \mathcal{N} \to \mathbb{R}^{n_v \times n_w}$ are functions of i_k. When $i_k = i \in \mathcal{N}$, clearly we have $A_{i_k} = A_i$, $B_{i_k} = B_i$, $C_{i_k} = C_i$, and $D_{i_k} = D_i$. If the process $\{i_k : k = 1, 2, \ldots\}$ is a Markov chain, the resultant jump system (5) is termed as a discrete-time Markovian jump linear system (MJLS). There is a large body of literature on MJLS in the controls field (Costa et al., 2006; Dragan et al., 2010). We confine our scope to the special case where i_k is an identically and independently distributed (IID) process, i.e.

$\mathbb{P}(i_k = i \,|\, \mathcal{F}_{k-1}) = \mathbb{P}(i_k = i)$ for all $k \geq 0$ and $i \in \mathcal{N}$. When i_k is sampled from a uniform distribution, we have $\mathbb{P}(i_k = i) = \frac{1}{n}$. When i_k is generated cyclically based on a deterministic order, (5) is not a jump system but a linear periodic system. There is also a large body of control literature on linear periodic systems (Bittanti and Colaneri, 2008). When i_k is a constant, then the state matrices are constant matrices and the model (5) is just an LTI system. LTI system theory is also well established (Hespanha, 2009).

3. A General Jump System Model for Stochastic Optimization Methods

Now we introduce the following general jump system model which governs the dynamics of a large family of stochastic optimization methods.

$$\begin{aligned} \xi^{k+1} &= A_{i_k}\xi^k + B_{i_k}w^k \\ v^k &= C\xi^k \\ w^k &= \begin{bmatrix} \nabla f_1(v^k) \\ \nabla f_2(v^k) \\ \vdots \\ \nabla f_n(v^k) \end{bmatrix} \end{aligned} \tag{6}$$

The above model builds upon the linear jump system model (5) by further enforcing a nonlinear relationship between w^k and v^k, i.e. $w^k = \left[\nabla f_1(v^k)^T \cdots \nabla f_n(v^k)^T\right]^T$. We can represent a large family of stochastic optimization methods using the unified jump system model (6) with properly chosen (A_{i_k}, B_{i_k}, C). In this paper, we consider the following stochastic methods.

1. SAGA (Defazio et al., 2014a): The iteration rule is the follows

$$x^{k+1} = x^k - \alpha\left(\nabla f_{i_k}(x^k) - y_{i_k}^k + \frac{1}{n}\sum_{i=1}^n y_i^k\right) \tag{7}$$

   where at each step k, a random training example i_k is drawn uniformly from the set $\mathcal{N}$ and

$$y_i^{k+1} := \begin{cases} \nabla f_i(x^k) & \text{if } i = i_k \\ y_i^k & \text{otherwise} \end{cases}. \tag{8}$$

2. SAG (Roux et al., 2012; Schmidt et al., 2013): The main iteration rule is

$$x^{k+1} = x^k - \alpha\left(\frac{\nabla f_{i_k}(x^k) - y_{i_k}^k}{n} + \frac{1}{n}\sum_{i=1}^n y_i^k\right) \tag{9}$$

   where at each k, i_k is uniformly drawn from the set $\mathcal{N}$ and y_i^k is updated by (8).

3. Finito (Defazio et al., 2014b): Suppose $x_i^k \in \mathbb{R}^p$ and $y_i^k \in \mathbb{R}^p$ for each k and all $i \in \mathcal{N}$. At each k, an index i_k is drawn from the set $\mathcal{N}$, and x_i^{k+1} is updated as

$$x_i^{k+1} := \begin{cases} \frac{1}{n}\sum_{i=1}^n x_i^k - \alpha\sum_{i=1}^n y_i^k & \text{if } i = i_k \\ x_i^k & \text{otherwise} \end{cases}. \tag{10}$$

where α is the stepsize [1]. Then y_i^{k+1} is updated as

$$y_i^{k+1} := \begin{cases} \nabla f_i(x_i^{k+1}) & \text{if } i = i_k \\ y_i^k & \text{otherwise} \end{cases} . \tag{11}$$

4. SDCA (Shalev-Shwartz, 2016, Algorithm 1): There are several versions of SDCA. For simplicity, we consider SDCA without duality, which solves the ℓ_2-regularized problem

$$\underset{x\in\mathbb{R}^p}{\text{minimize}} \;\; g(x) := \frac{1}{n}\sum_{i=1}^{n} f_i(x) + \frac{m}{2}\|x\|^2 \tag{12}$$

To solve the above problem, SDCA without duality requires updating $x^k \in \mathbb{R}^p$ and $y_i^{k+1} \in \mathbb{R}^p$ at each step. It first updates x^k using y_i^k as follows

$$x^k = \frac{1}{mn}\sum_{i=1}^{n} y_i^k \tag{13}$$

Then y_i^{k+1} is updated as

$$y_i^{k+1} := \begin{cases} y_i^k - \alpha mn(\nabla f_i(x^k) + y_i^k) & \text{if } i = i_k \\ y_i^k & \text{otherwise} \end{cases} . \tag{14}$$

where i_k is randomly sampled from $\mathcal{N}$. In the actual computation, the update (13) for $k \geq 1$ is performed using the formula $x^k = x^{k-1} - \alpha(\nabla f_{i_{k-1}}(x^{k-1}) + y_i^{k-1})$ due to efficiency considerations. However, (13) is more general and governs the updates of SDCA for all k.

To represent the above methods in the general jump system model (6), we can choose the state matrices as $A_{i_k} = \tilde{A}_{i_k} \otimes I_p$, $B_{i_k} = \tilde{B}_{i_k} \otimes I_p$, and $C = \tilde{C} \otimes I_p$ where $\tilde{A}_{i_k}$, $\tilde{B}_{i_k}$ and $\tilde{C}$ are defined according to Table 1.

Method	$\tilde{A}_{i_k}$	$\tilde{B}_{i_k}$	$\tilde{C}$
SAGA	$\begin{bmatrix} I_n - e_{i_k}e_{i_k}^T & \tilde{0} \\ -\frac{\alpha}{n}(e - ne_{i_k})^T & 1 \end{bmatrix}$	$\begin{bmatrix} e_{i_k}e_{i_k}^T \\ -\alpha e_{i_k}^T \end{bmatrix}$	$\begin{bmatrix} \tilde{0}^T & 1 \end{bmatrix}$
SAG	$\begin{bmatrix} I_n - e_{i_k}e_{i_k}^T & \tilde{0} \\ -\frac{\alpha}{n}(e - e_{i_k})^T & 1 \end{bmatrix}$	$\begin{bmatrix} e_{i_k}e_{i_k}^T \\ -\frac{\alpha}{n}e_{i_k}^T \end{bmatrix}$	$\begin{bmatrix} \tilde{0}^T & 1 \end{bmatrix}$
Finito	$\begin{bmatrix} I_n - e_{i_k}e_{i_k}^T & \tilde{0} \\ -\alpha(e_{i_k}e^T) & I_n - e_{i_k}(e_{i_k}^T - \frac{1}{n}e^T) \end{bmatrix}$	$\begin{bmatrix} e_{i_k}e_{i_k}^T \\ \tilde{0}\tilde{0}^T \end{bmatrix}$	$\begin{bmatrix} -\alpha e^T & \frac{1}{n}e^T \end{bmatrix}$
SDCA	$I_n - \alpha mne_{i_k}e_{i_k}^T$	$-\alpha mne_{i_k}e_{i_k}^T$	$\frac{1}{mn}e^T$

Table 1: State Matrices for Feedback Representations of SAG, SAGA, Finito, and SDCA

For illustrative purposes, we explain the jump system formulation for SAGA. The jump system formulations for SAG, Finito, and SDCA are further explained in Appendix A. For SAGA, we

1. One typical choice of α under the big data condition is $\alpha = \frac{1}{2nm}$.

define the stacked vector $y^k := \begin{bmatrix}(y_1^k)^T & \cdots & (y_n^k)^T\end{bmatrix}^T$. Then the SAGA gradient update rule (8) can be rewritten as:

$$y^{k+1} = \left((I_n - e_{i_k}e_{i_k}^T) \otimes I_p\right) y^k + \left((e_{i_k}e_{i_k}^T) \otimes I_p\right) w^k \tag{15}$$

where $w^k = \begin{bmatrix}\nabla f_1(x^k)^T \cdots \nabla f_n(x^k)^T\end{bmatrix}^T$. Notice $\sum_{i=1}^n y_i^k = (e^T \otimes I_p)y^k$ and $\nabla f_{i_k}(x^k) - y_{i_k}^k = (e_{i_k}^T \otimes I_p)(w^k - y^k)$. Thus the iteration rule (7) can be rewritten as follows:

$$\begin{aligned} x^{k+1} &= x^k - \alpha(e_{i_k}^T \otimes I_p)(w^k - y^k) - \frac{\alpha}{n}(e^T \otimes I_p)y^k \\ &= x^k - \frac{\alpha}{n}\left((e - ne_{i_k})^T \otimes I_p\right) y^k - \alpha(e_{i_k}^T \otimes I_p)w^k \end{aligned} \tag{16}$$

Now the update rules in (15) and (16) can be expressed as:

$$\begin{aligned} \begin{bmatrix} y^{k+1} \\ x^{k+1} \end{bmatrix} &= \begin{bmatrix} (I_n - e_{i_k}e_{i_k}^T) \otimes I_p & \tilde{0} \otimes I_p \\ -\frac{\alpha}{n}(e - ne_{i_k})^T \otimes I_p & I_p \end{bmatrix} \begin{bmatrix} y^k \\ x^k \end{bmatrix} + \begin{bmatrix} (e_{i_k}e_{i_k}^T) \otimes I_p \\ (-\alpha e_{i_k}^T) \otimes I_p \end{bmatrix} w^k \\ v^k &= \begin{bmatrix} \tilde{0}^T \otimes I_p & I_p \end{bmatrix} \begin{bmatrix} y^k \\ x^k \end{bmatrix} \\ w^k &= \begin{bmatrix} \nabla f_1(v^k) \\ \vdots \\ \nabla f_n(v^k) \end{bmatrix} \end{aligned} \tag{17}$$

which is exactly in the form of the general jump system model (6) with $\xi^k = \begin{bmatrix} y^k \\ x^k \end{bmatrix}$.

The computation of w^k at each k requires a full gradient computation (or n individual oracle accesses). However, B_{i_k} is sparse such that $B_{i_k}w^k$ only involves one individual oracle access. The low per-iteration cost of stochastic methods is captured by the sparsity of B_{i_k}. Most entries of w^k are "phantom" iterates which facilitates our analysis but do not appear in the actual computation.

Since $g \in \mathcal{S}(m, L)$ with $m > 0$, there exists unique $x^* \in \mathbb{R}^p$ satisfying $\nabla g(x^*) = 0$. To make (6) a good model for optimization methods, we have to ensure its equilibrium point is related to x^*. Define $w^* := \begin{bmatrix}\nabla f_1(x^*)^T & \ldots & \nabla f_n(x^*)^T\end{bmatrix}^T$, and $v^* := x^*$. If (6) is an optimization method which converges to x^*, then ξ^k should converge to some equilibrium state ξ^* capturing the information of x^* and satisfying

$$\begin{aligned} \xi^* &= A_i\xi^* + B_i w^* \\ v^* &= C\xi^* \\ w^* &= \begin{bmatrix} \nabla f_1(v^*) \\ \vdots \\ \nabla f_n(v^*) \end{bmatrix} \end{aligned} \tag{18}$$

for all $i \in \mathcal{N}$. Now we set up ξ^* for SAGA, SAG, Finito, and SDCA as follows.

1. For SAG and SAGA, we have $\xi^k := \begin{bmatrix} y^k \\ x^k \end{bmatrix}$ and $\xi^* := \begin{bmatrix} w^* \\ x^* \end{bmatrix}$. If we can show that ξ^k converges to ξ^*, then we can conclude that x^k converges to x^* and y_i^k converges to $\nabla f_i(x^*)$.

2. For Finito, we have $x^k := \begin{bmatrix}(x_1^k)^T & \cdots & (x_n^k)^T\end{bmatrix}^T$, $\xi^* = \begin{bmatrix} y^k \\ x^k \end{bmatrix}$ and $\xi^* = \begin{bmatrix} w^* \\ e \otimes x^* \end{bmatrix}$. If we can show that ξ^k converges to ξ^*, then y_i^k converges to $\nabla f_i(x^*)$ and x_i^k converges to x^*.

3. For SDCA (without duality), we have $\xi^k = y^k$ and $\xi^* = -w^*$. For the ℓ_2-regularized problem (12) with strongly-convex g, the optimal point x^* satisfies $mx^* + \frac{1}{n}\sum_{i=1}^n \nabla f_i(x^*) = 0$. Hence, if ξ^k converges to ξ^*, then y_i^k converges to $-\nabla f_i(x^*)$ and x^k converges to x^*.

It is straightforward to verify that (18) holds for the above ξ^* due to the fact $\nabla g(x^*) = 0$.

4. Analysis of Stochastic Methods Using Semidefinite Programs

4.1. An Unified LMI Condition for Analysis of Stochastic Methods

From the above discussion, we always want to show ξ^k converges to ξ^* at a given linear rate ρ. Now we present a unified LMI condition for such linear convergence using jump system theory and the basic quadratic inequalities (2) (3) which capture the key properties of the loss functions.

Theorem 1 *Consider the general jump system model (6), where $A_i = \tilde{A}_i \otimes I_p$, $B_i = \tilde{B}_i \otimes I_p$, and $C = \tilde{C} \otimes I_p$. Assume i_k is sampled in an IID manner from a uniform distribution $\mathbb{P}(i_k = i) = \frac{1}{n}$. Suppose there exists a unique $x^* \in \mathbb{R}^p$ such that $\nabla g(x^*) = 0$. The function f_i is assumed to satisfy the following two inequalities for any $x \in \mathbb{R}^p$,*

$$\begin{bmatrix} x - x^* \\ \frac{\sum_{i=1}^n \nabla f_i(x)}{n} - \frac{\sum_{i=1}^n \nabla f_i(x^*)}{n} \end{bmatrix}^T \begin{bmatrix} 2L\nu I_p & (L-\nu) I_p \\ (L-\nu) I_p & -2 I_p \end{bmatrix} \begin{bmatrix} x - x^* \\ \frac{\sum_{i=1}^n \nabla f_i(x)}{n} - \frac{\sum_{i=1}^n \nabla f_i(x^*)}{n} \end{bmatrix} \geq 0 \tag{19}$$

$$\begin{bmatrix} x - x^* \\ \nabla f_i(x) - f_i(x^*) \end{bmatrix}^T \begin{bmatrix} 2L\gamma I_p & (L-\gamma) I_p \\ (L-\gamma) I_p & -2 I_p \end{bmatrix} \begin{bmatrix} x - x^* \\ \nabla f_i(x) - f_i(x^*) \end{bmatrix} \geq 0 \tag{20}$$

where ν and γ are some prescribed scalars. Define $\tilde{D}_{\psi 1} \in \mathbb{R}^{2n+2}$ and $\tilde{D}_{\psi 2} \in \mathbb{R}^{(2n+2)\times n}$ as

$$\begin{aligned} \tilde{D}_{\psi 1} &= \begin{bmatrix} L & \nu & L & \gamma & \dots & L & \gamma \end{bmatrix}^T \\ \tilde{D}_{\psi 2} &= \begin{bmatrix} -\frac{1}{n} e & \frac{1}{n} e & -e_1 & e_1 & \dots & -e_n & e_n \end{bmatrix}^T . \end{aligned} \tag{21}$$

If $\exists$ an $n_\xi \times n_\xi$ matrix $\tilde{P} = \tilde{P}^T > 0$ and nonnegative scalars λ_1, λ_2 such that

$$\begin{aligned} &\begin{bmatrix} \frac{1}{n}\sum_{i=1}^n \tilde{A}_i^T \tilde{P} \tilde{A}_i - \rho^2 \tilde{P} & \frac{1}{n}\sum_{i=1}^n \tilde{A}_i^T \tilde{P} \tilde{B}_i \\ \frac{1}{n}\sum_{i=1}^n \tilde{B}_i^T \tilde{P} \tilde{A}_i & \frac{1}{n}\sum_{i=1}^n \tilde{B}_i^T \tilde{P} \tilde{B}_i \end{bmatrix} + \\ &\qquad \begin{bmatrix} \tilde{C}^T \tilde{D}_{\psi 1}^T \\ \tilde{D}_{\psi 2}^T \end{bmatrix} \left(\begin{bmatrix} \lambda_1 & \tilde{0}^T \\ \tilde{0} & \frac{\lambda_2}{n} I_n \end{bmatrix} \otimes \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \right) \begin{bmatrix} \tilde{D}_{\psi 1} \tilde{C} & \tilde{D}_{\psi 2} \end{bmatrix} \leq 0 \end{aligned} \tag{22}$$

then all $k \geq 1$ and $\xi^0 \in \mathbb{R}^{n_\xi}$, the following inequality holds

$$\mathbb{E}\left[(\xi^{k+1} - \xi^*)^T (\tilde{P} \otimes I_p)(\xi^{k+1} - \xi^*) \right] \leq \rho^2 \mathbb{E}\left[(\xi^k - \xi^*)^T (\tilde{P} \otimes I_p)(\xi^k - \xi^*) \right]. \tag{23}$$

Consequently, $\mathbb{E}\left[\|\xi^k - \xi^*\|^2\right] \leq \rho^{2k} \left(\text{cond}(\tilde{P}) \|\xi^0 - \xi^*\|^2 \right)$ holds for all $k \geq 1$ and $\xi^0 \in \mathbb{R}^{n_\xi}$, where cond *denotes the condition number of a given positive definite matrix.*

Proof A detailed proof is presented in Section 5.1. Here we briefly sketch the proof idea. Denote $P = \tilde{P} \otimes I_p$, and define a Lyapunov function by $V(\xi^k) = (\xi^k - \xi^*)^T P(\xi^k - \xi^*)$. Then one can use the LMI condition and the basic quadratic inequalities (19) (20) to show that V satisfies $\mathbb{E}V(\xi^{k+1}) - \rho^2 \mathbb{E}V(\xi^k) \leq 0$. This immediately leads to the desired conclusion. We can see the LMI condition gives us an automated way to search quadratic Lyapunov functions. ■

The initial condition $\|\xi^0 - \xi^*\|^2$ is related to the so-called variance term since ξ^* is typically determined by x^* and $\nabla f_i(x^*)$. When ρ^2 is given, the testing condition (22) is linear in $\tilde{P}$, λ_1, and λ_2. Therefore, (22) is an LMI whose feasible set is convex and can be effectively searched using the state-of-the-art convex optimization techniques, e.g. interior point method. Many optimization solvers are available such that coding this LMI condition is a straightforward task.

One can automate the proposed LMI analysis of stochastic optimization methods by modifying the values of ν and γ to reflect various assumptions on g and f_i. For SAG, SAGA and Finito, we always assume $g \in \mathcal{S}(m, L)$ with $m > 0$ and hence we should set $\nu = -m$ in our analysis. The value of γ is chosen based on the assumptions on f_i as follows.

$$\gamma = \begin{cases} -m & \text{if } f_i \in \mathcal{F}(m, L) \\ 0 & \text{if } f_i \in \mathcal{F}(0, L) \\ L & \text{if } f_i \text{ is } L\text{-smooth} \end{cases}$$

For SDCA, (12) is considered. We assume $\frac{1}{n}\sum_{i=1}^n f_i \in \mathcal{F}(0, L)$. By co-coercivity, we can set $\nu = 0$. In addition, we have $\gamma = 0$ if $f_i \in \mathcal{F}(0, L)$ and $\gamma = L$ if f_i is only assumed to be L-smooth.

4.2. Numerical Pre-analysis of Stochastic Methods Using Semidefinite Programs

Theorem 1 provides a simple unified tool for linear rate analysis of stochastic optimization methods governed by the general jump system model (6). In principle, one can implement LMI (22) once. Then given a stochastic method (6), one only needs to modify the $(\tilde{A}_i, \tilde{B}_i, \tilde{C})$ matrices in the code. Notice the size of the LMI condition (22) scales proportionally with n, and hence we can only solve LMI (22) numerically for n up to several hundred. However, these numerical results with n being several hundred provide informative clues for further proof constructions. Notice the following two questions are important when analyzing a finite-sum method using Lyapunov arguments:

1. Which inequalities describing the function properties should be used in the proof?
2. What is the simplest form of Lyapunov function required by the proof?

Answers to these questions in the early stage of the analysis can guide researchers in their search for proofs. Usually one has to make a rough guess based on personal expertise. Theorem 1 provides a complementary numerical tool for this purpose. The numerical feasibility results from LMI (22) with n being several hundred roughly answer the questions above by providing clues for selecting related function inequalities and simplified forms of Lyapunov functions. For example, numerical tests of LMI (22) for SAGA show that enforcing the Lyapunov function to be diagonal does not change the feasibility results. This suggests using a diagonal Lyapunov function for SAGA. When analyzing Finito, the numerical tests of (22) immediately indicate that Finito requires Lyapunov functions with off-diagonal terms. When we test the existing rate results for SAG (Schmidt et al., 2013, Theorem 1), LMI (22) becomes infeasible. This indicates that the analysis of SAG requires less conservative function inequalities in addition to the simple quadratic inequalities (19) (20). The

details of the numerical tests of LMI (22) are presented in Appendix B. Notice our proposed analysis heavily relies on the quadratic constraints used in the LMI formulations. Some stochastic methods, e.g. SAGA, SDCA and Finito, are relatively easier to analyze, since they only require the simple quadratic inequalities (19) (20). Some other methods, e.g. SAG, are more involved, and require more advanced quadratic constraints in addition to (19) (20). Theorem 1 provides a simple tool to distinguish these two classes of stochastic methods. We will further discuss SAG in Section 4.5. Next, we reduce LMI (22) to some equivalent small LMIs for SAGA, Finito, and SDCA.

4.3. Dimension Reduction for the Proposed LMI

The preliminary numerical test results of LMI (22) actually shed light on possible simplifications of the proposed LMI condition. Based on the preliminary numerical tests documented in Appendix B, it seems that (22) is sufficient for analysis of SAGA, Finito, and SDCA. As mentioned before, we notice various simplified parameterizations of $\tilde{P}$ are required for different algorithms. These simplified parameterizations seem not to introduce further conservatism for our analysis. The resultant LMI (22) with such $\tilde{P}$ consists of blocks which have the special form $\mu I_n + qee^T$ where μ and q are some scalars. We summarize our preliminary findings in Table 2.

Method	Parameterization of $\tilde{P}$	Matrix Form of the Resultant LMI (22)
SAGA	$\begin{bmatrix} p_1 I_n & \tilde{0} \\ \tilde{0}^T & p_2 \end{bmatrix}$	$\begin{bmatrix} \mu_1 I_n + q_1 ee^T & q_4 e & \mu_6 I_n + q_6 ee^T \\ q_4 e^T & \mu_2 & q_5 e^T \\ \mu_6 I_n + q_6 ee^T & q_5 e & \mu_3 I_n + q_3 ee^T \end{bmatrix}$
SDCA	$p_1 I_n + p_2 ee^T$	$\begin{bmatrix} \mu_1 I_n + q_1 ee^T & \mu_3 I_n + q_3 ee^T \\ \mu_3 I_n + q_3 ee^T & \mu_2 I_n + q_2 ee^T \end{bmatrix}$
Finito	$\begin{bmatrix} p_1 I_n + p_2 ee^T & p_3 ee^T \\ p_3 ee^T & p_4 I_n + p_5 ee^T \end{bmatrix}$	$\begin{bmatrix} \mu_1 I_n + q_1 ee^T & \mu_4 I_n + q_4 ee^T & \mu_6 I_n + q_6 ee^T \\ \mu_4 I_n + q_4 ee^T & \mu_2 I_n + q_2 ee^T & \mu_5 I_n + q_5 ee^T \\ \mu_6 I_n + q_6 ee^T & \mu_5 I_n + q_5 ee^T & \mu_3 I_n + q_3 ee^T \end{bmatrix}$

Table 2: Parameterization of $\tilde{P}$ and Matrix Forms in (22) for SAGA, SDCA and Finito

The special matrix form of (22) is due to the same assumption on f_i for all i and the uniform sampling of i_k. We can take advantage of the special matrix forms and convert (22) into equivalent small LMIs whose sizes do not depend on n. For example, we know $\begin{bmatrix} \mu_1 I_n + q_1 ee^T & \mu_3 I_n + q_3 ee^T \\ \mu_3 I_n + q_3 ee^T & \mu_2 I_n + q_2 ee^T \end{bmatrix} \leq 0$ if and only if $\begin{bmatrix} \mu_1 & \mu_3 \\ \mu_3 & \mu_2 \end{bmatrix} \leq 0$ and $\begin{bmatrix} \mu_1 & \mu_3 \\ \mu_3 & \mu_2 \end{bmatrix} + n \begin{bmatrix} q_1 & q_3 \\ q_3 & q_2 \end{bmatrix} \leq 0$. Hence the analysis of SDCA actually involves two coupled 2×2 LMIs. Similar linear algebra tricks can be used to convert (22) into equivalent small LMIs for SAGA and Finito. This leads to the following simplified testing conditions.

Theorem 2 *Suppose i_k is uniformly sampled and $m > 0$. Let a testing rate $0 \leq \rho \leq 1$ be given.*

1. *(SAGA): Suppose $g \in \mathcal{S}(m, L)$, and γ is defined by (4) based on assumptions on f_i. If there exist positive scalars p_1, p_2, and non-negative scalars λ_1, λ_2 such that*

$$\begin{bmatrix} p_2\alpha^2 + \left(\frac{n-1}{n} - \rho^2\right) np_1 & -\alpha^2 p_2 \\ -\alpha^2 p_2 & p_1 + \alpha^2 p_2 - 2\lambda_2 \end{bmatrix} \leq 0 \tag{24}$$

$$\begin{bmatrix} (1-\rho^2)p_2 - 2\lambda_1 mL + 2\lambda_2 L\gamma & -\alpha p_2 + (m+L)\lambda_1 + (L-\gamma)\lambda_2 \\ -\alpha p_2 + (m+L)\lambda_1 + (L-\gamma)\lambda_2 & p_1 + \alpha^2 p_2 - 2\lambda_2 - 2\lambda_1 \end{bmatrix} \leq 0 \tag{25}$$

Then SAGA (7) (8) initialized with any $x^0 \in \mathbb{R}^p$ and $y_i^0 \in \mathbb{R}^p$ satisfies

$$\mathbb{E}\left[\|x^k - x^*\|^2 + \frac{p_1}{p_2}\sum_{i=1}^n \|y_i^k - \nabla f_i(x^*)\|^2\right] \leq \rho^{2k} R^0 \tag{26}$$

where $R^0 = \|x^0 - x^*\|^2 + \frac{p_1}{p_2}\sum_{i=1}^n \|y_i^0 - \nabla f_i(x^*)\|^2$.

2. *(Finito): Suppose $g \in \mathcal{S}(m, L)$, and γ is defined by (4) based on assumptions on f_i. If there exist scalars p_1, p_2, p_3, p_4, p_5 and non-negative scalars λ_1, λ_2 such that $p_1 > 0$, $p_4 > 0$ and*

$$\begin{bmatrix} p_1 + np_2 & np_3 \\ np_3 & p_4 + np_4 \end{bmatrix} > 0 \tag{27}$$

$$\begin{bmatrix} p_2 - p_1 + n(1-\rho^2)p_1 & p_3 & -p_2 \\ p_3 & p_5 - p_4 + n(1-\rho^2)p_4 & -p_3 \\ -p_2 & -p_3 & p_1 + p_2 - 2\lambda_2 \end{bmatrix} \leq 0 \tag{28}$$

$$\begin{bmatrix} X_{11} & X_{12} & X_{13} \\ X_{12} & (p_4 + np_5)(1-\rho^2) - \frac{2Lm\lambda_1 - 2L\gamma\lambda_2}{n} & p_3 + \frac{(L+m)\lambda_1 + (L-\gamma)\lambda_2}{n} \\ X_{13} & p_3 + \frac{(L+m)\lambda_1 + (L-\gamma)\lambda_2}{n} & \frac{p_1 + p_2 - 2\lambda_1 - 2\lambda_2}{n} \end{bmatrix} \leq 0 \tag{29}$$

$$\begin{aligned} X_{11} = {} & (1 - \frac{1}{n} - \rho^2)p_1 + \frac{p_2}{n} - n\rho^2 p_2 + (n-2)p_2 - 2(1 - \frac{1}{n})p_3\alpha n \\ & + (p_4 + p_5 - 2Lm\lambda_1 + 2L\gamma\lambda_2)\alpha^2 n \end{aligned} \tag{30}$$

$$X_{12} = (1-\rho^2)p_3 n - p_3 - (p_4 + np_5 - 2Lm\lambda_1 + 2L\gamma\lambda_2)\alpha \tag{31}$$

$$X_{13} = (1 - \frac{1}{n})p_2 - (p_3 + \lambda_1(L+m) + \lambda_2(L-\gamma))\alpha \tag{32}$$

Then Finito (10) (11) with any initial condition $x_i^0 \in \mathbb{R}^p$ and $y_i^0 \in \mathbb{R}^p$ satisfies

$$\mathbb{E}V^k \leq \rho^{2k} V^0 \tag{33}$$

where $V^k = (\xi^k - \xi^*)^T P(\xi^k - \xi^*)$, $\xi^k = \begin{bmatrix} y^k \\ x^k \end{bmatrix}$, $P = \begin{bmatrix} p_1 I_n + p_2 ee^T & p_3 ee^T \\ p_3 ee^T & p_4 I_n + p_5 ee^T \end{bmatrix} \otimes I_p$.

3. *(SDCA): Suppose $\frac{1}{n}\sum_{i=1}^n f_i \in \mathcal{F}(0, L)$. Set $\gamma = 0$ if $f_i \in \mathcal{F}(0, L)$, and set $\gamma = L$ if f_i is only L-smooth. Denote $\tilde{\alpha} = \alpha mn$. If there exist real scalars p_1, p_2 and nonnegative λ_1, λ_2 such that $p_1 > 0$, $p_1 + np_2 > 0$, and*

$$\begin{bmatrix} p_1(\tilde{\alpha}^2 - 2\tilde{\alpha} + n(1-\rho^2)) + p_2\tilde{\alpha}^2 & p_1(\tilde{\alpha}^2 - \tilde{\alpha}) + \tilde{\alpha}^2 p_2 \\ p_1(\tilde{\alpha}^2 - \tilde{\alpha}) + \tilde{\alpha}^2 p_2 & (p_1 + p_2)\tilde{\alpha}^2 - 2\lambda_2 \end{bmatrix} \leq 0 \tag{34}$$

$$\begin{bmatrix} X_{11} & X_{12} \\ X_{12} & (p_1 + p_2)\tilde{\alpha}^2 - 2(\lambda_1 + \lambda_2) \end{bmatrix} \leq 0 \tag{35}$$

$$X_{11} = p_1(\tilde{\alpha}^2 - 2\tilde{\alpha} + n(1-\rho^2)) + p_2(\tilde{\alpha} - n)^2 - n^2\rho^2 p_2 + \frac{2\gamma L\lambda_2}{m^2} \tag{36}$$

$$X_{12} = p_1(\tilde{\alpha}^2 - \tilde{\alpha}) + \tilde{\alpha}(\tilde{\alpha} - n)p_2 + \frac{\lambda_1 L + (L-\gamma)\lambda_2}{m} \tag{37}$$

Then SDCA (13) (14) with stepsize α and initial condition y_0^k satisfies

$$\mathbb{E}\left[\|x^k - x^*\|^2 + \frac{p_1}{p_2 m^2 n^2}\sum_{i=1}^n \|y_i^k + \nabla f_i(x^*)\|^2\right] \leq \rho^{2k} R^0 \tag{38}$$

where $R^0 = \|x^0 - x^*\|^2 + \frac{p_1}{p_2 m^2 n^2}\sum_{i=1}^n \|y_i^0 + \nabla f_i(x^*)\|^2$.

Proof One can compute analytical expressions of the matrix on the left side of (22) and prove this theorem using the linear algebra tricks mentioned before. Detailed proofs are left to Appendix C. ■

4.4. New Analytical Rate Bounds for SAGA, Finito, and SDCA

We can analytically solve the LMIs in Theorem 2, and prove the following rate results for SAGA, Finito and SDCA.

Corollary 3 *(Rate Bounds for SAGA) Assume i_k is uniformly sampled from $\mathcal{N}$, and $g \in \mathcal{S}(m, L)$ with $m > 0$. Consider SAGA (7) (8) initialized from $x^0 \in \mathbb{R}^p$ and $y_i^0 \in \mathbb{R}^p$.*

1. *If $f_i \in \mathcal{F}(m, L)$, then for any $0 < \alpha \leq \frac{1}{2L}$, one has*

$$\mathbb{E}\left[\|x^k - x^*\|^2\right] \leq \left(1 - \min\left\{\frac{2L\alpha - 1}{(L\alpha - 1)n}, 2m\alpha - \frac{\alpha m^2}{(1 - L\alpha)L}\right\}\right)^k R^0 \tag{39}$$

where $R^0 = \|x^0 - x^*\|^2 + \frac{\alpha}{L}\sum_{i=1}^n \|y_i^0 - \nabla f_i(x^*)\|^2$. The following bound also holds for any $\alpha \leq \frac{4}{9L}$

$$\mathbb{E}\left[\|x^k - x^*\|^2\right] \leq \left(1 - \min\left\{\frac{9L\alpha - 4}{(3L\alpha - 4)n}, 2m\alpha - \frac{3\alpha m^2}{(4 - 3L\alpha)L}\right\}\right)^k R^0 \tag{40}$$

where $R^0 = \|x^0 - x^*\|^2 + \frac{2\alpha}{3L}\sum_{i=1}^n \|y_i^0 - \nabla f_i(x^*)\|^2$.

2. *If $f_i \in \mathcal{F}(0, L)$, then for any $0 < \alpha \leq \frac{1}{2L}$, one has*

$$\mathbb{E}\left[\|x^k - x^*\|^2\right] \leq \left(1 - \min\left\{\frac{2L\alpha - b}{(L\alpha - b)n}, 2(1-b)m\alpha - \frac{\alpha m^2 (1-b)^2}{(2 - b - L\alpha)L}\right\}\right)^k R^0 \tag{41}$$

where b can be any scalar in $[2L\alpha, 1]$, and $R^0 = \|x^0 - x^*\|^2 + \frac{b\alpha}{L}\sum_{i=1}^n \|y_i^0 - \nabla f_i(x^*)\|^2$. More specifically, when $\alpha = \frac{1}{3L}$, we can set $b = \frac{5}{6}$ and get the following bound:

$$\mathbb{E}\left[\|x^k - x^*\|^2\right] \leq \left(1 - \min\left\{\frac{1}{3n}, \frac{m}{10L}\right\}\right)^k R^0 \tag{42}$$

where $R^0 = \|x^0 - x^*\|^2 + \frac{5}{18L^2}\sum_{i=1}^n \|y_i^0 - \nabla f_i(x^*)\|^2$.

3. *If f_i is only assumed to be L-smooth, then the following bound holds for any $\alpha \leq \frac{3m}{8L^2}$,*

$$\mathbb{E}\left[\|x^k - x^*\|^2\right] \leq \left(1 - \min\left\{\frac{b-2}{(b-1)n}, \frac{3m\alpha}{2} - 2bL^2\alpha^2\right\}\right)^k R^0 \tag{43}$$

where b can be any scalar satisfying $2 \leq b \leq \frac{3m}{4\alpha L^2}$, and $R^0 = \|x^0 - x^*\|^2 + b\alpha^2 \sum_{i=1}^n \|y_i^0 - \nabla f_i(x^*)\|^2$. Specifically, when $\alpha = \frac{m}{8L^2}$, we can set $b = 3$ and get the following bound:

$$\mathbb{E}\left[\|x^k - x^*\|^2\right] \leq \left(1 - \min\left\{\frac{1}{2n}, \frac{3m^2}{32L^2}\right\}\right)^k R^0 \tag{44}$$

where $R^0 = \|x^0 - x^*\|^2 + \frac{3m^2}{64L^4} \sum_{i=1}^n \|y_i^0 - \nabla f_i(x^*)\|^2$. When $\alpha = \frac{m}{4(m^2n+L^2)}$, we can set $b = \frac{2(m^2n+L^2)}{L^2}$ and obtain

$$\mathbb{E}\left[\|x^k - x^*\|^2\right] \leq \left(1 - \frac{m^2}{8(m^2n + L^2)}\right)^k R^0 \tag{45}$$

where $R^0 = \|x^0 - x^*\|^2 + \frac{m^2}{8(m^2n+L^2)L^2} \sum_{i=1}^n \|y_i^0 - \nabla f_i(x^*)\|^2$. Hence, the ϵ-optimal iteration complexity of SAGA without individual convexity is $\tilde{\mathcal{O}}\left((\frac{L^2}{m^2} + n)\log(\frac{1}{\epsilon})\right)$.

Corollary 4 *(Rate Bounds for Finito) Assume i_k is uniformly sampled from $\mathcal{N}$, and $g \in \mathcal{S}(m, L)$ with $m > 0$. Consider Finito (10) (11) initialized from $x_i^0 \in \mathbb{R}^p$ and $y_i^0 \in \mathbb{R}^p$. Define $v^k = \frac{1}{n}\sum_{i=1}^n x_i^k - \alpha \sum_{i=1}^n y_i^k$.*

1. *If $f_i \in \mathcal{F}(m, L)$ and $n \geq \sqrt{\frac{50L}{m}}$, then Finito with $\alpha = \frac{1}{5L}$ satisfies*

$$\mathbb{E}\left[\frac{m}{10L}\sum_{i=1}^n \|x_i^k - x^*\|^2 + \|v^0 - x^*\|^2\right] \leq \left(1 - \min\left\{\frac{1}{2n}, \frac{m}{20L}\right\}\right)^k R^0 \tag{46}$$

where $R^0 = \frac{m}{10L}\sum_{i=1}^n \|x_i^0 - x^*\|^2 + \frac{1}{5L^2}\sum_{i=1}^n \|y_i^0 - \nabla f_i(x^*)\|^2 + \|v^0 - x^*\|^2$.

2. *If $f_i \in \mathcal{F}(0, L)$ and $n \geq \sqrt{\frac{64L}{m}}$, then Finito with $\alpha = \frac{1}{8L}$ satisfies*

$$\mathbb{E}\left[\frac{m}{16L}\sum_{i=1}^n \|x_i^k - x^*\|^2 + \|v^0 - x^*\|^2\right] \leq \left(1 - \min\left\{\frac{1}{3n}, \frac{5m}{176L}\right\}\right)^k R^0 \tag{47}$$

where $R^0 = \frac{m}{16L}\sum_{i=1}^n \|x_i^0 - x^*\|^2 + \frac{1}{16L^2}\sum_{i=1}^n \|y_i^0 - \nabla f_i(x^*)\|^2 + \|v^0 - x^*\|^2$.

3. *If f_i is L-smooth and $n \geq \frac{48L^2}{m^2}$, then Finito with $\alpha = \frac{1}{2nm}$ satisfies*

$$\mathbb{E}\left[\frac{3}{8n}\sum_{i=1}^n \|x_i^k - x^*\|^2 + \|v^0 - x^*\|^2\right] \leq \left(1 - \frac{1}{3n}\right)^k R^0 \tag{48}$$

where $R^0 = \frac{3}{8n}\sum_{i=1}^n \|x_i^0 - x^*\|^2 + \frac{1}{n^2m^2}\sum_{i=1}^n \|y_i^0 - \nabla f_i(x^*)\|^2 + \|v^0 - x^*\|^2$.

Corollary 5 *(Rate Bounds for SDCA without Duality) Assume i_k is uniformly sampled from $\mathcal{N}$, and $\sum_{i=1}^n f_i \in \mathcal{F}(0, L)$. Consider SDCA (13) (14) initialized from y_i^0.*

1. *If $f_i \in \mathcal{F}(0, L)$, then for any $0 < \alpha \leq \frac{2}{L+2mn}$, one has*

$$\mathbb{E}\left[\|x^k - x^*\|^2 + \frac{\alpha}{(1-\alpha mn)mn}\sum_{i=1}^n \|y_i^k + \nabla f_i(x^*)\|^2\right] \leq (1-m\alpha)^k R^0 \tag{49}$$

   where $R^0 = \|x^0 - x^*\|^2 + \frac{\alpha}{(1-\alpha mn)mn}\sum_{i=1}^n \|y_i^0 + \nabla f_i(x^*)\|^2$.

2. *If f_i is L-smooth, then (49) holds for any $0 < \alpha \leq \frac{m}{L^2+m^2n}$. When $\alpha = \frac{m}{(m^2n+L^2)}$, the following bound holds*

$$\mathbb{E}\left[\|x^k - x^*\|^2 + \frac{1}{L^2 n}\sum_{i=1}^n \|y_i^k + \nabla f_i(x^*)\|^2\right] \leq \left(1 - \frac{m^2}{m^2n + L^2}\right)^k R^0 \tag{50}$$

   where $R^0 = \|x^0 - x^*\|^2 + \frac{1}{L^2 n}\sum_{i=1}^n \|y_i^0 + \nabla f_i(x^*)\|^2$.

All the proofs are presented in Section 5. All three corollaries are actually proved via analytically solving the LMI conditions in Theorem 2. When f_i is assumed to be only smooth (not necessarily convex), we only need to modify the value of γ to be L and then analytically construct a feasible solution for the resultant LMIs. We believe our rate bounds for SAGA and Finito without individual convexity (Statement 3 in Corollary 3 and Statement 3 in Corollary 4) are new. Now we briefly discuss the connections between our results and some existing rate bounds.

1. (SAGA) Statement 1 in Corollary 3 is new in the sense that it works for a range of α and also highlights the trade-off between the dependence of ρ^2 on n and $\frac{m}{L}$. Notice that (39) works better under the big data condition while (40) is less conservative with large condition number L/m. Suppose $f_i \in \mathcal{F}(m, L)$. If one chooses $\alpha = \frac{1}{3L}$ in (40) and applies the fact $L \geq m$, (40) directly leads to

$$\mathbb{E}\left[\|x^k - x^*\|^2\right] \leq \left(1 - \min\left\{\frac{1}{3n}, \frac{m}{3L}\right\}\right)^k \left(\|x^0 - x^*\|^2 + \frac{2}{9L^2}\sum_{i=1}^n \|y_i^0 - \nabla f_i(x^*)\|^2\right) \tag{51}$$

   The convergence rate in the above bound agrees with the result in Defazio et al. (2014a, Section 2). On the other hand, one can also choose $\alpha = \frac{1}{3L}$ in (39) and obtain

$$\mathbb{E}\left[\|x^k - x^*\|^2\right] \leq \left(1 - \min\left\{\frac{1}{2n}, \frac{m}{6L}\right\}\right)^k \left(\|x^0 - x^*\|^2 + \frac{1}{3L^2}\sum_{i=1}^n \|y_i^0 - \nabla f_i(x^*)\|^2\right) \tag{52}$$

   Clearly, the above bound is better than (51) under the big data condition $n \geq \frac{3L}{m}$. In principle, one can generate a family of bounds to describe this trade-off in more details. But all these bounds will only affect the iteration complexity $\tilde{\mathcal{O}}\left((n + \frac{L}{m})\log(\frac{1}{\epsilon})\right)$ by a constant factor.

Actually, we can also recover some other existing rate bounds for SAGA with individual convexity by modifying the proofs. See Remark 6 for further discussions.

We notice that for any fixed m and L, SAGA (with n sufficiently large) can achieve a rate $\rho^2 = 1 - \frac{1}{cn}$ where c is arbitrarily close to 1. For example, consider $f_i \in \mathcal{F}(m, L)$. Given any $c \in (1, \infty)$, we can choose a sufficiently small α to ensure $\frac{L\alpha - 1}{2L\alpha - 1} < c$. For this specific value of α, (39) just leads to a rate bound $\rho^2 = 1 - \frac{1}{cn}$ under the condition $\left(2m\alpha - \frac{\alpha m^2}{(1-L\alpha)L}\right) n \geq \frac{2L\alpha - 1}{L\alpha - 1}$. Similar arguments also work when $f_i \in \mathcal{F}(0, L)$ or f_i being L-smooth.

2. (Finito): When $f_i \in \mathcal{F}(m, L)$ with $m > 0$, our result states a linear rate bound for $\alpha = \frac{1}{5L}$, which is a stepsize independent of the parameter m. This could be useful since sometimes m is unknown for practical problems. On the other hand, the rate proofs in Defazio et al. (2014b, Theorem 1) work for $\alpha = \frac{1}{2nm}$ under the big data condition $n \geq \frac{2L}{m}$.

   In general, our rate bounds for Finito are not as good as the rate bounds for SAGA. This is due to the fact that the LMI conditions for Finito are more complicated and involve more decision variables. We are only able to analytically solve these LMIs under the big data condition, although our preliminary numerical tests on the feasibility of these LMIs suggest that Finito and SAGA have similar convergence rates.

3. (SDCA) Statement 1 in the above corollary is very similar to Shalev-Shwartz (2015, Theorem 1). Actually, when $\alpha = \frac{1}{L+mn}$, (49) becomes

$$\mathbb{E}\left[\|x^k - x^*\|^2 + \frac{1}{Lmn}\sum_{i=1}^{n}\|y_i^k + \nabla f_i(x^*)\|^2\right] \leq \left(1 - \frac{m}{L+mn}\right)^k R^0 \tag{53}$$

   where $R^0 = \|x^0 - x^*\|^2 + \frac{1}{Lmn}\sum_{i=1}^{n}\|y_i^0 + \nabla f_i(x^*)\|^2$. This is almost identical to Shalev-Shwartz (2015, Theorem 1). Statement 1 in Corollary 5 is slightly stronger since it only requires $\alpha \leq \frac{2}{L+2mn}$. Notice Shalev-Shwartz (2015, Theorem 1) requires $\alpha \leq \frac{1}{L+mn}$. Similarly, Statement 2 in Corollary 5 slightly improves Shalev-Shwartz (2015, Theorem 2) by allowing a slightly larger value of α.

4.5. Further Discussion on SAG

Finally, we explain why Theorem 1 fails in recovering the existing SAG rate bounds in Schmidt et al. (2013, Theorem 1), and briefly sketch how to extend our LMI-based analysis for SAG. The fundamental reason is that the proof of Schmidt et al. (2013, Theorem 1) requires $g \in \mathcal{F}(m, L)$, which is stronger than the condition $g \in \mathcal{S}(m, L)$. Notice in Theorem 1, we only incorporate one property of g, i.e.

$$\begin{bmatrix} v^k - x^* \\ \nabla g(v^k) \end{bmatrix}^T \begin{bmatrix} -2mLI_p & (L+m)I_p \\ (L+m)I_p & -2I_p \end{bmatrix} \begin{bmatrix} v^k - x^* \\ \nabla g(v^k) \end{bmatrix} \geq 0 \tag{54}$$

The above inequality couples v^k with x^*, and is satisfied for any $g \in \mathcal{S}(m, L)$. However, the proof for Schmidt et al. (2013, Theorem 1) actually relies on some advanced inequalities [2] coupling $f(v^{k+1})$ with $f(v^k)$. Such advanced inequalities typically require $g \in \mathcal{F}(m, L)$. In other words, the

2. See (11) in Schmidt et al. (2013) for such an inequality.

convexity of g is required in the convergence proof of SAG while our proofs for SAGA and Finito hold for some non-convex g.

Here is a similar example. The linear convergence of the full gradient descent method does not require convexity of the objective function, and can be proved using a basic quadratic inequality similar to (54). However, the linear convergence of Nesterov's accelerated method cannot be proved using this simple inequality and relies on some advanced inequalities coupling the current iterates with the past iterates. These advanced inequalities decode convexity much better than the simple inequality used in the proof of the full gradient descent method. One such advanced inequality is the so-called weighted off-by-one IQC (Lessard et al., 2016, Lemma 10). See Lessard et al. (2016, Section 4.5) for a detailed discussion on how to incorporate the weighted off-by-one IQC for analysis of Nesterov's accelerated method. The use of the weighted off-by-one IQC typically leads to larger LMIs which are difficult to solve analytically. Very recently, Hu and Lessard (2017) have proposed another inequality of similar nature to simplify the LMI-based analysis of Nesterov's accelerated method. The resultant LMI in Hu and Lessard (2017) is smaller and can be solved analytically to recover the standard rate of Nesterov's method. As a summary, more advanced quadratic inequalities which further exploit the property of convexity are required in the analysis of Nesterov's accelerated method, and this makes the analysis of Nesterov's accelerated method much more complicated than the analysis of the full gradient descent method.

Due to similar reasons, the analysis of SAG is more involved than other stochastic methods. Our quadratic constraint approach actually reveals the difficulties in analyzing different methods: SAGA, SDCA, and Finito only require simple constraints (19) (20) while SAG further requires more advanced quadratic constraints, e.g. weighted off-by-one IQC.

Now we briefly sketch two ways to address the analysis of SAG. First, one can combine our proposed jump system theory with the quadratic constraint derivation procedure in Hu and Lessard (2017). We can obtain a modified LMI condition which searches for a Lyapunov function in the form of $\left((\xi^k-\xi^*)^T P(\xi^k-\xi^*)+g(v^k)-g(x^*)\right)$ where P is some positive semidefinite matrix. We have some preliminary numerical rate results indicating that formulating such an LMI to search for Lyapunov functions in the more general form is sufficient to numerically analyze SAG. Actually, the original proof of Schmidt et al. (2013, Theorem 1) constructs such a Lyapunov function (Schmidt et al., 2013, Section B.2).

Another way to address the analysis of SAG is to incorporate the weighted off-by-one IQC (Lessard et al., 2016, Lemma 10) into our jump system framework. In this case, we can formulate an LMI condition to search for a quadratic function which is not a Lyapunov function in the technical sense but serves the purpose of linear convergence certifications. See Lessard et al. (2016, Remarks on Lyapunov Functions) for more explanations. We also have some preliminary numerical rate results suggesting that applying the weighted off-by-one IQC can recover the linear convergence rates in Schmidt et al. (2013, Theorem 1) and lead to new linear rate bounds under various assumptions on f_i.

Although there is no technical difficulty in incorporating these more advanced quadratic constraints into the LMI formulations for SAG, we have not been able to analytically solve these resultant LMIs. In addition, the use of such advanced quadratic constraints requires much heavier mathematical notation. For readability purposes, we do not include a detailed numerical rate analysis of SAG in this paper. See Lessard et al. (2016) and Hu and Lessard (2017) for detailed discussions on weighted off-by-one IQC and other more advanced quadratic constraints.

5. Main Technical Proofs

We present the proofs of Theorem 1, Corollary 3, Corollary 4, and Corollary 5 in this section. The proof of Theorem 2 is quite tedious, and hence left to Appendix C.

5.1. Proof of the Main LMI Condition (Theorem 1)

Based on the state space model in (6) and (18), we have

$$\begin{aligned} \xi^{k+1} - \xi^* &= A_{i_k}(\xi^k - \xi^*) + B_{i_k}(w^k - w^*) \\ v^k - v^* &= C(\xi^k - \xi^*) \end{aligned} \tag{55}$$

Denote $P = \tilde{P} \otimes I_p$, and define the Lyapunov function by $V(\xi^k) = (\xi^k - \xi^*)^T P(\xi^k - \xi^*)$. Based on (55), we have the following key relation:

$$\begin{aligned} &\mathbb{E}[V(\xi^{k+1}) \,|\, \mathcal{F}_{k-1}] \\ =&\mathbb{E}[(\xi^{k+1} - \xi^*)^T P(\xi^{k+1} - \xi^*) \,|\, \mathcal{F}_{k-1}] \\ =&\sum_{i=1}^n \mathbb{P}(i_k = i) \left[A_i(\xi^k - \xi^*) + B_i(w^k - w^*)\right]^T P \left[A_i(\xi^k - \xi^*) + B_i(w^k - w^*)\right] \\ =&\begin{bmatrix} \xi^k - \xi^* \\ w^k - w^* \end{bmatrix}^T \begin{bmatrix} \frac{1}{n}\sum_{i=1}^n A_i^T P A_i & \frac{1}{n}\sum_{i=1}^n A_i^T P B_i \\ \frac{1}{n}\sum_{i=1}^n B_i^T P A_i & \frac{1}{n}\sum_{i=1}^n B_i^T P B_i \end{bmatrix} \begin{bmatrix} \xi^k - \xi^* \\ w^k - w^* \end{bmatrix} \end{aligned} \tag{56}$$

Suppose $D_{\psi 1} = \tilde{D}_{\psi 1} \otimes I_p$ and $D_{\psi 2} = \tilde{D}_{\psi 2} \otimes I_p$. Notice we always have

$$\begin{bmatrix} 2L\nu I_p & (L-\nu)I_p \\ (L-\nu)I_p & -2I_p \end{bmatrix} = \begin{bmatrix} LI_p & -I_p \\ \nu I_p & I_p \end{bmatrix}^T \begin{bmatrix} 0_p & I_p \\ I_p & 0_p \end{bmatrix} \begin{bmatrix} LI_p & -I_p \\ \nu I_p & I_p \end{bmatrix} \tag{57}$$

Moreover, we have $C(\xi^k - \xi^*) = v^k - x^*$. Hence another key relation also holds as follows

$$\begin{aligned} &\begin{bmatrix} \xi^k - \xi^* \\ w^k - w^* \end{bmatrix}^T \begin{bmatrix} C^T D_{\psi 1}^T \\ D_{\psi 2}^T \end{bmatrix} \left(\begin{bmatrix} \lambda_1 & \tilde{0}^T \\ \tilde{0} & \frac{\lambda_2}{n} I_n \end{bmatrix} \otimes \begin{bmatrix} 0_p & I_p \\ I_p & 0_p \end{bmatrix} \right) \begin{bmatrix} D_{\psi 1} C & D_{\psi 2} \end{bmatrix} \begin{bmatrix} \xi^k - \xi^* \\ w^k - w^* \end{bmatrix} \\ =&\lambda_1 \begin{bmatrix} v^k - x^* \\ \frac{\sum_{i=1}^n (\nabla f_i(v^k) - \nabla f_i(x^*))}{n} \end{bmatrix}^T \begin{bmatrix} 2L\nu I_p & (L-\nu)I_p \\ (L-\nu)I_p & -2I_p \end{bmatrix} \begin{bmatrix} v^k - x^* \\ \frac{\sum_{i=1}^n (\nabla f_i(v^k) - \nabla f_i(x^*))}{n} \end{bmatrix} \\ +&\frac{\lambda_2}{n} \sum_{i=1}^n \begin{bmatrix} v^k - x^* \\ \nabla f_i(v^k) - \nabla f_i(x^*) \end{bmatrix}^T \begin{bmatrix} 2L\gamma I_p & (L-\gamma)I_p \\ (L-\gamma)I_p & -2I_p \end{bmatrix} \begin{bmatrix} v^k - x^* \\ \nabla f_i(v^k) - \nabla f_i(x^*) \end{bmatrix} \geq 0 \end{aligned} \tag{58}$$

The last step follows from (19) and (20), which are some simple quadratic inequalities capturing the properties of f_i. Now we can take the Kronecker product of the left side of (22) with I_p and immediately get

$$\begin{aligned} &\begin{bmatrix} \frac{1}{n}\sum_{i=1}^n A_i^T P A_i - \rho^2 P & \frac{1}{n}\sum_{i=1}^n A_i^T P B_i \\ \frac{1}{n}\sum_{i=1}^n B_i^T P A_i & \frac{1}{n}\sum_{i=1}^n B_i^T P B_i \end{bmatrix} + \\ &\qquad \begin{bmatrix} C^T D_{\psi 1}^T \\ D_{\psi 2}^T \end{bmatrix} \left(\begin{bmatrix} \lambda_1 & \tilde{0}^T \\ \tilde{0} & \frac{\lambda_2}{n} I_n \end{bmatrix} \otimes \begin{bmatrix} 0_p & I_p \\ I_p & 0_p \end{bmatrix} \right) \begin{bmatrix} D_{\psi 1} C & D_{\psi 2} \end{bmatrix} \leq 0 \end{aligned} \tag{59}$$

Therefore, left and right multiply the above inequality by $[(\xi^k - \xi^*)^T, (w^k - w^*)^T]$ and $[(\xi^k - \xi^*)^T, (w^k - w^*)^T]^T$ and apply (56), (58) to show that V satisfies:

$$\mathbb{E}[V(\xi^{k+1}) \,|\, \mathcal{F}_{k-1}] - \rho^2 V(\xi^k) \leq 0 \tag{60}$$

We can take full expectation to get $\mathbb{E}V(\xi^{k+1}) - \rho^2 \mathbb{E}V(\xi^k) \leq 0$. Consequently, we immediately have $\mathbb{E}V(\xi^k) \leq \rho^{2k} V(\xi^0)$ and $\mathbb{E}[\|\xi^k - \xi^*\|^2] \leq \rho^{2k} \left(\text{cond}(P)\|\xi^0 - \xi^*\|^2\right)$.

5.2. Analytical Proof for SAGA (Corollary 3)

To prove Statement 1, we set $\gamma = -m$ to reflect the assumption $f_i \in \mathcal{F}(m, L)$. Hence LMI (25) becomes

$$\begin{bmatrix} (1-\rho^2)p_2 - 2\lambda_1 mL - 2\lambda_2 mL & -\alpha p_2 + (L+m)(\lambda_1+\lambda_2) \\ -\alpha p_2 + (L+m)(\lambda_1+\lambda_2) & p_1 + \alpha^2 p_2 - 2(\lambda_1+\lambda_2) \end{bmatrix} \leq 0 \tag{61}$$

By Shur complements, LMIs (24) (25) are equivalent to

$$p_1 + \alpha^2 p_2 - 2\lambda_2 \leq 0 \tag{62}$$

$$\rho^2 \geq 1 - \frac{1}{n} - \left(\frac{\alpha^4 p_2^2}{p_1 + \alpha^2 p_2 - 2\lambda_2} - \alpha^2 p_2 \right) \frac{1}{np_1} \tag{63}$$

$$\rho^2 \geq 1 - 2(\lambda_1+\lambda_2) mL p_2^{-1} - \frac{(-\alpha p_2 + (L+m)(\lambda_1+\lambda_2))^2}{(p_1 + \alpha^2 p_2 - 2(\lambda_1+\lambda_2))p_2} \tag{64}$$

We can see that (63) describes how ρ^2 depends on n, while (64) describes how ρ^2 depends on m and L. We need the common feasible set for both (63) and (64).

More formally, given the testing rate $\rho^2 = 1 - \min\left\{\frac{2L\alpha - 1}{(L\alpha - 1)n}, 2m\alpha - \frac{\alpha m^2}{(1 - L\alpha)L}\right\}$, it is straightforward to verify $0 \leq \rho^2 \leq 1$ when $\alpha \leq \frac{1}{2L}$. For this particular rate, the condition (62) (63) (64) is feasible with $p_1 = \frac{1}{L}$, $p_2 = \frac{1}{\alpha}$, $\lambda_1 = 0$, and $\lambda_2 = \frac{1}{L}$. By Theorem 2, (39) holds as desired. Similarly, given the testing rate $\rho^2 = 1 - \min\left\{\frac{9L\alpha - 4}{(3L\alpha - 4)n}, 2m\alpha - \frac{3\alpha m^2}{(4 - 3L\alpha)L}\right\}$, we can choose $p_1 = \frac{2}{3L}$, $p_2 = \frac{1}{\alpha}$, $\lambda_1 = 0$, and $\lambda_2 = \frac{1}{L}$ to prove the bound (40). Therefore, Statement 1 is true.

To prove Statement 2, we set $\gamma = 0$ in (25) to reflect the assumption $f_i \in \mathcal{F}(0, L)$. Again, by Schur complements, LMIs (24) (25) are equivalent to (62) (63) and

$$\rho^2 \geq 1 - 2\lambda_1 mL p_2^{-1} - \frac{(-\alpha p_2 + (L+m)\lambda_1 + L\lambda_2)^2}{(p_1 + \alpha^2 p_2 - 2\lambda_1 - 2\lambda_2)p_2} \tag{65}$$

Given the testing rate $\rho^2 = 1 - \min\left\{\frac{2L\alpha - b}{(L\alpha - b)n}, 2(1-b)m\alpha - \frac{m^2(1-b)^2\alpha}{(2 - b - L\alpha)L}\right\}$, it is straightforward to verify $0 \leq \rho^2 \leq 1$ when $b \geq 2L\alpha$. For this particular rate, the condition (62) (63) (65) is feasible with $p_1 = \frac{b}{L} > 0$, $p_2 = \frac{1}{\alpha}$, $\lambda_1 = \frac{1-b}{L} \geq 0$, and $\lambda_2 = \frac{b}{L}$. By Theorem 2, (41) holds as desired. When $\alpha = \frac{1}{3L}$, we can choose any $b \in [\frac{2}{3}, 1]$ and (41) holds. Hence we can easily obtain (42) by choosing $b = \frac{5}{6}$ and applying the fact $\frac{m}{L} \leq 1$.

To prove Statement 3, we set $\gamma = L$ in (25) to reflect the assumption f_i being L-smooth. Again, by Schur complements, LMIs (24) (25) are equivalent to (62), (63) and

$$\rho^2 \geq 1 - 2\lambda_1 mL p_2^{-1} + 2\lambda_2 L^2 p_2^{-1} - \frac{(-\alpha p_2 + (L+m)\lambda_1)^2}{(p_1 + \alpha^2 p_2 - 2\lambda_1 - 2\lambda_2)p_2} \tag{66}$$

Given the testing rate $\rho^2 = 1 - \min\left\{\frac{b-2}{(b-1)n}, \frac{3m\alpha}{2} - 2bL^2\alpha^2\right\}$, it is straightforward to verify $0 \leq \rho^2 \leq 1$ when $2 \leq b \leq \frac{3m}{4\alpha L^2}$. For this particular rate, the condition (62) (63) (66) is feasible with $p_1 = b\alpha > 0$, $p_2 = \frac{1}{\alpha}$, $\lambda_1 = \frac{1}{L} \geq 0$, and $\lambda_2 = b\alpha$. Notice the facts $m \leq L$ and $b \geq 2$ are required when checking the feasibility of the LMI condition. By Theorem 2, (43) holds as desired. When $\alpha = \frac{m}{8L^2}$, we can choose any $b \in [2, 3]$ and (43) holds. Hence we can easily obtain (44) by choosing $b = 3$ and applying the fact $\frac{m}{L} \leq 1$. Similarly, when $\alpha = \frac{m^2}{4(m^2n+L^2)}$, we can choose any $2 \leq b \leq \frac{3(m^2n+L^2)}{L^2}$ and (43) holds. Hence we can also obtain (45) by choosing $b = \frac{2(m^2n+L^2)}{L^2}$ and apply the fact $\frac{2m^2}{2m^2n+L^2} \leq \frac{m^2}{8(m^2n+L^2)}$. This completes the proof.

Remark 6 *Based on the above proof, we can actually recover two other known results in Defazio et al. (2014a). First, it is known that SAGA achieves the rate $\rho^2 = 1 - \frac{m}{2(mn+L)}$ given the assumption $f_i \in \mathcal{F}(m, L)$ and the stepsize $\alpha = \frac{1}{2(mn+L)}$. To recover this result, we first consider the case where $L \geq 2m$. Clearly $\alpha L < \frac{1}{2}$. Then the formula (39) leads to a rate $\rho^2 = 1 - \frac{m}{mn+L} + \frac{m^2}{(L+2mn)L}$. If $L \geq 2m$, then the above rate bound is always better than $\rho^2 = 1 - \frac{m}{2(mn+L)}$. On the other hand, if $L \leq 2m$, we can use $p_2 = \frac{1}{\alpha}$, $\lambda_1 = 0$, $\lambda_2 = \frac{1}{L}$ and $p_1 = 0.75\lambda_2$ to prove the LMI condition is feasible with $\rho^2 = 1 - \min\left\{\frac{15mn-L}{n(15mn+9L)}, \frac{m}{mn+L} - \frac{2m^2}{(3L+5mn)L}\right\}$. Under the condition $L \leq 2m$, this rate bound is always lower than $1 - \frac{m}{2(mn+L)}$. Consequently, we successfully recover the existing rate bound $\rho^2 = 1 - \frac{m}{2(mn+L)}$ for $\alpha = \frac{1}{2(mn+L)}$. Second, when f_i is only assumed to convex and smooth, i.e. $f_i \in \mathcal{F}(0, L)$, we can also choose $\alpha = \frac{1}{3(mn+L)}$ in (41) and set $b = \frac{2}{3}$. This leads to*

$$\begin{aligned}
\mathbb{E}\left[\|x^k - x^*\|^2\right] &\leq \left(1 - \min\left\{\frac{2m}{L+2mn}, \frac{2m}{9(L+mn)} - \frac{m^2}{27L^2+36mnL}\right\}\right)^k R^0 \\
&= \left(1 - \frac{2m}{9(L+mn)} + \frac{m^2}{27L^2+36mnL}\right)^k R^0 \\
&\leq \left(1 - \frac{m}{6(mn+L)}\right)^k R^0
\end{aligned} \tag{67}$$

where $R^0 = \|x^0 - x^*\|^2 + \frac{2}{9(mn+L)L}\sum_{i=1}^n \|y_i^0 - \nabla f_i(x^*)\|^2$. The rate bound here is also consistent with the known result in Defazio et al. (2014a).

5.3. Analytical Proof for Finito (Corollary 4)

First, we need the following linear algebra result to relax the LMI conditions (28) (29) to some simpler testing conditions.

Lemma 7 *Suppose Y_{11}, Y_{12}, Y_{22}, α, and n are scalars. In addition, $Y_{11} \leq 0$, $Y_{22} \leq 0$, $\alpha > 0$ and $n > 0$. The following two statements are true.*

1. *If $Y_{12} \leq 0$, then* $\begin{bmatrix} Y_{11} + \alpha n Y_{12} & Y_{12} \\ Y_{12} & Y_{22} + \frac{Y_{12}}{\alpha n} \end{bmatrix} \leq 0.$
2. *If $Y_{12} \geq 0$, then* $\begin{bmatrix} Y_{11} - \alpha n Y_{12} & Y_{12} \\ Y_{12} & Y_{22} - \frac{Y_{12}}{\alpha n} \end{bmatrix} \leq 0.$

Proof Statement 1 can be proved using the fact $\begin{bmatrix} \alpha n & 1 \\ 1 & \frac{1}{\alpha n} \end{bmatrix} \geq 0$. Statement 2 can be proved using the fact $\begin{bmatrix} \alpha n & -1 \\ -1 & \frac{1}{\alpha n} \end{bmatrix} \geq 0$. ■

Next, we relax the LMIs (28) (29) to some simpler (but more conservative) testing conditions. The relaxed conditions are sufficiently useful for analysis of Finito under some big data condition.

Corollary 8 *Consider Finito (10) (11) with i_k sampled from a uniform distribution. Define $v^k = \frac{1}{n}\sum_{i=1}^n x_i^k - \alpha \sum_{i=1}^n y_i^k$. Suppose $g \in \mathcal{S}(m, L)$ with $m > 0$, and γ is defined by (4) based on assumptions on f_i. Given any testing rate $1 - \frac{1}{n} \leq \rho^2 \leq 1$, if there exist positive scalars p_1, p_4, and nonnegative scalars λ_1, λ_2 such that*

$$\alpha^2 - 2\lambda_2 + p_1 < 0 \tag{68}$$

$$n(1-\rho^2)p_1 - p_1 + 2\alpha^2 - \frac{2\alpha^4}{\alpha^2 - 2\lambda_2 + p_1} \leq 0 \tag{69}$$

$$n(1-\rho^2)p_4 - p_4 + \frac{2}{n^2} - \frac{2\alpha^2}{n^2(\alpha^2 - 2\lambda_2 + p_1)} \leq 0 \tag{70}$$

$$p_4 - \rho^2 + 2L\gamma\lambda_2 - 2Lm\lambda_1 + 1 - \frac{((L+m)\lambda_1 + (L-\gamma)\lambda_2 - \alpha)^2}{\alpha^2 - 2\lambda_1 - 2\lambda_2 + p_1} \leq 0 \tag{71}$$

then Finito (10) (11) with any initial condition $x_i^0 \in \mathbb{R}^p$ and $y_i^0 \in \mathbb{R}^p$ satisfies

$$\mathbb{E}\left[p_4 \sum_{i=1}^n \|x_i^k - x^*\|^2 + p_1 \sum_{i=1}^n \|y_i^k - \nabla f_i(x^*)\|^2 + \|v^k - x^*\|^2 \right] \leq \rho^{2k} R^0 \tag{72}$$

where $R^0 = p_4 \sum_{i=1}^n \|x_i^0 - x^*\|^2 + p_1 \sum_{i=1}^n \|y_i^0 - \nabla f_i(x^*)\|^2 + \|v^0 - x^*\|^2$.

Proof Consider $p_2 = \alpha^2$, $p_3 = -\frac{\alpha}{n}$, and $p_5 = \frac{1}{n^2}$. Clearly, we have

$$\begin{bmatrix} p_1 + np_2 & np_3 \\ np_3 & p_4 + np_5 \end{bmatrix} = \begin{bmatrix} p_1 & 0 \\ 0 & p_4 \end{bmatrix} + n \begin{bmatrix} -\alpha \\ \frac{1}{n} \end{bmatrix} \begin{bmatrix} -\alpha & \frac{1}{n} \end{bmatrix} > 0 \tag{73}$$

Applying Schur complement with respect to the $(3,3)$-entry of (28), we can immediately rewrite (28) as $p_1 + p_2 - 2\lambda_2 = \alpha^2 - 2\lambda_1 + p_1 \leq 0$ and $\begin{bmatrix} Y_{11} + \alpha n Y_{12} & Y_{12} \\ Y_{12} & Y_{22} + \frac{Y_{12}}{\alpha n} \end{bmatrix} \leq 0$, where Y_{11} is equal to the left side of (69), Y_{22} is equal to the left side of (70), and $Y_{12} = \frac{\alpha^3}{n(\alpha^2 - 2\lambda_2 + p_1)} - \frac{\alpha}{n}$. Similarly, we can apply Schur complement with respect to the $(3,3)$-entry of (29) and rewrite (29) as $p_1 + p_2 - 2\lambda_1 - 2\lambda_2 \leq 0$ and $\begin{bmatrix} Z_{11} - \alpha n Z_{12} & Z_{12} \\ Z_{12} & Z_{22} - \frac{Z_{12}}{\alpha n} \end{bmatrix} \leq 0$, where $Z_{11} = p_1(1 - \rho^2 - \frac{1}{n})$, $Z_{22} = p_4(1 - \rho^2 - \frac{1}{n})$, and Z_{12} is equal to the multiplication of α and the left side of (71). Based on the conditions in the corollary statement, we can directly apply Lemma 7 to show that (28) and (29) hold. Finally, notice

$$\begin{bmatrix} p_1 I_n + p_2 ee^T & p_3 ee^T \\ p_3 ee^T & p_4 I_n + p_5 ee^T \end{bmatrix} = \begin{bmatrix} p_1 & 0 \\ 0 & p_4 \end{bmatrix} \otimes I_n + \begin{bmatrix} -\alpha e \\ \frac{1}{n} e \end{bmatrix} \begin{bmatrix} -\alpha e^T & \frac{1}{n} e^T \end{bmatrix} \tag{74}$$

We can directly apply Statement 3 in Theorem 2 to complete the proof of this corollary. ■

Now we can choose p_1, p_4, λ_1 and λ_2 to prove Corollary 4. Notice (69), (70), and (71) are equivalent to

$$\rho^2 \geq 1 - \frac{1}{n} + \frac{2\alpha^2(p_1 - 2\lambda_2)}{np_1(\alpha^2 - 2\lambda_2 + p_1)} \tag{75}$$

$$\rho^2 \geq 1 - \frac{1}{n} + \frac{2(p_1 - 2\lambda_2)}{n^3p_4(\alpha^2 - 2\lambda_2 + p_1)} \tag{76}$$

$$\rho^2 \geq 1 - 2Lm\lambda_1 + 2L\gamma\lambda_2 + p_4 - \frac{((L+m)\lambda_1 + (L-\gamma)\lambda_2 - \alpha)^2}{\alpha^2 - 2\lambda_1 - 2\lambda_2 + p_1} \tag{77}$$

1. To prove Statement 1, we set $\gamma = -m$ to reflect the assumption $f_i \in \mathcal{F}(m, L)$. We choose $p_1 = \frac{\alpha}{L}$, $p_4 = 0.5m\alpha$, $\lambda_1 = 0$, and $\lambda_2 = \frac{\alpha}{L}$. Then (75), (76), and (77) become

$$\rho^2 \geq 1 - \frac{1}{n} + \frac{2\alpha L}{n(1 - \alpha L)} \tag{78}$$

$$\rho^2 \geq 1 - \frac{1}{n} + \frac{4}{n^3 m\alpha(1 - L\alpha)} \tag{79}$$

$$\rho^2 \geq 1 - 1.5m\alpha + \frac{m^2\alpha}{L(1 - L\alpha)} \tag{80}$$

   When $\alpha = \frac{1}{5L}$, the testing rate $\rho^2 = 1 - \min\left\{\frac{1}{2n}, \frac{m}{20L}\right\}$ satisfies (78) and (80). In addition, this testing rate also satisfies (79) under the further assumption $n \geq \sqrt{\frac{50L}{m}}$. Therefore, Statement 1 directly follows from Corollary 8.

2. To prove Statement 2, we set $\gamma = 0$ to reflect the assumption $f_i \in \mathcal{F}(0, L)$. We choose $p_1 = \frac{\alpha}{2L}$, $p_4 = 0.5m\alpha$, $\lambda_1 = \frac{\alpha}{2L}$, and $\lambda_2 = \frac{\alpha}{2L}$. Then (75), (76), and (77) become

$$\rho^2 \geq 1 - \frac{1}{n} + \frac{4\alpha L}{n(1 - 2\alpha L)} \tag{81}$$

$$\rho^2 \geq 1 - \frac{1}{n} + \frac{4}{n^3 m\alpha(1 - 2L\alpha)} \tag{82}$$

$$\rho^2 \geq 1 - 0.5m\alpha + \frac{m^2\alpha}{2L(3 - 2L\alpha)} \tag{83}$$

   When $\alpha = \frac{1}{8L}$, the testing rate $\rho^2 = 1 - \min\left\{\frac{1}{3n}, \frac{5m}{176L}\right\}$ satisfies (81) and (83). In addition, this testing rate also satisfies (82) under the further assumption $n \geq \sqrt{\frac{64L}{m}}$. Therefore, Statement 2 directly follows from Corollary 8.

3. To prove Statement 3, we set $\gamma = L$ to reflect the assumption f_i being L-smooth. We choose $p_1 = 4\alpha^2$, $p_4 = 0.75m\alpha$, $\lambda_1 = \frac{\alpha}{L}$, and $\lambda_2 = 4\alpha^2$. Then (75), (76), and (77) become

$$\rho^2 \geq 1 - \frac{1}{3n} \tag{84}$$

$$\rho^2 \geq 1 - \frac{1}{n} + \frac{32}{9n^3 m\alpha} \tag{85}$$

$$\rho^2 \geq 1 - 1.25m\alpha + 8L^2\alpha^2 + \frac{m^2\alpha}{L(2 + 3L\alpha)} \tag{86}$$

When $\alpha = \frac{1}{2nm}$, the testing rate $\rho^2 = 1 - \frac{1}{3n}$ satisfies (84). This testing rate also satisfies (85) if $n \geq 11$. Moreover, this testing rate also satisfies (86) under the further assumption $n \geq \frac{48L^2}{m^2}$. Due to the fact $L \geq m$, we always have $n \geq 11$ when $n \geq \frac{48L^2}{m^2}$. Therefore, Statement 3 directly follows from Corollary 8.

Now the proof is complete.

5.4. Analytical Proof for SDCA (Corollary 5)

To prove Statement 1 in Corollary 5, we set $\gamma = 0$ to reflect the assumption $f_i \in \mathcal{F}(0, L)$. When $\alpha \leq \frac{2}{L+2mn}$, we have $\tilde{\alpha} = \alpha mn \leq \frac{2mn}{L+2mn} < 1$. Given the testing rate $\rho^2 = 1 - m\alpha = 1 - \frac{\tilde{\alpha}}{n}$, it is straightforward to verify $0 \leq \rho^2 \leq 1$ when $\alpha \leq \frac{2}{L+2mn}$. For this particular rate, the coupled LMI conditions (34) and (35) in Statement 2 of Theorem 2 are feasible with $p_1 = \frac{1}{\tilde{\alpha}}$, $p_2 = \frac{1-\tilde{\alpha}}{\tilde{\alpha}^2}$, $\lambda_1 = 0$, and $\lambda_2 = \frac{(1-\tilde{\alpha})mn}{\tilde{\alpha}L}$. To see this, first notice $p_2 > 0$ and $0 < \lambda_2 \leq \frac{1}{2}$ given the fact $\tilde{\alpha} \leq \frac{2mn}{L+2mn} < 1$. With the given rate $\rho^2 = 1 - \frac{\tilde{\alpha}}{n}$ and the current choice of $(p_1, p_2, \lambda_1, \lambda_2)$, LMIs (34) and (35) become

$$\begin{bmatrix} \frac{n}{\tilde{\alpha}}(1-\rho^2) - 1 & 0 \\ 0 & 1 - 2\lambda_2 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 1 - 2\lambda_2 \end{bmatrix} \leq 0 \tag{87}$$

$$\begin{bmatrix} -1 - \frac{2n(1-\tilde{\alpha})}{\tilde{\alpha}} + (1-\rho^2)(\frac{n}{\tilde{\alpha}} + \frac{n^2(1-\tilde{\alpha})}{\tilde{\alpha}^2}) & 0 \\ 0 & 1 - 2\lambda_2 \end{bmatrix} = \begin{bmatrix} n(1 - \frac{1}{\tilde{\alpha}}) & 0 \\ 0 & 1 - 2\lambda_2 \end{bmatrix} \leq 0 \tag{88}$$

The above LMIs hold due to the fact $\lambda_2 \leq \frac{1}{2}$ and $\tilde{\alpha} < 1$. By Theorem 2, (49) holds.

To prove Statement 2 in Corollary 5, we set $\gamma = 0$ to reflect the assumption $f_i \in \mathcal{F}(0, L)$. When $\alpha \leq \frac{m}{L^2+m^2n}$, we have $\tilde{\alpha} = \alpha mn \leq \frac{m^2n}{L^2+m^2n} < 1$. Given the testing rate $\rho^2 = 1 - m\alpha = 1 - \frac{\tilde{\alpha}}{n}$, it is straightforward to verify $0 \leq \rho^2 \leq 1$ when $\alpha \leq \frac{m}{L^2+m^2n}$. For this particular rate, the coupled LMI conditions (34) and (35) in Statement 2 of Theorem 2 are feasible with $p_1 = \frac{1}{\tilde{\alpha}}$, $p_2 = \frac{1-\tilde{\alpha}}{\tilde{\alpha}^2}$, $\lambda_1 = \frac{(1-\tilde{\alpha})mn}{\tilde{\alpha}L}$, and $\lambda_2 = \frac{1}{2}$. With the given rate $\rho^2 = 1 - \frac{\tilde{\alpha}}{n}$ and the current choice of $(p_1, p_2, \lambda_1, \lambda_2)$, the left side of (34) becomes a zero matrix and clearly (34) holds. In addition, (35) becomes

$$\begin{bmatrix} n(1 - \frac{1}{\tilde{\alpha}}) + \frac{L^2}{m^2} & 0 \\ 0 & -2\lambda_1 \end{bmatrix} \leq 0 \tag{89}$$

The above inequality holds since we have $\tilde{\alpha} \leq \frac{m^2n}{L^2+m^2n}$. By Theorem 2, we can conclude that Statement 2 is true.

6. Conclusion and Future Work

In this paper, we developed a unified routine for analysis of stochastic optimization methods and demonstrate the utility of our proposed routine by analyzing SAGA, Finito, and SDCA under various conditions (with or without individual convexity, etc). Our routine includes five steps:

1. Choose proper (A_i, B_i, C) to rewrite the stochastic optimization method as a special case of our general jump system model (6).

2. Apply Theorem 1 to obtain an LMI testing condition for the linear convergence rate analysis.

3. Test LMI (22) numerically to narrow down Lyapunov function structures and useful function inequalities required by the further analysis.

4. Apply linear algebra tricks to convert LMI (22) into some equivalent small LMIs whose size do not depend on n.

5. Construct analytical proofs for linear convergence rate bounds using the resultant small LMIs.

The first step is case-dependent. However, this step is usually straightforward and technically not difficult. The second and third steps are completely automated and require no tricks at all. These two steps can even be done for non-uniform sampling strategy if we slightly modify the LMI condition in Theorem 1. In principle, one can implement (22) once, and just needs to update $(\tilde{A}_i, \tilde{B}_i, \tilde{C})$ matrices given any new method. The fourth step is case-dependent but only requires very basic linear algebra tricks. As long as the assumptions on f_i are the same for all i and a uniform sampling is used, one should be able to obtain such equivalent small LMIs. The fifth step is the most technical step. This step is case-dependent and can be non-trivial for some complicated algorithms, e.g. Finito. However, at least one can numerically solve the resultant small LMIs using semidefinite programming solvers and use the numerical results to guide the analytical proofs.

In the third step, one may realize that LMI (22) is not sufficient for analysis of certain methods, e.g. SAG. Then one needs to exploit more advanced function properties and incorporate more advanced quadratic constraints into the LMI formulations. See Lessard et al. (2016) and Hu and Lessard (2017) for detailed discussions on weighted off-by-one IQC and other advanced quadratic constraints. The applications of these advanced quadratic constraints require much heavier mathematical notation. A detailed analysis of more complicated stochastic methods using such advanced quadratic constraints is beyond the scope of this paper, and will be pursued in future research.

We believe our work is just a starting point for further studies of empirical risk minimization using tools from control theory. We briefly comment on several possible extensions of our proposed framework to conclude the paper.

Non-uniform sampling strategy: Theorem 1 can be easily modified to handle non-uniform sampling strategy. However, the LMI dimension reduction in this case is non-trivial since the solution for the resultant LMI cannot be easily parameterized using a few scalar decision variables. It requires more efforts to investigate how to reduce the dimension of the resultant LMI in this case. A possible solution may involve properly scaling Lyapunov functions with the sampling distribution.

Stochastic quadratic constraints and SVRG: SVRG (Johnson and Zhang, 2013) is an important method which cannot be represented by our jump system model (6). The main issue is that SVRG has a deterministic periodic component which cannot be captured by a jump system model. One needs to take the periodicity and the randomness into accounts simultaneously. It will be interesting to develop an LMI-based approach for automated analysis and design of SVRG and its non-convex variants (Allen-Zhu and Hazan, 2016). One possible idea is to absorb the randomness and the periodicity into an uncertainty block whose input/output behavior can be characterized by some stochastic quadratic constraints. Similar ideas have already been used to recover the standard convergence results for the SG method (Hu, 2016, Chapter 6).

Automated design procedure of stochastic optimization methods: One may apply our proposed LMIs to numerically design stochastic optimization methods for practical problems. A direct design approach relies on grid search and is similar to the design procedure in Lessard et al. (2016,

Section 6). A more general design approach may be developed using the following sparse optimization formulation. Based on our general model (6), a stochastic method is typically characterized by the matrices (A_i, B_i, C). Hence, the design of stochastic methods can be formulated as a sparse optimization problem where we need to select (A_i, C) and sparse B_i for $i = 1, \ldots, n$ to minimize the convergence rate ρ under the LMI constraint (22) and some other structure constraints. The sparsity of B_i is important since it ensures the per-iteration cost of the resultant method to be low.

Larger family of non-convex functions: Notice the main assumption in this paper is $g \in \mathcal{S}(m, L)$, and the convexity of g is not required. There exist convergence results for other families of non-convex functions, e.g. functions satisfying Polyak- Lojasiewicz (PL) inequality (Karimi et al., 2016; Reddi et al., 2016a,b). It is an important task to investigate how to extend our quadratic constraint approach for more general non-convex functions.

Accelerated methods: Various acceleration techniques (Nitanda, 2014; Lin et al., 2015; Shalev-Shwartz and Zhang, 2016; Defazio, 2016) have been proposed to improve the convergence guarantees of the stochastic optimization methods when the big data condition is not met. We will extend our LMI method to analyze stochastic accelerated methods (with or without individual convexity) in the future.

Randomly-Permuted ADMM with multiple blocks: The alternating direction method of multipliers (ADMM) (Boyd et al., 2011) is an important distributed optimization algorithm. There are some initial convergence results on ADMM with multiple blocks (Hong and Luo, 2012; Chen et al., 2016). The quantification of the mean-square convergence rates of the so-called randomly-permuted ADMM with multiple blocks (Sun et al., 2015) remains an open topic. IQCs have been successfully applied to analyze ADMM with two blocks (Nishihara et al., 2015). The extension of jump system theory for random-permuted ADMM with multiple blocks is an important future task.

Asynchronous settings: In parallel computing, the algorithm performance will typically be impacted by the communication delay and memory contention (Recht et al., 2011; Zhang and Kwok, 2014). In this case, it is necessary to assess the robustness of the optimization methods with respect to the delays in the gradient update. There exist many IQCs for time-varying delays in the controls literature (Kao, 2012; Kao and Lincoln, 2004; Kao and Rantzer, 2007; Pfifer and Seiler, 2015). One may apply a scaling trick to tailor these IQCs for convergence rate analysis (Hu and Seiler, 2016). Hence the IQC analysis may be extended to study the impacts of time delays on SAG, SAGA, Finito, SDCA and other related stochastic optimization methods.

Acknowledgments

The authors would like to thank the anonymous reviewers for their constructive comments. Bin Hu and Peter Seiler were supported by the National Science Foundation under Grant No. NSF-CMMI-1254129 entitled CAREER: Probabilistic Tools for High Reliability Monitoring and Control of Wind Farms. Bin Hu and Peter Seiler were also supported by the NASA Langley NRA Cooperative Agreement NNX12AM55A entitled Analytical Validation Tools for Safety Critical Systems Under Loss-ofControl Conditions, Dr. Christine Belcastro technical monitor. Anders Rantzer is a member of the LCCC Linnaeus Center and the ELLIIT Excellence Center at Lund University. His contribution was supported by the Swedish Research Council, grant 2016-04764, and the Institute for Mathematics and its Applications at University of Minnesota.

References

Z. Allen-Zhu and E. Hazan. Variance reduction for faster non-convex optimization. In *Advances in Neural Information Processing Systems*, 2016.

S. Bittanti and P. Colaneri. *Periodic systems: filtering and control*. Springer Science & Business Media, 2008.

L. Bottou and Y. LeCun. Large scale online learning. *Advances in neural information processing systems*, 16:217, 2004.

S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.

C. Chen, B. He, Y. Ye, and X. Yuan. The direct extension of ADMM for multi-block convex minimization problems is not necessarily convergent. *Mathematical Programming*, 155(1-2): 57–79, 2016.

O. Costa, M. Fragoso, and R. Marques. *Discrete-time Markov jump linear systems*. Springer Science & Business Media, 2006.

Inc. CVX Research. CVX: Matlab software for disciplined convex programming, version 2.0. http://cvxr.com/cvx, August 2012.

A. Defazio. A simple practical accelerated method for finite sums. In *Advances in Neural Information Processing Systems*, pages 676–684, 2016.

A. Defazio, F. Bach, and S. Lacoste-Julien. Saga: A fast incremental gradient method with support for non-strongly convex composite objectives. In *Advances in Neural Information Processing Systems*, 2014a.

A. Defazio, J. Domke, and T. Caetano. Finito: A faster, permutable incremental gradient method for big data problems. In *Proceedings of the 31st International Conference on Machine Learning*, pages 1125–1133, 2014b.

V. Dragan, T. Morozan, and A. Stoica. *Mathematical methods in robust control of discrete-time linear stochastic systems*. Springer, 2010.

Y. Drori and M. Teboulle. Performance of first-order methods for smooth convex minimization: a novel approach. *Mathematical Programming*, 145(1-2):451–482, 2014.

M. Grant and S. Boyd. Graph implementations for nonsmooth convex programs. In V. Blondel, S. Boyd, and H. Kimura, editors, *Recent Advances in Learning and Control*, Lecture Notes in Control and Information Sciences, pages 95–110. Springer-Verlag Limited, 2008.

J. Hespanha. *Linear systems theory*. Princeton university press, 2009.

M. Hong and Z. Luo. On the linear convergence of the alternating direction method of multipliers. *arXiv preprint arXiv:1208.3922*, 2012.

B. Hu. *A Robust Control Perspective on Optimization of Strongly-Convex Functions*. PhD thesis, University of Minnesota, 2016.

B. Hu and L. Lessard. Dissipativity theory for Nesterov's accelerated method. In *Proceedings of the 34th International Conference on Machine Learning*, 2017.

B. Hu and P. Seiler. Exponential decay rate conditions for uncertain linear systems using integral quadratic constraints. *IEEE Transactions on Automatic Control*, 61(11):3561–3567, 2016.

R. Johnson and T. Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in Neural Information Processing Systems*, pages 315–323, 2013.

C. Kao. On stability of discrete-time LTI systems with varying time delays. *IEEE Transactions on Automatic Control*, 57:1243–1248, 2012.

C. Kao and A. Rantzer. Stability analysis of systems with uncertain time-varying delays. *Automatica*, 43(6):959–970, 2007.

C.Y. Kao and B. Lincoln. Simple stability criteria for systems with time-varying delays. *Automatica*, 40:1429–1434, 2004.

H. Karimi, J. Nutini, and M. Schmidt. Linear convergence of gradient and proximal-gradient methods under the Polyak-Lojasiewicz condition. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 795–811, 2016.

D. Kim and J. Fessler. Optimized first-order methods for smooth convex minimization. *Mathematical programming*, 159(1-2):81–107, 2016.

L. Lessard, B. Recht, and A. Packard. Analysis and design of optimization algorithms via integral quadratic constraints. *SIAM Journal on Optimization*, 26(1):57–95, 2016.

H. Lin, J. Mairal, and Z. Harchaoui. A universal catalyst for first-order optimization. In *Advances in Neural Information Processing Systems*, pages 3384–3392, 2015.

A. Megretski and A. Rantzer. System analysis via integral quadratic constraints. *IEEE Transactions on Automatic Control*, 42:819–830, 1997.

R. Nishihara, L. Lessard, B. Recht, A. Packard, and M. Jordan. A general analysis of the convergence of ADMM. In *Proceedings of the 32nd International Conference on Machine Learning*, pages 343–352, 2015.

A. Nitanda. Stochastic proximal gradient descent with acceleration techniques. In *Advances in Neural Information Processing Systems*, pages 1574–1582, 2014.

H. Pfifer and P. Seiler. Integral quadratic constraints for delayed nonlinear and parameter-varying systems. *Automatica*, 56:36 – 43, 2015.

B. Recht, C. Re, S. Wright, and F. Niu. Hogwild: A lock-free approach to parallelizing stochastic gradient descent. In *Advances in Neural Information Processing Systems*, pages 693–701, 2011.

S. Reddi, A. Hefny, S. Sra, B. Póczós, and A. Smola. Stochastic variance reduction for nonconvex optimization. In *Proceedings of The 33rd International Conference on Machine Learning*, pages 314–323, 2016a.

S. Reddi, S. Sra, B. Póczós, and A. Smola. Fast incremental method for nonconvex optimization. In *IEEE Conf. on Decision and Control*, pages 1971–1977, 2016b.

H. Robbins and S. Monro. A stochastic approximation method. *The Annals of Mathematical Statistics*, 22(3):400–407, 1951.

N. Roux, M. Schmidt, and F. Bach. A stochastic gradient method with an exponential convergence rate for strongly-convex optimization with finite training sets. In *Advances in Neural Information Processing Systems*, 2012.

M. Schmidt, N. Roux, and F. Bach. Minimizing finite sums with the stochastic average gradient. *ArXiv preprint*, 2013.

S. Shalev-Shwartz and T. Zhang. Stochastic dual coordinate ascent methods for regularized loss. *The Journal of Machine Learning Research*, 14(1):567–599, 2013.

S. Shalev-Shwartz and T. Zhang. Accelerated proximal stochastic dual coordinate ascent for regularized loss minimization. *Mathematical Programming*, 155:105–145, 2016.

Shai Shalev-Shwartz. Sdca without duality. *arXiv preprint arXiv:1502.06177*, 2015.

Shai Shalev-Shwartz. Sdca without duality, regularization, and individual convexity. In *Proceedings of the 33rd International Conference on Machine Learning*, pages 747–754, 2016.

R. Sun, Z. Luo, and Y. Ye. On the expected convergence of randomly permuted ADMM. *arXiv preprint arXiv:1503.06387*, 2015.

A. Taylor, J. Hendrickx, and F. Glineur. Smooth strongly convex interpolation and exact worst-case performance of first-order methods. *Mathematical Programming*, 161(1-2):307–345, 2017.

C. Teo, A. Smola, S. Vishwanathan, and Q. Le. A scalable modular convex solver for regularized risk minimization. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 727–736, 2007.

K.C. Toh, M.J. Todd, and R.H. Tutuncu. SDPT3 - a matlab software package for semidefinite programming. *Optimization Methods and Software*, 11:545–581, 1999.

R.H Tutuncu, K.C. Toh, and M.J. Todd. Solving semidefinite-quadratic-linear programs using SDPT3. *Mathematical Programming Ser. B*, 95:189–217, 2003.

R. Zhang and J. Kwok. Asynchronous distributed ADMM for consensus optimization. In *Proceedings of the 31st International Conference on Machine Learning*, pages 1701–1709, 2014.

Appendix A. Jump System Formulations of SAG, Finito, and SDCA

1. (SAG): Define $w^k = \left[\nabla f_1(x^k)^T \cdots \nabla f_n(x^k)^T\right]^T$, and then the SAG gradient update rule (8) can still be rewritten as (15). Notice $\sum_{i=1}^n y_i^k = (e^T \otimes I_p) y^k$ and $\nabla f_{i_k}(x^k) - y_{i_k}^k = (e_{i_k}^T \otimes I_p)(w^k - y^k)$. Thus the iteration rule (9) can be rewritten as follows:

$$\begin{aligned} x^{k+1} &= x^k - \alpha \left(\frac{\nabla f_{i_k}(x^k) - y_{i_k}^k}{n} + \frac{1}{n} \sum_{i=1}^n y_i^k \right) \\ &= x^k - \frac{\alpha}{n} (e_{i_k}^T \otimes I_p)(w^k - y^k) - \frac{\alpha}{n} (e^T \otimes I_p) y^k \\ &= x^k - \frac{\alpha}{n} \left((e - e_{i_k})^T \otimes I_p \right) y^k - \frac{\alpha}{n} (e_{i_k}^T \otimes I_p) w^k \end{aligned} \tag{90}$$

   At this point, both the gradient update in (15) and the iteration update in (90) depend on $w^k = \left[\nabla f_1(x^k)^T \cdots \nabla f_n(x^k)^T\right]^T$. The key step in the modeling is to "separate out" this nonlinear term. Setting $v^k = x^k$ and then $w^k = \left[\nabla f_1(v^k)^T \cdots \nabla f_n(v^k)^T\right]^T$. Now the update rules in (15) and (90) can be expressed as:

$$\begin{aligned} \begin{bmatrix} y^{k+1} \\ x^{k+1} \end{bmatrix} &= \begin{bmatrix} (I_n - e_{i_k} e_{i_k}^T) \otimes I_p & \tilde{0} \otimes I_p \\ -\frac{\alpha}{n}(e - e_{i_k})^T \otimes I_p & I_p \end{bmatrix} \begin{bmatrix} y^k \\ x^k \end{bmatrix} + \begin{bmatrix} (e_{i_k} e_{i_k}^T) \otimes I_p \\ (-\frac{\alpha}{n} e_{i_k}^T) \otimes I_p \end{bmatrix} w^k \\ v^k &= \begin{bmatrix} \tilde{0}^T \otimes I_p & I_p \end{bmatrix} \begin{bmatrix} y^k \\ x^k \end{bmatrix} \\ w^k &= \begin{bmatrix} \nabla f_1(v^k) \\ \vdots \\ \nabla f_n(v^k) \end{bmatrix} \end{aligned} \tag{91}$$

   which is exactly in the form of the general jump system model (6) with $\xi^k = \begin{bmatrix} y^k \\ x^k \end{bmatrix}$. Recall that $w^* = \begin{bmatrix} \nabla f_1(x^*)^T & \dots & \nabla f_n(x^*)^T \end{bmatrix}^T$. It is trivial to set $\xi^* = \begin{bmatrix} (w^*)^T & (x^*)^T \end{bmatrix}^T$, and verify that (18) holds.

2. (Finito): Recall that we denote $y^k = \begin{bmatrix} (y_1^k)^T & \cdots & (y_n^k)^T \end{bmatrix}^T$ and $x^k = \begin{bmatrix} (x_1^k)^T & \cdots & (x_n^k)^T \end{bmatrix}^T$. We set v^k as

$$v^k = \frac{1}{n} \sum_{i=1}^n x_i^k - \alpha \sum_{i=1}^n y_i^k \tag{92}$$

   Again, we set $w^k = \left[\nabla f_1(v^k)^T \cdots \nabla f_n(v^k)^T\right]^T$. Then we can immediately rewrite (11) as

$$y^{k+1} = \left((I_n - e_{i_k} e_{i_k}^T) \otimes I_p \right) y^k + \left((e_{i_k} e_{i_k}^T) \otimes I_p \right) w^k \tag{93}$$

   It is also straightforward to rewrite (10) as

$$x^{k+1} = \left((I_n - e_{i_k} e_{i_k}^T + \frac{1}{n} (e_{i_k} e^T)) \otimes I_p \right) x^k - \alpha \left((e_{i_k} e^T) \otimes I_p \right) y^k \tag{94}$$

Therefore, we can combine (92), (93), and (94) to obtain

$$\begin{aligned}\begin{bmatrix} y^{k+1} \\ x^{k+1} \end{bmatrix} &= \begin{bmatrix} (I_n - e_{i_k} e_{i_k}^T) \otimes I_p & (\tilde{0}\tilde{0}^T) \otimes I_p \\ -\alpha (e_{i_k} e^T) \otimes I_p & (I_n - e_{i_k} e_{i_k}^T + \frac{1}{n}(e_{i_k} e^T)) \otimes I_p \end{bmatrix} \begin{bmatrix} y^k \\ x^k \end{bmatrix} + \begin{bmatrix} (e_{i_k} e_{i_k}^T) \otimes I_p \\ (\tilde{0}\tilde{0}^T) \otimes I_p \end{bmatrix} w^k \\ v^k &= \begin{bmatrix} -\alpha e^T \otimes I_p & \frac{1}{n} e^T \otimes I_p \end{bmatrix} \begin{bmatrix} y^k \\ x^k \end{bmatrix} \\ w^k &= \begin{bmatrix} \nabla f_1(v^k) \\ \vdots \\ \nabla f_n(v^k) \end{bmatrix} \end{aligned} \tag{95}$$

which is exactly in the form of the general jump system model (6) with $\xi^k = \begin{bmatrix} y^k \\ x^k \end{bmatrix}$.

Notice $\xi^k \in \mathbb{R}^{(n+1)p}$ for SAG and SAGA, but $\xi^k \in \mathbb{R}^{2np}$ for Finito. Hence in general, Finito requires more memory compared with SAG and SAGA. Based on the fact $\sum_{i=1}^n \nabla f_i(x^*) = 0$, we can set $\xi^* = \begin{bmatrix} w^* \\ e \otimes x^* \end{bmatrix}$, and verify that (18) holds. Therefore, if ξ^k converges to ξ^*, then y_i^k converges to $\nabla f_i(x^*)$ and x_i^k converges to x^*.

3. (SDCA): We still have $y^k = \begin{bmatrix} (y_1^k)^T & \cdots & (y_n^k)^T \end{bmatrix}^T$. The update rule (13) can be rewritten as

$$x^k = \frac{1}{mn}(e^T \otimes I_p) y^k \tag{96}$$

Again, $w^k = \begin{bmatrix} \nabla f_1(v^k)^T \cdots \nabla f_n(v^k)^T \end{bmatrix}^T$. Hence we can set $v^k = x^k$ and rewrite the update rule (14) as

$$y^{k+1} = \left((I_n - \alpha mn e_{i_k} e_{i_k}^T) \otimes I_p \right) y^k - \alpha mn \left((e_{i_k} e_{i_k}^T) \otimes I_p \right) w^k \tag{97}$$

We can augment (96) and (97) as

$$\begin{aligned} y^{k+1} &= \left((I_n - \alpha mn e_{i_k} e_{i_k}^T) \otimes I_p \right) y^k - \alpha mn \left((e_{i_k} e_{i_k}^T) \otimes I_p \right) w^k \\ v^k &= \left(\frac{1}{mn} e^T \otimes I_p \right) y^k \\ w^k &= \begin{bmatrix} \nabla f_1(v^k) \\ \vdots \\ \nabla f_n(v^k) \end{bmatrix} \end{aligned} \tag{98}$$

which is exactly in the form of the general jump system model (6) with $\xi^k = y^k$. Notice the state ξ^k is completely determined by y^k, and does not directly depend on x^k.

Appendix B. Numerical Tests Using the LMI Condition in Theorem 1

We can numerically solve the LMI (22) in Theorem 1 and get some rough ideas of the feasibility of the proposed LMI conditions.

First, we apply the proposed LMI condition to analyze the convergence rate of SAGA. The most relevant existing result for this case was presented in Defazio et al. (2014a, Section 2) and states the following fact. Under the assumption that $g \in \mathcal{F}(m, L)$ and $f_i \in \mathcal{F}(m, L)$, the SAGA iteration with the stepsize $\alpha = \frac{1}{3L}$ converges at a linear rate $\rho = \sqrt{1 - \min\{\frac{m}{3L}, \frac{1}{4n}\}}$ in the mean square sense. Therefore, for any m, L, and n, we can choose $\rho = \sqrt{1 - \min\{\frac{m}{3L}, \frac{1}{4n}\}}$ and numerically test the feasibility of the resultant LMI (22) using CVX (CVX Research, 2012; Grant and Boyd, 2008) with the solver SDPT3 (Tutuncu et al., 2003; Toh et al., 1999). As discussed before, we should set $\nu = \gamma = -m$ to reflect the assumptions $g \in \mathcal{F}(m, L)$ and $f_i \in \mathcal{F}(m, L)$. A practical issue is that the LMI is homogeneous, i.e. if $(\tilde{P}, \lambda_1, \lambda_2)$ is a feasible solution then $(c\tilde{P}, c\lambda_1, c\lambda_2)$ is also a feasible solution for any $c > 0$. This homogeneity can cause numerical issues. One method to break this homogeneity is to replace $\tilde{P} > 0$ with the condition $\tilde{P} \geq 10^{-2} I$. Based on some preliminary feasibility tests with relatively small n ($n < 100$), the proposed LMI remains feasible even if the following simple parameterization of $\tilde{P}$ is used

$$\tilde{P} = \begin{bmatrix} p_1 I_n & \tilde{0} \\ \tilde{0}^T & p_2 \end{bmatrix} \tag{99}$$

We notice that LMI (22) seems always feasible with the choice of $\rho = \sqrt{1 - \min\{\frac{m}{3L}, \frac{1}{4n}\}}$. This numerically confirms the existing rate result for n being up to several hundred. We further notice that the LMI can be feasible with ρ^2 smaller than $1 - \min\{\frac{m}{3L}, \frac{1}{4n}\}$. This indicates that one may get sharper rate bounds for SAGA using our proposed LMI. Finally, treating $\tilde{P}$ as an unknown matrix or parameterizing $\tilde{P}$ as (99) often does not change the feasibility of the resultant LMI. This implies that adopting the parameterization (99) does not introduce further conservatism into our analysis.

Similar testing can also be performed if f_i is only assumed to be L-smooth. We only need to modify the value of γ to be L. The numerical results suggest that using a simple parameterization (99) does not introduce further conservatism in this case. We can also perform such naive numerical analysis for SDCA, Finito and SAG for n being up to several hundred. The numerical results obtained by the proposed semidefinite programs actually inspire our analytical proofs for SAGA, SDCA, and Finito.

Appendix C. Proof of Theorem 2

The proof is based on the following key linear algebra result which can be used to transform certain high dimensional LMIs into two much smaller coupled LMIs.

Lemma 9 *The following statements are true:*

1. $\mu_1 I_n + q_1 e e^T > 0$ *if and only if* $\mu_1 > 0$ *and* $\mu_1 + n q_1 > 0$.

2.

$$\begin{bmatrix} \mu_1 I_n + q_1 e e^T & \mu_3 I_n + q_3 e e^T \\ \mu_3 I_n + q_3 e e^T & \mu_2 I_n + q_2 e e^T \end{bmatrix} \leq 0 \tag{100}$$

if and only if

$$\begin{bmatrix} \mu_1 & \mu_3 \\ \mu_3 & \mu_2 \end{bmatrix} \leq 0, \tag{101}$$

$$\begin{bmatrix} \mu_1 & \mu_3 \\ \mu_3 & \mu_2 \end{bmatrix} + n \begin{bmatrix} q_1 & q_3 \\ q_3 & q_2 \end{bmatrix} \leq 0 \tag{102}$$

3.

$$\begin{bmatrix} \mu_1 I_n + q_1 ee^T & q_4 e & \mu_6 I_n + q_6 ee^T \\ q_4 e^T & \mu_2 & q_5 e^T \\ \mu_6 I_n + q_6 ee^T & q_5 e & \mu_3 I_n + q_3 ee^T \end{bmatrix} \leq 0 \tag{103}$$

if and only if

$$\begin{bmatrix} \mu_1 & 0 & \mu_6 \\ 0 & \mu_2 & 0 \\ \mu_6 & 0 & \mu_3 \end{bmatrix} \leq 0, \tag{104}$$

$$\begin{bmatrix} \mu_1 + nq_1 & \sqrt{n} q_4 & \mu_6 + nq_6 \\ \sqrt{n}\mu_4 & \mu_2 & \sqrt{n} q_5 \\ \mu_6 + nq_6 & \sqrt{n} q_5 & \mu_3 + nq_3 \end{bmatrix} \leq 0 \tag{105}$$

4.

$$\begin{bmatrix} \mu_1 I_n + q_1 ee^T & \mu_4 I_n + q_4 ee^T & \mu_6 I_n + q_6 ee^T \\ \mu_4 I_n + q_4 ee^T & \mu_2 I_n + q_2 ee^T & \mu_5 I_n + q_5 ee^T \\ \mu_6 I_n + q_6 ee^T & \mu_5 I_n + q_5 ee^T & \mu_3 I_n + q_3 ee^T \end{bmatrix} \leq 0 \tag{106}$$

if and only if

$$\begin{bmatrix} \mu_1 & \mu_4 & \mu_6 \\ \mu_4 & \mu_2 & \mu_5 \\ \mu_6 & \mu_5 & \mu_3 \end{bmatrix} \leq 0, \tag{107}$$

$$\begin{bmatrix} \mu_1 & \mu_4 & \mu_6 \\ \mu_4 & \mu_2 & \mu_5 \\ \mu_6 & \mu_5 & \mu_3 \end{bmatrix} + n \begin{bmatrix} q_1 & q_4 & q_6 \\ q_4 & q_2 & q_5 \\ q_6 & q_5 & q_3 \end{bmatrix} \leq 0 \tag{108}$$

Proof Let $Q \in \mathbb{R}^{n \times (n-1)}$ be a matrix such that $\begin{bmatrix} \frac{e}{\sqrt{n}} & Q \end{bmatrix}$ is orthogonal. Then

$$\begin{bmatrix} \frac{e}{\sqrt{n}} & Q \end{bmatrix}^T (\mu_1 I_n + q_1 ee^T) \begin{bmatrix} \frac{e}{\sqrt{n}} & Q \end{bmatrix} = \text{diag}(\mu_1 + nq_1, \mu_1, \ldots, \mu_1) \tag{109}$$

Statement 1 directly follows since $\begin{bmatrix} \frac{e}{\sqrt{n}} & Q \end{bmatrix}$ is invertible. Similarly, Statement 2 can be immediately proved using the following fact:

$$\begin{bmatrix} \frac{e}{\sqrt{n}} & \tilde{0} & Q & 0Q \\ \tilde{0} & \frac{e}{\sqrt{n}} & 0Q & Q \end{bmatrix}^T \begin{bmatrix} \mu_1 I_n + q_1 ee^T & \mu_3 I_n + q_3 ee^T \\ \mu_3 I_n + q_3 ee^T & \mu_2 I_n + q_2 ee^T \end{bmatrix} \begin{bmatrix} \frac{e}{\sqrt{n}} & \tilde{0} & Q & 0Q \\ \tilde{0} & \frac{e}{\sqrt{n}} & 0Q & Q \end{bmatrix} \tag{110}$$

$$= \text{diag}\left(\begin{bmatrix} \mu_1 + nq_1 & \mu_3 + nq_3 \\ \mu_3 + nq_3 & \mu_2 + nq_2 \end{bmatrix}, \begin{bmatrix} \mu_1 & \mu_3 \\ \mu_3 & \mu_2 \end{bmatrix} \otimes I_{n-1} \right) \tag{111}$$

Statement 4 can be proved using a similar argument. Finally, Statement 3 can be proved using Statement 2 and a Schur complement argument. ■

When analyzing SDCA, we can apply Statement 2 of the above lemma to convert LMI (22) into two coupled 2×2 LMIs whose feasibility can be checked analytically. Similarly, Statement 3 of the above lemma is useful for the rate analysis of SAGA, and Statement 4 of the above lemma is useful for the rate analysis of Finito. Now we only need to substitute $(\tilde{A}_i, \tilde{B}_i, \tilde{C})$ and $\tilde{P}$ into the left side of (22), and then Theorem 2 directly follows from the above lemma.

1. To prove Statement 1 of Theorem 2, recall that we have $\tilde{P} = \begin{bmatrix} p_1 I_n & \tilde{0} \\ \tilde{0}^T & p_2 \end{bmatrix}$. For SAGA, it is straightforward to verify

$$\frac{1}{n}\sum_{i=1}^{n} \tilde{A}_i \tilde{P} \tilde{A}_i = \begin{bmatrix} (\frac{p_2\alpha^2}{n} + \frac{n-1}{n} p_1) I_n - \frac{\alpha^2 p_2}{n^2} ee^T & \tilde{0} \\ \tilde{0}^T & p_2 \end{bmatrix} \tag{112}$$

$$\frac{1}{n}\sum_{i=1}^{n} \tilde{B}_i \tilde{P} \tilde{A}_i = \begin{bmatrix} -\frac{\alpha^2 p_2}{n} I_n + \frac{\alpha^2 p_2}{n^2} ee^T \\ -\frac{\alpha p_2}{n} e^T \end{bmatrix} \tag{113}$$

$$\frac{1}{n}\sum_{i=1}^{n} \tilde{B}_i \tilde{P} \tilde{B}_i = \frac{p_1 + \alpha^2 p_2}{n} I_n \tag{114}$$

   In addition, we have

$$\begin{bmatrix} \tilde{C}^T \tilde{D}_{\psi 1}^T \\ \tilde{D}_{\psi 2}^T \end{bmatrix} \left(\begin{bmatrix} \lambda_1 & \tilde{0}^T \\ \tilde{0} & \frac{\lambda_2}{n} I_n \end{bmatrix} \otimes \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \right) \begin{bmatrix} \tilde{D}_{\psi 1} \tilde{C} & \tilde{D}_{\psi 2} \end{bmatrix} = \\ \lambda_1 \begin{bmatrix} 0_n & \tilde{0} & 0_n \\ \tilde{0}^T & -2mL & \frac{m+L}{n} e^T \\ 0_n & \frac{m+L}{n} e & -\frac{2}{n^2} ee^T \end{bmatrix} + \lambda_2 \begin{bmatrix} 0_n & \tilde{0} & 0_n \\ \tilde{0}^T & 2L\gamma & \frac{L-\gamma}{n} e^T \\ 0_n & \frac{L-\gamma}{n} e & -\frac{2}{n} I_n \end{bmatrix} \tag{115}$$

   Now we can directly prove Statement 1 of Theorem 2 by applying Statement 3 of Lemma 9 to convert (22) into small coupled LMIs.

2. To prove Statement 2 of Theorem 2, recall that we have

$$\tilde{P} = \begin{bmatrix} p_1 I_n + p_2 ee^T & p_3 ee^T \\ p_3 ee^T & p_4 I_n + p_5 ee^T \end{bmatrix} \tag{116}$$

   Hence it is straightforward to verify:

$$\frac{1}{n}\sum_{i=1}^{n} \tilde{A}_i \tilde{P} \tilde{A}_i = \begin{bmatrix} W_{11} & W_{12} \\ W_{12}^T & W_{22} \end{bmatrix} \tag{117}$$

$$\frac{1}{n}\sum_{i=1}^{n} \tilde{B}_i \tilde{P} \tilde{A}_i = \begin{bmatrix} -\frac{p_2}{n} I_n + \frac{1}{n}(p_2 - p_3\alpha) ee^T \\ -\frac{p_3}{n} I_n + \frac{(n+1)p_3}{n^2} ee^T \end{bmatrix} \tag{118}$$

$$\frac{1}{n}\sum_{i=1}^{n} \tilde{B}_i \tilde{P} \tilde{B}_i = \frac{p_1 + p_2}{n} I_n \tag{119}$$

where W_{11}, W_{12} and W_{22} are computed as

$$W_{11} = \left(\frac{p_2}{n} + \frac{n-1}{n}p_1\right) I_n + \left((1-\frac{2}{n})p_2 - 2(1-n^{-1})p_3\alpha + (p_4+p_5)\alpha^2\right) ee^T \quad (120)$$

$$W_{12} = \frac{p_3}{n} I_n + \frac{(n-1-n^{-1})p_3 - p_4\alpha - np_5\alpha}{n} ee^T \quad (121)$$

$$W_{22} = \left(\frac{p_5}{n} + (1-\frac{1}{n})p_4\right) I_n + \left(\frac{p_4}{n^2} + (1-\frac{1}{n^2})p_5\right) ee^T \quad (122)$$

Then we can combine Statement 4 of Lemma 9 with the following formula to prove Statement 2 of Theorem 2.

$$\begin{bmatrix} \tilde{C}^T\tilde{D}_{\psi 1}^T \\ \tilde{D}_{\psi 2}^T \end{bmatrix} \left(\begin{bmatrix} \lambda_1 & \tilde{0}^T \\ \tilde{0} & \frac{\lambda_2}{n} I_n \end{bmatrix} \otimes \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \right) \begin{bmatrix} \tilde{D}_{\psi 1}\tilde{C} & \tilde{D}_{\psi 2} \end{bmatrix} =$$
$$\lambda_1 \begin{bmatrix} -2Lm\alpha^2 ee^T & \frac{2Lm\alpha}{n} ee^T & -\frac{(m+L)\alpha}{n} ee^T \\ \frac{2Lm\alpha}{n} ee^T & -\frac{2mL}{n^2} ee^T & \frac{L+m}{n^2} ee^T \\ -\frac{(m+L)\alpha}{n} ee^T & \frac{L+m}{n^2} ee^T & -\frac{2}{n^2} ee^T \end{bmatrix} + \lambda_2 \begin{bmatrix} 2L\gamma\alpha^2 ee^T & -\frac{2L\gamma\alpha}{n} ee^T & -\frac{(L-\gamma)\alpha}{n} ee^T \\ -\frac{2L\gamma\alpha}{n} ee^T & \frac{2L\gamma}{n^2} ee^T & \frac{L-\gamma}{n^2} ee^T \\ -\frac{(L-\gamma)\alpha}{n} ee^T & \frac{L-\gamma}{n^2} ee^T & -\frac{2}{n} I_n \end{bmatrix} \quad (123)$$

3. To prove Statement 3 of Theorem 2, we have $\tilde{P} = p_1 I_n + p_2 ee^T$ and $\tilde{\alpha} = \alpha mn$. Hence it is straightforward to obtain the following formulas:

$$\frac{1}{n}\sum_{i=1}^n \tilde{A}_i\tilde{P}\tilde{A}_i = \left(\frac{p_1(\tilde{\alpha}^2 - 2\tilde{\alpha} + n)}{n} + \frac{p_2\tilde{\alpha}^2}{n}\right) I_n - \frac{p_2(2\tilde{\alpha} - n)}{n} ee^T \quad (124)$$

$$\frac{1}{n}\sum_{i=1}^n \tilde{B}_i\tilde{P}\tilde{A}_i = \left(\frac{p_1(\tilde{\alpha}^2 - \tilde{\alpha})}{n} + \frac{p_2\tilde{\alpha}^2}{n}\right) I_n - \frac{\tilde{\alpha}p_2}{n} ee^T \quad (125)$$

$$\frac{1}{n}\sum_{i=1}^n \tilde{B}_i\tilde{P}\tilde{B}_i = \frac{(p_1+p_2)\tilde{\alpha}^2}{n} I_n \quad (126)$$

In addition, we can directly obtain

$$\begin{bmatrix} \tilde{C}^T\tilde{D}_{\psi 1}^T \\ \tilde{D}_{\psi 2}^T \end{bmatrix} \left(\begin{bmatrix} \lambda_1 & \tilde{0}^T \\ \tilde{0} & \frac{\lambda_2}{n} I_n \end{bmatrix} \otimes \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \right) \begin{bmatrix} \tilde{D}_{\psi 1}\tilde{C} & \tilde{D}_{\psi 2} \end{bmatrix} =$$
$$\lambda_1 \begin{bmatrix} 0_n & \frac{L}{mn^2} ee^T \\ \frac{L}{mn^2} ee^T & -\frac{2}{n^2} ee^T \end{bmatrix} + \lambda_2 \begin{bmatrix} \frac{2L\gamma}{m^2n^2} ee^T & \frac{L-\gamma}{mn^2} ee^T \\ \frac{L-\gamma}{mn^2} ee^T & -\frac{2}{n} I_n \end{bmatrix} \quad (127)$$

Now Statement 3 of Theorem 2 directly follows from Statement 2 of Lemma 9.

Now the proof of Theorem 2 is complete.