A Service System with On-Demand Agents, Stochastic Gradient Algorithms and the SARAH Algorithm

by

Lam Minh Nguyen

Presented to the Graduate and Research Committee

of Lehigh University

in Candidacy for the Degree of

Doctor of Philosophy

in

Industrial and Systems Engineering

Lehigh University

August 2018

Approved and recommended for acceptance as a dissertation in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

_____
Date

_____
Dissertation Advisor

Committee Members:

_____
Katya Scheinberg, Committee Chair

_____
Martin Takáč

_____
Frank E. Curtis

_____
Alexander L. Stolyar

Acknowledgements

It would not have been possible to produce this dissertation without the help from the following people and it is a great pleasure to have the opportunity to express my gratitude.

First, I would like to express my sincerest gratitude to my PhD advisor – Katya Scheinberg for her invaluable guidance during my PhD life. Thank you for all opportunities you provided for me. My current and future success would definitely not have been possible without your constant help.

This dissertation would not have been completed without the support of my previous PhD advisor – Alexander Stolyar. Thank you for everything you did for me during the first few years of my PhD. I really achieve a lot of knowledge from you.

I would especially like to thank Martin Takáč for his guidance. Thank you for being such a great advisor, mentor and friend for my PhD life.

I wish to thank Frank E. Curtis, a member of my dissertation committee, for his consistent support and constructive criticism for my courses and my dissertation. I also would like to thank Tamás Terlaky for helping me during my initial study at Lehigh University and providing me useful advice. I am also grateful to Dzung Phan, Nam Nguyen, Jayant Kalagnanam, who served as mentors during my internship at IBM T. J. Watson Research Center.

In addition, I would like to thank my previous advisors Prasad Vemala, Kiran Desai, Susie Cox for helping me in doing research during my MBA and encouraging me to pursue my PhD degree, and my undergraduate advisors Vladimir I. Dmitriev and Alexander V. Il'in for their support at Lomonosov Moscow State University.

I also would like to thank all of my friends, colleagues, and professors for giving me

experience in my life.

Furthermore, I am extremely grateful to my parents, my sisters (Dasa and Masa), and my other relatives for their support.

Finally, my heartfelt gratitude goes out to my lovely wife Dung Nguyen and my beloved daughter Sarah Nguyen, who are always a motivation for me to step ahead and never give up. Before ending, I sincerely thank my wife for always supporting and being with me in my life. I would definitely not be able to get my today's achievements without your love, support, and motivation. I will always be grateful.

Lam M. Nguyen

August, 2018

Contents

List of Tables

List of Figures

Abstract

We consider a system, where a random flow of customers is served by agents invited on-demand. Each invited agent arrives into the system after a random time, and leaves it with some probability after each service completion. Customers and/or agents may be impatient. The objective is to design a real-time adaptive invitation scheme that minimizes customer and agent waiting times.

We study some aspects of the SGD method with a fixed, large learning rate and propose a novel assumption of the objective function, under which this method has improved convergence rates. We also propose a convergence analysis of SGD within a diminishing learning rate regime without bounded gradient assumption in the strongly convex case.

We propose the SARAH algorithm for solving finite-sum minimization problems in the strongly convex, convex, and nonconvex cases. We also consider a general stochastic optimization problem by using the SARAH algorithm with inexactness.

Introduction

This dissertation contains three parts: A Service System with On-Demand Agents (Part I: Chapter 1), Stochastic Gradient Algorithms (Part II: Chapters 2 and 3), and SARAH Algorithm (Part III: Chapters 4, 5 and 6). This work appears as [55, 56, 52, 53, 51, 54] .

In part I, we consider a system, where a random flow of customers is served by agents invited on-demand. Each invited agent arrives into the system after a random time, and leaves it with some probability after each service completion. Customers and/or agents may be impatient. The objective is to design a real-time adaptive invitation scheme that minimizes customer and agent waiting times. We consider a queue-length-based feedback scheme, study it in the asymptotic regime where the customer arrival rate goes to infinity; and derive a variety of sufficient conditions for the system local stability at the desired equilibrium point. Under these conditions, simulations show good overall performance of the scheme.

In part II, we study some aspects of the Stochastic Gradient Algorithm (or Stochastic Gradient Descent or SGD). In Chapter 2, we consider a standard stochastic gradient descent (SGD) method with a fixed, large step size and propose a novel assumption on the objective function, under which this method has improved convergence rates (to a neighborhood of the optimal solution set). We then empirically demonstrate that these assumptions hold for logistic regression and standard deep neural networks on classical data sets. Thus our analysis helps to explain when efficient behavior can be expected from the SGD method in training classification models and deep neural networks. In Chapter 3, we propose a convergence analysis of SGD within a diminishing learning rate regime without bounded gradient assumption in the strongly convex case, which results in more relaxed conditions

than those in [13]. We then move on the asynchronous parallel setting, and prove convergence of the Hogwild! algorithm in the same regime, obtaining the first convergence results for this method in the case of diminished learning rate.

In part III, we propose the SARAH algorithm for solving finite-sum minimization problems. We study SARAH as well as its practical variant SARAH+ for the convex case in Chapter 4. The linear convergence rate of SARAH is proven under a strong convexity assumption. We also prove a linear convergence rate (in the strongly convex case) for an inner loop of SARAH, a property that SVRG does not possess. In Chapter 5, we also consider a mini-batch version of SARAH for solving empirical loss minimization problems in the case of nonconvex losses. We provide a sublinear convergence rate (to stationary points) for general nonconvex functions and a linear convergence rate for gradient dominated functions, both of which have some advantages compared to other modern stochastic gradient algorithms for nonconvex losses. In Chapter 6, we consider the SARAH algorithm with inexactness. Instead of computing a full gradient at each outer iteration, we only compute a subset of samples. We also consider a general stochastic optimization problem.

Part I

A Service System With On-demand Agents

Chapter 1

A service system with on-demand agents

We study a system where a random flow of customers is served by servers (called agents) invited on-demand. Each invited agent arrives into the system after a random time; after each service completion, an agent returns to the system or leaves it with some fixed probabilities. Customers and/or agents may be impatient, that is, while waiting in queue, they leave the system at a certain rate (which may be zero). We consider the queue-length-based feedback scheme, which controls the number of pending agent invitations, depending on the customer and agent queue lengths and their changes. The basic objective is to minimize both customer and agent waiting times.

We establish the system process fluid limits in the asymptotic regime where the customer arrival rate goes to infinity. We use the machinery of switched linear systems and common quadratic Lyapunov functions to approach the stability of fluid limits at the desired equilibrium point, and derive a variety of sufficient local stability conditions. For our model, we conjecture that local stability is in fact sufficient for global stability of fluid limits; the validity of this conjecture is supported by numerical and simulation experiments. When local stability conditions do hold, simulations show good overall performance of the scheme.

1.1 Introduction

Consider a service system where a random flow of customers arrive exogenously. Servers, called *agents*, can be invited on-demand at any time. Invited agents arrive into the system not immediately, but after a random delay. When a customer is matched with an agent, a service occurs. After completing the service, the agent can either leave the system or return to serve more customers. Customers and/or agents may be impatient, that is, they abandon the system if their wait in queue exceeds some random *patience time*. The objective is to keep waiting times of both customers and agents small. Such system is schematically shown in Figure 1.1.

The model we consider is a generalized version of that in [55, 58]. In [58], there is no abandonment for both queues, and agents always leave the system after service completions. The model in [55] also has no abandonment, but, like in our model, an agent may return to the system after a service completion. Thus, our model is more realistic in many scenarios because customer abandonment is a key factor for call center operations (see e.g. [21, 80]).

More specifically, the model in this chapter is as follows. Customers arrive as a Poisson process and join a customer queue if no agent is available. Agents can be invited into the system exogenously, and join an agent queue after a random exponentially distributed time. There is an infinite pool of potential agents, which can be invited to serve customers. Customer service times are i.i.d. exponential. After the service completion, the customer leaves the system while the agent can return to the agent queue with some fixed probability. The matching of customers and agents is done in first-come-first-served (FCFS) order. The head-of-the-line customer and agent are matched immediately and together go to service, that is, there cannot be non-zero number of customers and agents simultaneously in the customer and agent queues. Customers and/or agents may be impatient and the patience times are independently exponentially distributed.

The model is primarily motivated by call/contact centers (see [75]), where agents that we consider are highly skilled. It is not reasonable to set a fixed working schedule for these agents since their time is very valuable. Instead, they are invited on-demand in real time. The purpose is to design a real-time adaptive agent invitation scheme that minimizes

customer and agent waiting times. However, designing an effective, simple and robust agent invitation strategy is non-trivial due to randomness in agent behavior.

We study a feedback-based adaptive scheme of [75, 58, 55], called *queue-length-based feedback scheme*, which controls the number of pending agent invitations, depending on the customer and/or agent queue lengths and their changes. The algorithm analysis in this chapter is substantially more challenging due to greater generality of our model. Just like in [55, 58], we consider a "stylized" version of the invitation scheme to make the analysis more tractable. Our simulation experiments in section 1.7.2 show that the behavior of the stylized scheme is very close to that of the more practical version of the queue-length-based feedback scheme.

We consider the system in the asymptotic regime where the customer arrival rate goes to infinity while the distributions of the agent response times, the service times and the patience times are fixed. We show convergence of the fluid-scaled process to the fluid limit (Theorem 1.4.1), which satisfies a system of differential equations. The key property of interest is the convergence of the fluid limit trajectories to the equilibrium point (at which the queues are zero). This property is referred to as *global stability* of the fluid limits. Establishing global stability appears to be very challenging, due to the fact that fluid limits have complicated behavior – there are two domains where they follow different ODEs, and a "reflecting" boundary. In this chapter, we focus on the *local stability* of fluid limits, defined as the stability of the dynamic system which describes fluid limit trajectories away from the boundary. The **main results** in this chapter (Theorem 1.4.2) give sufficient local stability conditions; the proof uses the machinery of switched linear systems and common quadratic Lyapunov functions [39, 74]. Theorem 1.4.2 implies many useful sufficient local stability conditions (Corollaries 1.4.1 - 1.4.12) for special cases, including those where customers never abandon or agents certainly leave the system after service completions. (Some of these corollaries – namely, Corollaries 1.4.9, 1.4.10 and 1.4.12 – strengthen the results in [55] for the non-abandonment system.) These sufficient local stability conditions are robust and easy to achieve in practice. Finally, we conjecture that, for our model, local stability is in fact sufficient for global stability, based on a large number of numerical and simulation experiments. Our simulation experiments also show good overall performance

of the feedback scheme when the local stability conditions do hold.

The model has many applications, or potential applications. For a general discussion of modern call/contact centers and their management, see, e.g. [2, 45]. Another example is telemedicine [7], where "agents" are doctors, invited on-demand to serve patients remotely. The model also arises in other applications, such as crowdsourcing-based customer service (see e.g. [20, 9]), taxi-service system, buyers and sellers in a trading market, and assembly systems. The model has relation to classical assemble-to-order models, where customers are orders and "invited agents" are products, which cannot be produced/assembled instantly. The model is also related to "double-ended queues" (see e.g. [29, 41]) and matching systems (see e.g. [24]); although in such models arrivals of all types into the system are typically exogenous, as opposed to being controlled.

Organization. The rest of the chapter is organized as follows. Some background facts on switched linear systems and common quadratic Lyapunov functions are given in section 1.2. In section 1.3, we describe the model and algorithm in detail. Section 1.4 states the main results of the chapter, which are proved in sections 1.5 and 1.6. Section 1.7 provides numerical and simulation experiments; it also contains our conjectures about global and local stability of fluid limits, supported by these experiments. A discussion of the results and future work is in section 1.8.

Basic notation: Symbols $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{R}$, $\mathbb{R}_+$ denote the sets of natural, integer, real, real non-negative numbers, respectively. $\mathbb{R}^d$ denotes the d-dimensional vector space. $\mathbb{R}^{d\times d}$ denotes the set of all $d\times d$ real matrices. The standard Euclidean norm of a vector $x\in\mathbb{R}^n$ is denoted $\|x\|$. For a vector a and matrix A, we write their transposes as a^T and A^T, respectively. For a matrix A, we write its inverse and determinant as A^{-1} and $\det(A)$, respectively. We write $x(\cdot)$ to mean the function (or random process) $(x(t), t\geq 0)$. For a real-valued function $x(\cdot):\mathbb{R}_+\to\mathbb{R}$, we use either $x'(t)$ or $(d/dt)x(t)$ to denote the derivative, and for $x(\cdot):\mathbb{R}_+\to\mathbb{R}^d$, $(d/dt)x(t) = (x_1'(t),\ldots,x_d'(t))$. For $x\in\mathbb{R}$, $x^+ = \max\{x,0\}$ and $x^- = -\min\{x,0\}$; and $\mathrm{sgn}(x) = 1$ if $x>0$, $\mathrm{sgn}(x)=0$ if $x=0$, and $\mathrm{sgn}(x)=-1$ if $x<0$. For $x,y\in\mathbb{R}$, we denote $x\wedge y = \min\{x,y\}$ and $x\vee y = \max\{x,y\}$. $a\Leftrightarrow b$ means "a is equivalent to b"; $a\Rightarrow b$ means "a implies b". We write $x^r\to x\in\mathbb{R}^n$ to denote ordinary convergence in $\mathbb{R}^n$. For a finite set of scalar functions $f_n(t)$, $t\geq 0$, $n\in\mathbb{N}$, a point t is called

regular if for any subset $\mathbb{N}_0 \subseteq \mathbb{N}$, the derivatives

$$\frac{d}{dt} \max_{n \in \mathbb{N}_0} f_n(t) \text{ and } \frac{d}{dt} \min_{n \in \mathbb{N}_0} f_n(t)$$

exist. (To be precise, we require that each derivative is proper: both left and right derivatives exist and are equal.)

Abbreviations: *u.o.c.* means *uniform on compact sets* convergence of functions, with the argument determined by the context (usually in $[0, \infty)$); *w.p.1* means *with probability 1*; *i.i.d.* means *independent identically distributed*; RHS means *right hand side*; FSLLN means *functional strong law of large numbers*; *CQLF* means *common quadratic Lyapunov function*; *LTI system* means *linear time-invariant system.*

1.2 Switched Linear Systems and CQLF

Common quadratic Lyapunov functions for switched linear systems play an important role in deriving our results. In this section, we provide some necessary background.

Consider a *switched linear system*

$$\Sigma_S : u'(t) = A(t)u(t) \ , \ A(t) \in \mathcal{A} = \{A_1, \ldots, A_m\} \tag{1.1}$$

where $\mathcal{A}$ is a set of matrices in $\mathbb{R}^{n \times n}$, and $t \to A(t)$ is a mapping from nonnegative real numbers into $\mathcal{A}$. (Usually, as in [74], this mapping is required to be piecewise constant with only finitely many discontinuities in any bounded time-interval. In our case this additional condition is not important, because our switched system will have a continuous derivative; see equation (1.8) below.) For $1 \leq i \leq m$, the i^{th} constituent system of the switched linear system (1.1) is the *linear time-invariant (LTI) system*

$$\Sigma_{A_i} : u'(t) = A_i u(t). \tag{1.2}$$

The origin is an *exponentially stable equilibrium* of the switched linear system Σ_s if there exist real constants $C > 0$, $a > 0$ such that $\|u(t)\| \leq Ce^{-at}\|u(0)\|$ for $t \geq 0$, for all solutions

$u(t)$ of the system (1.1) (see [27, 74]).

A symmetric square $n \times n$ matrix M with real coefficients is *positive definite* if $z^T M z > 0$ for every non-zero column vector $z \in \mathbb{R}^n$. A symmetric square $n \times n$ matrix M with real coefficients is *negative definite* if $z^T M z < 0$ for every non-zero column vector $z \in \mathbb{R}^n$. A square matrix A is called a *Hurwitz matrix* (or *stable matrix*) if every eigenvalue of A has strictly negative real part. The following fact is the Hurwitz criterion of matrices in $\mathbb{R}^{3\times 3}$ (see [62]).

Proposition 1.2.1 ([62]). *Let $L(\lambda) = \det(A - \lambda I) = 0$ be the characteristic equation of matrix A in $\mathbb{R}^{3\times 3}$:*

$$L(\lambda) = a_0\lambda^3 + a_1\lambda^2 + a_2\lambda + a_3 = 0 \ , \ a_0 > 0. \tag{1.3}$$

Matrix A is Hurwitz if and only if a_1, a_2, a_3 are positive and $a_1 a_2 > a_0 a_3$.

The function $V(u) = u^T P u$ is a *quadratic Lyapunov function* (QLF) for the system $\Sigma_A : u'(t) = Au(t)$ if (i) P is symmetric and positive definite, and (ii) $PA + A^T P$ is negative definite. Let $\{A_1, \ldots, A_m\}$ be a collection of $n \times n$ Hurwitz matrices, with associated stable LTI systems $\Sigma_{A_1}, \ldots, \Sigma_{A_m}$. Then the function $V(u) = u^T P u$ is a *common quadratic Lyapunov function* (CQLF) for these systems if V is a QLF for each individual system (see [39, 74]).

The following facts will be used in the proof of our main results (Theorem 1.4.2).

Proposition 1.2.2 ([39, 74]). *The existence of a CQLF for the LTI systems is sufficient for the exponential stability of the switched linear system.*

Proposition 1.2.3 ([39, 74]). *Let A_1 and A_2 be Hurwitz matrices in $\mathbb{R}^{n\times n}$, and the difference $A_1 - A_2$ has rank one. Then two systems $u'(t) = A_1 u(t)$ and $u'(t) = A_2 u(t)$ have a CQLF if and only if the matrix product $A_1 A_2$ has no negative real eigenvalues.*

Proposition 1.2.4 ([73]). *If A_1^{-1} is non-singular, the product $A_1 A_2$ has no negative eigenvalues if and only if $A_1^{-1} + \tau A_2$ is non-singular for all $\tau \geq 0$.*

1.3 Model and Algorithm

1.3.1 Model

Our model is a generalization of that considered in [55, 58]. Customers arrive according to a Poisson process of rate $\Lambda > 0$, and join a customer queue waiting for an available agent and are served in the order of their arrival. There is an infinite pool of 'potential' agents, which can be invited to serve customers. After a potential agent is invited, it becomes a 'pending' agent; we refer to such an event as an *invitation*. A pending agent 'accepts' its invitation and becomes 'active' agent after a random, exponentially distributed, time with mean $1/\beta$; we refer to such an event as an *acceptance*. Upon acceptance events, the new active agents join the (active) agent queue. The customer and agent queues cannot be positive simultaneously: the head-of-the-line customer and agent are immediately matched, leave their queues, and together go to service. Each service time is an exponentially distributed random variable with mean $1/\mu$; after the service completion, the customer leaves the system, while the corresponding agent either remains active and rejoins the agent queue – this occurs with probability $\alpha \in [0, 1)$ – or leaves the system with probability $1-\alpha$. Thus, there are two ways in which agents join the queue – when an agent becomes active (upon acceptance event) and already active agents rejoining the queue after service completions. The patience times of customers and agents are independent sequences of i.i.d. exponential random variables with rate $\delta \geq 0$ and $\theta \geq 0$, respectively. When its patience time expires while a customer or server wait in queue, they leave the system. (The model in [55] is a special case of ours, with $\delta = 0$ and $\theta = 0$; in other words, customers and agents certainly wait in their queues until they are matched. The model in [58] is a special case of ours, with $\delta = 0$, $\theta = 0$ and $\alpha = 0$.) Figure 1.1 depicts such a system.

Let $X(t)$ be the number of pending agents at time t. Let $Y(t) = Q_a(t) - Q_c(t)$ be the difference between the agent and customer queue lengths at time t. (Note that $Q_a(t) = Y^+(t)$ and $Q_c(t) = Y^-(t)$.) Let $Z(t)$ be the number of customers (or agents) in service at time t. The system state at time t is $(X(t), Y(t), Z(t))$.

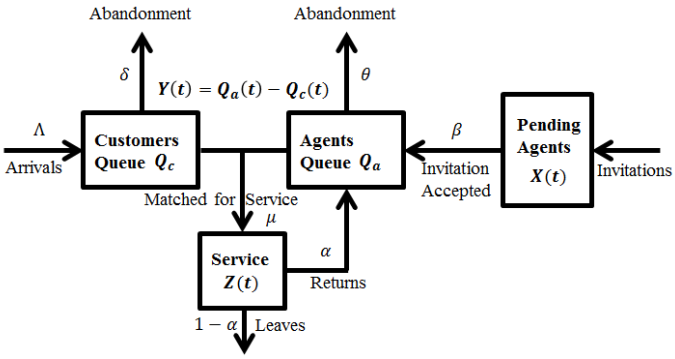


Figure 1.1: An agent invitation system

1.3.2 Algorithm

The queue-length-based feedback scheme in [55, 58, 75], referred to as the *actual scheme*, maintains a "target" $X_{target}(t)$ for the number of pending agents $X(t)$. $X_{target}(t)$ is changed by $\Delta X_{target}(t) = [-\gamma \Delta Y(t) - \epsilon Y(t)\Delta t]$ at each time t when $Y(t)$ changes by $\Delta Y(t)$ ($+1$ or -1), where $\gamma > 0$ and $\epsilon > 0$ are the algorithm parameters and Δt is the time duration from the previous change of Y. New agent invitations occur (i.e., the number of pending agents increases) if and only if $X(t) < X_{target}(t)$, where $X(t)$ is the actual number of pending agents; therefore, $X(t) \geq X_{target}(t)$ holds at all times. In addition, $X_{target}(t) \geq 0$; i.e. if an update of $X_{target}(t)$ makes it negative, its value is immediately reset to zero. Note that $X_{target}(t)$ is not necessarily an integer.

Just like in [55, 58], to simplify our theoretical analysis, we consider a "stylized" version of the actual scheme, referred to as the *stylized scheme*, which has the same basic dynamics, but keeps $X_{target}(t)$ integer and assumes that $X(t) = X_{target}(t)$ at all times; the latter is equivalent to assuming that not only agents can be invited instantly, but pending agents can be removed from the system at any time. Formally, the stylized scheme is defined as follows. There are six types of mutually independent, and independent of the past, events that affect the dynamics of $X(t)$, $Y(t)$ and $Z(t)$ in a small time interval $[t, t + dt]$:

- a customer arrival with probability $\Lambda dt + o(dt)$,
- an acceptance with probability $\beta X(t)dt + o(dt)$,
- an additional event (we will call it a type-3 event) with probability $\epsilon|Y(t)|dt + o(dt)$;

unlike other events, it is triggered by the algorithm itself, as opposed to other events triggered by customers' and/or agents' "movement" in the system,

- a service completion with probability $\mu Z(t)dt + o(dt)$,
- an abandonment in the customer queue with probability $\delta Y^-(t)dt + o(dt)$,
- an abandonment in the agent queue with probability $\theta Y^+(t)dt + o(dt)$.

The changes at these event times are described as follows:

- Upon a customer arrival, if $Y(t) > 0$, $Z(t)$ changes by $\Delta Z(t) = 1$; and if $Y(t) \leq 0$, $Z(t)$ changes by $\Delta Z(t) = 0$. $Y(t)$ changes by $\Delta Y(t) = -1$, and $X(t)$ changes by a random quantity with average $\gamma > 0$. For example, if $\gamma = 1.7$ and $\Delta Y(t) = -1$, then $\Delta X(t) = 2$ with probability 0.7 and $\Delta X(t) = 1$ with probability 0.3. Note that if γ is integer, $\Delta X(t) = \gamma$ w.p.1. To simplify the exposition, we assume that $\gamma > 0$ is an integer.
- Upon an acceptance event, if $Y(t) < 0$, $Z(t)$ changes by $\Delta Z(t) = 1$; and if $Y(t) \geq 0$, $Z(t)$ changes by $\Delta Z(t) = 0$. $Y(t)$ changes by $\Delta Y(t) = 1$, and $X(t)$ changes by $\Delta X(t) = -(\gamma \wedge X(t))$, that is, the change is by $-\gamma$ but $X(t)$ is kept to be nonnegative.
- Upon a type-3 event, if $X(t) \geq 1$, the change $\Delta X(t) = -\text{sgn}(Y(t))$ occurs; and if $X(t) = 0$, the change $\Delta X(t) = 1$ occurs if $Y(t) < 0$ and $\Delta X(t) = 0$ if $Y(t) \geq 0$.
- Upon a service completion, (a) if the agent returns to the agent queue (with probability α), then if $Y(t) < 0$, the change $\Delta Z(t) = 0$ occurs; and if $Y(t) \geq 0$, the change $\Delta Z(t) = -1$ occurs; $Y(t)$ changes by $\Delta Y(t) = 1$, and $\Delta X(t) = -(\gamma \wedge X(t))$. (b) If the agent leaves the system (with probability $1-\alpha$), then $Z(t)$ changes by $\Delta Z(t) = -1$.
- Upon a customer abandonment, $Y(t)$ changes by $\Delta Y(t) = 1$, and $X(t)$ changes by $\Delta X(t) = -(\gamma \wedge X(t))$.
- Upon an agent abandonment, $Y(t)$ changes by $\Delta Y(t) = -1$, and $X(t)$ changes by $\Delta X(t) = \gamma$.

Let $V(t) = Y^+(t) + Z(t)$ be the total number of agents in the system at time t. Obviously, $(X(t), Y(t), V(t))$ is a random process with states being 3-dimensional integer vectors. However, *very informally*, the basic dynamics of $(X(t), Y(t), V(t))$ under the stylized scheme can be thought of as described by the following ODE

$$\begin{cases} (d/dt)X = -\gamma(d/dt)Y - \epsilon Y \\ (d/dt)Y = \beta X - \Lambda + \alpha\mu Z + \delta Y^- - \theta Y^+ \\ (d/dt)V = \beta X - (1-\alpha)\mu Z - \theta Y^+. \end{cases} \tag{1.4}$$

ODE (1.4) is only to provide the basic intuition for the system dynamics – it is not used in the analysis.

1.4 Main Results

We consider a sequence of systems, indexed by a scaling parameter $r \to \infty$. In the system with index r, the arrival rate is $\Lambda = \lambda r$, while the parameters α, β, μ, δ, θ, ϵ, γ do not depend on r. The corresponding process is $(X^r(t), Y^r(t), Z^r(t)), t \geq 0$. The desired system operating point, at which $(X^r(t), Y^r(t), Z^r(t))$ should be centered is given by $(\lambda r(1-\alpha)/\beta, 0, \lambda r/\mu)$. The explanation of this choice is as follows. If an invitation scheme works as desired, $Y^r(t)$ should be close to 0; the number of customer-agent pairs $Z^r(t)$ should be close to its average value, which is $\lambda r/\mu$, so that the customers leave the system at rate λr; finally, $X^r(t)$ should be close to the value χ, such that the total average rate at which agents join the agent queue, which is $\chi\beta + [(\lambda r)/\mu]\mu\alpha$, is equal to the customer arrival rate λr – this gives $\chi = \lambda r(1-\alpha)/\beta$.

However, instead of considering process $(X^r(t), Y^r(t), Z^r(t))$, we will consider process $(X^r(t),$
$Y^r(t), V^r(t))$, which is more convenient for the analysis. (Recall that $Z^r(t) = V^r(t) - (Y^r(t))^+$.) Then the natural centering value for $V^r(t)$ is same as for $Z^r(t)$, namely $\lambda r/\mu$.

We define fluid-scaled process with centering as

$$(\bar{X}^r(t), \bar{Y}^r(t), \bar{V}^r(t)) = r^{-1}\left(X^r(t) - \frac{\lambda r(1-\alpha)}{\beta}, Y^r(t), V^r(t) - \frac{\lambda r}{\mu}\right), \quad t \geq 0. \tag{1.5}$$

Theorem 1.4.1. *Consider a sequence of processes $(\bar{X}^r(\cdot), \bar{Y}^r(\cdot), \bar{V}^r(\cdot))$, $r \to \infty$, with deterministic initial states such that $(\bar{X}^r(0), \bar{Y}^r(0), \bar{V}^r(0)) \to (x(0), y(0), v(0))$ for some fixed $(x(0), y(0), v(0)) \in \mathbb{R}^3$, $x(0) \geq -\frac{\lambda(1-\alpha)}{\beta}$. Then, these processes can be constructed on a common probability space, so that the following holds. W.p.1, from any subsequence of r, there exists a further subsequence such that*

$$(\bar{X}^r(\cdot), \bar{Y}^r(\cdot), \bar{V}^r(\cdot)) \to (x(\cdot), y(\cdot), v(\cdot)) \quad \textit{u.o.c. as} \quad r \to \infty \tag{1.6}$$

where $(x(\cdot), y(\cdot), v(\cdot))$ is a locally Lipschitz trajectory such that at any regular point $t \geq 0$

$$\begin{cases} x'(t) = \begin{cases} -\gamma y'(t) - \epsilon y(t), & \text{if } x(t) > -\frac{\lambda(1-\alpha)}{\beta} \\ [-\gamma y'(t) - \epsilon y(t)] \vee 0, & \text{if } x(t) = -\frac{\lambda(1-\alpha)}{\beta} \end{cases} \\ y'(t) = \beta x(t) + \alpha\mu(v(t) - y^+(t)) + \delta y^-(t) - \theta y^+(t) \\ v'(t) = \beta x(t) - (1-\alpha)\mu(v(t) - y^+(t)) - \theta y^+(t). \end{cases} \tag{1.7}$$

A limit trajectory $(x(\cdot), y(\cdot), v(\cdot))$ specified in Theorem 1.4.1 will be called a *fluid limit* starting from $(x(0), y(0), v(0))$.

Remark 1.4.1. *Equations (1.7), which a fluid limit must satisfy, are very natural. They can be thought of as rescaled centered versions of the (informal) equations (1.4). In addition, (1.7) includes a "reflection" (or, "regulation") at the boundary $x = -\frac{\lambda(1-\alpha)}{\beta}$, i.e. condition $x(t) \geq -\frac{\lambda(1-\alpha)}{\beta}$ is "enforced" as all times. This additional condition is the centered rescaled version of the condition $X^r(t) \geq 0$, which obviously must hold at all times.*

Consider a dynamic system $(x(t), y(t), v(t)) \in \mathbb{R}^3$:

$$\begin{cases} x'(t) = -\gamma y'(t) - \epsilon y(t) \\ y'(t) = \beta x(t) + \alpha\mu(v(t) - y^+(t)) + \delta y^-(t) - \theta y^+(t) \\ v'(t) = \beta x(t) - (1-\alpha)\mu(v(t) - y^+(t)) - \theta y^+(t). \end{cases} \tag{1.8}$$

Note that the RHS of (1.8) is continuous. This dynamic system describes the dynamics of fluid limit trajectories when the state is away from the boundary $x = -\frac{\lambda(1-\alpha)}{\beta}$. System (1.8) is a generalization of the system considered in [55], referred to as a *non-abandonment system*, which is a special case of ours with $\delta = 0$ and $\theta = 0$.

We say that the fluid limit is *globally stable* if every fluid limit trajectory converges to the equilibrium point $(0, 0, 0)$; and it is *locally stable* if every trajectory of the dynamic system (1.8) converges to the equilibrium point $(0, 0, 0)$. Note that exponential stability of the system (1.8) implies local stability.

The following theorem is the main result of this chapter. It provides sufficient exponential stability conditions for the system (1.8).

Theorem 1.4.2 (Sufficient exponential stability conditions)**.** *For any set of positive β, μ, ϵ, γ, non-negative δ and θ, and $\alpha \in [0, 1)$, such that either* (i)

$$\gamma > \max\left\{\frac{\alpha\mu - \delta}{\beta}, \sqrt{\frac{(2-\alpha)\epsilon\mu + \alpha\epsilon\delta}{\beta\mu}}\right\}, \tag{1.9}$$

or (ii)

$$\gamma > \max\left\{\frac{\alpha\mu - \delta + \sqrt{(\alpha\mu - \delta)^2 + 4\alpha\mu^2}}{2\beta}, \sqrt{\max\left\{\frac{\alpha\epsilon(\delta - \mu)}{\beta\mu}, 0\right\}}\right\} \tag{1.10}$$

holds, a common quadratic Lyapunov function (CQLF) of the system (1.8) *exists, and the system* (1.8) *is exponentially stable.*

In other words, conditions (1.9) and (1.10) are sufficient for local stability of our system. Theorem 1.4.2 implies the following useful sufficient local stability conditions (Corollaries 1.4.1 - 1.4.12) for special cases. Figure 1.2 depicts the connection between these results.

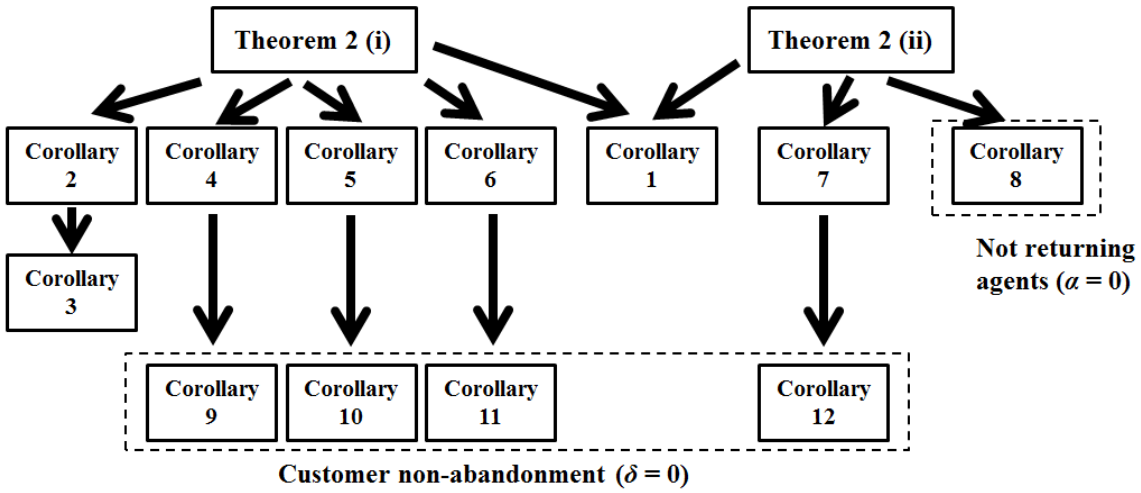


Figure 1.2: Result's diagram

Corollary 1.4.1. *Given all other parameters are fixed, the system* (1.8) *is exponentially stable for all sufficiently large* γ.

Corollary 1.4.2. *If* $\alpha\mu \leq \delta$, *then the system* (1.8) *is exponentially stable under condition*

$$\gamma > \sqrt{\frac{(2-\alpha)\epsilon\mu + \alpha\epsilon\delta}{\beta\mu}}. \tag{1.11}$$

Corollary 1.4.3. *If* $\alpha\mu \leq \delta$, *then the system* (1.8) *is exponentially stable for all sufficiently small* ϵ.

Corollary 1.4.4. *If* $\alpha\mu > \delta$ *and* $\epsilon \leq \frac{(\alpha\mu-\delta)^2\mu}{(2-\alpha)\mu\beta+\alpha\delta\beta}$, *then the system* (1.8) *is exponentially stable under condition*

$$\gamma > \frac{\alpha\mu - \delta}{\beta}. \tag{1.12}$$

Corollary 1.4.5. *If* $\alpha\mu > \delta$ *and* $\epsilon > \frac{(\alpha\mu-\delta)^2\mu}{(2-\alpha)\mu\beta+\alpha\delta\beta}$, *then the system* (1.8) *is exponentially stable under condition*

$$\gamma > \sqrt{\frac{(2-\alpha)\epsilon\mu + \alpha\epsilon\delta}{\beta\mu}}. \tag{1.13}$$

Corollary 1.4.6. *If $\alpha\mu \geq \delta$, then the system* (1.8) *is exponentially stable under condition*

$$\gamma > \frac{\alpha\mu - \delta + \sqrt{(\alpha\mu - \delta)^2 + 8\beta\epsilon}}{2\beta}. \tag{1.14}$$

Corollary 1.4.7. *If $\mu > \delta$, then the system* (1.8) *is exponentially stable under condition*

$$\gamma > \frac{\alpha\mu - \delta + \sqrt{(\alpha\mu - \delta)^2 + 4\alpha\mu^2}}{2\beta}. \tag{1.15}$$

(Note that this condition does not depend on ϵ.)

We also have the following result for the system where agents do not return to the agent queue after service completions.

Corollary 1.4.8. *If $\alpha = 0$, then the system* (1.8) *is exponentially stable for all positive β, μ, ϵ, γ, and $\delta \geq 0$, $\theta \geq 0$.*

Let us consider a special case when $\delta = 0$, referred to as a *customer non-abandonment system*. Then, Corollaries 1.4.4, 1.4.5, 1.4.6, and 1.4.7 imply the following sufficient local stability conditions of the customer non-abandonment system.

Corollary 1.4.9. *If $\delta = 0$, $\alpha \in (0, 1)$, and $\epsilon \leq \frac{\alpha^2\mu^2}{(2-\alpha)\beta}$, then the system* (1.8) *is exponentially stable under condition*

$$\gamma > \frac{\alpha\mu}{\beta}. \tag{1.16}$$

Corollary 1.4.10. *If $\delta = 0$, $\alpha \in (0, 1)$, and $\epsilon > \frac{\alpha^2\mu^2}{(2-\alpha)\beta}$, then the system* (1.8) *is exponentially stable under condition*

$$\gamma > \sqrt{\frac{(2-\alpha)\epsilon}{\beta}}. \tag{1.17}$$

Corollary 1.4.11. *If $\delta = 0$, and $\alpha \in [0, 1)$, then the system* (1.8) *is exponentially stable under condition*

$$\gamma > \frac{\alpha\mu + \sqrt{\alpha^2\mu^2 + 8\beta\epsilon}}{2\beta}. \tag{1.18}$$

Note that if $\theta = 0$, then Corollary 1.4.11 is a simpler, equivalent version of the sufficient local stability condition in [55] for the non-abandonment system (Theorem 3 in [55]). Moreover, condition (1.10) in Theorem 1.4.2 implies the following result, which does not depend on ϵ, for the non-abandonment system.

Corollary 1.4.12. *If $\delta = 0$, and $\alpha \in [0, 1)$, then the system* (1.8) *is exponentially stable under condition*

$$\gamma > \frac{(\alpha + \sqrt{\alpha^2 + 4\alpha})\mu}{2\beta}. \tag{1.19}$$

Having a variety of these sufficient local stability conditions is useful, because some or others may be easier to verify/ensure, depending on the scenario. Note that γ and ϵ are control parameters, while all other parameters are those of the system – they can be potentially measured/estimated in real time. It is not easy to give an intuitive meaning/interpretation of the above local stability conditions. Perhaps Corollary 1.4.1 is the easiest to interpret: if magnitude γ of the system response to changes in the queue length is large enough, this is sufficient for local stability.

Remark 1.4.2. *We note that our* local *stability results apply to more general systems, exhibiting same* local *behavior. For example, suppose the total number of potential agents is not infinite, by finite, scaling with r as κr, where $\kappa > \lambda(1-\alpha)/\beta$. Then, the fluid limits of such system satisfy the same ODE (1.8) in the vicinity of the origin, and therefore our local stability results apply as is.*

1.5 Proof of Theorem 1.4.1

The proof of Theorem 1.4.1 is a generalization of the proof of Theorem 1 in [58]. However, it requires additional technical details – we present it here for completeness.

In order to prove Theorem 1.4.1, it suffices to show that w.p.1 from any subsequence of r, we can choose a further subsequence, along which a u.o.c. convergence to a fluid limit holds.

Let $N_i(\cdot)$, $i = 1, \ldots, 8$ be mutually independent unit-rate Poisson processes. N_1 is the

process which drives customer arrivals. N_2 is the process which drives the acceptance of invitations. N_3 is the process which drives the service completions with agents leaving the system. N_4 is the process which drives the service completions with agents returning the agent queue. N_5 and N_6 are the processes which drive type-3 events, when variable $Y^r(t)$ is negative and positive, respectively. N_7 is the process which drives the abandonment of customers. N_8 is the process which drives the abandonment of agents. Given the initial state $(X^r(0), Y^r(0), V^r(0))$, we construct the process $(X^r(\cdot), Y^r(\cdot), V^r(\cdot))$, for all r, on the same probability space via a common set of independent Poisson process [59] as follows:

$$X^r(t) = G^r(t) + \left(- \min_{0 \le s \le t} G^r(s) \right) \vee 0, \tag{1.20}$$

$$\begin{aligned} G^r(t) = X^r(0) &+ \gamma N_1(\lambda r t) - \gamma N_2 \left(\beta \int_0^t X^r(s) ds \right) - \gamma N_4 \left(\alpha\mu \int_0^t (V^r(s) - (Y^r(s))^+) ds \right) \\ &- \gamma N_7 \left(\delta \int_0^t (Y^r(s))^- ds \right) + \gamma N_8 \left(\theta \int_0^t (Y^r(s))^+ ds \right) + \\ &+ N_5 \left(\epsilon \int_0^t (Y^r(s))^- ds \right) - N_6 \left(\epsilon \int_0^t (Y^r(s))^+ ds \right), \end{aligned} \tag{1.21}$$

$$\begin{aligned} Y^r(t) = Y^r(0) &+ N_2 \left(\beta \int_0^t X^r(s) ds \right) - N_1(\lambda r t) + N_4 \left(\alpha\mu \int_0^t (V^r(s) - (Y^r(s))^+) ds \right) + \\ &+ N_7 \left(\delta \int_0^t (Y^r(s))^- ds \right) - N_8 \left(\theta \int_0^t (Y^r(s))^+ ds \right), \end{aligned} \tag{1.22}$$

$$\begin{aligned} V^r(t) = V^r(0) &+ N_2 \left(\int_0^t \beta X^r(s) ds \right) - N_3 \left(\int_0^t (1-\alpha)\mu (V^r(s) - (Y^r(s))^+) ds \right) - \\ &- N_8 \left(\theta \int_0^t (Y^r(s))^+ ds \right). \end{aligned} \tag{1.23}$$

W.p.1, for any r, relations (1.20)-(1.23) uniquely define the realization of $(X^r(\cdot), Y^r(\cdot), V^r(\cdot))$ via the realizations of the driving processes $N_i(\cdot)$. Relation (1.20), the "reflection" at zero, corresponds to the property that $X^r(t)$ cannot become negative.

The functional strong law of large numbers (FSLLN) holds for each Poisson process N_i:

$$\frac{N_i(rt)}{r} \to t \ , \ r \to \infty \ , \ \text{u.o.c.}, \ \text{w.p.1.} \tag{1.24}$$

We consider the sequence of associated fluid-scaled processes with centering $(\bar{X}^r(\cdot), \bar{Y}^r(\cdot), \bar{V}^r(\cdot))$ as defined in (1.5). Let a constant $m > \|(x(0), y(0), v(0)\|$ be fixed. For each

r, on the same probability space as $(\bar{X}^r(\cdot), \bar{Y}^r(\cdot), \bar{V}^r(\cdot))$, let us define a modified fluid-scaled process $(\bar{X}^r_m(\cdot), \bar{Y}^r_m(\cdot), \bar{V}^r_m(\cdot))$. Let $(\bar{X}^r_m(\cdot), \bar{Y}^r_m(\cdot), \bar{V}^r_m(\cdot))$ start from the same initial state as $(\bar{X}^r(\cdot), \bar{Y}^r(\cdot), \bar{V}^r(\cdot))$, i.e., $(\bar{X}^r_m(0), \bar{Y}^r_m(0), \bar{V}^r_m(0)) = (\bar{X}^r(0), \bar{Y}^r(0), \bar{V}^r(0))$. The modified process $(\bar{X}^r_m(\cdot), \bar{Y}^r_m(\cdot), \bar{V}^r_m(\cdot))$ follows the same path as $(\bar{X}^r(\cdot), \bar{Y}^r(\cdot), \bar{V}^r(\cdot))$ until the first time t, such that $\|(\bar{X}^r(t), \bar{Y}^r(t), \bar{V}^r(t))\| \geq m$. Denote this time by τ^r_m. We then freeze the process $(\bar{X}^r_m(\cdot), \bar{Y}^r_m(\cdot), \bar{V}^r_m(\cdot))$ at the value $(\bar{X}^r(\tau^r_m), \bar{Y}^r(\tau^r_m), \bar{V}^r(\tau^r_m))$, i.e. $(\bar{X}^r_m(t), \bar{Y}^r_m(t), \bar{V}^r_m(t)) = (\bar{X}^r(\tau^r_m), \bar{Y}^r(\tau^r_m), \bar{V}^r(\tau^r_m))$ for all $t \geq \tau^r_m$.

Lemma 1.5.1. *Fix $(x(0), y(0), v(0))$ and a finite constant $m > \|(x(0), y(0), v(0))\|$. Then, w.p.1 for any subsequence of r, there exists a further subsequence, along which $(\bar{X}^r_m, \bar{Y}^r_m, \bar{V}^r_m)$ converges u.o.c. to a Lipschitz continuous trajectory (x_m, y_m, v_m), which satisfies properties* (1.7) *at any regular time $t \geq 0$ such that $\|(x_m(t), y_m(t), v_m(t))\| < m$.*

Proof. For the modified fluid-scaled processes $(\bar{X}^r_m(\cdot), \bar{Y}^r_m(\cdot), \bar{V}^r_m(\cdot))$, we define the associated counting processes for upward and downward jumps. For $t \leq \tau^r_m$,

$$\bar{X}^{r\uparrow}_m(t) = r^{-1}\gamma N_1(\lambda r t) + r^{-1}\gamma N_8\left(\theta r \int_0^t (\bar{Y}^r_m(s))^+ ds\right) + r^{-1} N_5\left(\epsilon r \int_0^t (\bar{Y}^r_m(s))^- ds\right), \tag{1.25}$$

$$\begin{aligned}\bar{X}^{r\downarrow}_m(t) = {} & r^{-1}\gamma N_2\left(\beta r \int_0^t \left[\bar{X}^r_m(s) + \frac{\lambda(1-\alpha)}{\beta}\right] ds\right) + \\ & + r^{-1}\gamma N_4\left(\alpha\mu r \int_0^t \left[\bar{V}^r_m(s) + \frac{\lambda}{\mu} - (\bar{Y}^r_m(s))^+\right] ds\right) + \\ & + r^{-1}\gamma N_7\left(\delta r \int_0^t (\bar{Y}^r_m(s))^- ds\right) + r^{-1} N_6\left(\epsilon r \int_0^t (\bar{Y}^r_m(s))^+ ds\right),\end{aligned} \tag{1.26}$$

$$\begin{aligned}\bar{Y}^{r\uparrow}_m(t) = {} & r^{-1} N_2\left(\beta r \int_0^t \left[\bar{X}^r_m(s) + \frac{\lambda(1-\alpha)}{\beta}\right] ds\right) + \\ & + r^{-1} N_4\left(\alpha\mu r \int_0^t \left[\bar{V}^r_m(s) + \frac{\lambda}{\mu} - (\bar{Y}^r_m(s))^+\right] ds\right) + \\ & + r^{-1} N_7\left(\delta r \int_0^t (\bar{Y}^r_m(s))^- ds\right),\end{aligned} \tag{1.27}$$

$$\bar{Y}^{r\downarrow}_m(t) = r^{-1} N_1(\lambda r t) + r^{-1} N_8\left(\theta r \int_0^t (\bar{Y}^r_m(s))^+ ds\right) \tag{1.28}$$

$$\bar{V}^{r\uparrow}_m(t) = r^{-1} N_2\left(\beta r \int_0^t \left[\bar{X}^r_m(s) + \frac{\lambda(1-\alpha)}{\beta}\right] ds\right), \tag{1.29}$$

$$\bar{V}^{r\downarrow}_m(t) = r^{-1} N_3\left((1-\alpha)\mu r \int_0^t \left[\bar{V}^r_m(s) + \frac{\lambda}{\mu} - (\bar{Y}^r_m(s))^+\right] ds\right) +$$

$$+ r^{-1} N_8 \left(\theta r \int_0^t (\bar{Y}_m^r(s))^+ ds \right), \tag{1.30}$$

and for $t > \tau_m^r$, all these counting processes are frozen at their values at time τ_m^r, that is,

$$\begin{cases} \bar{X}_m^{r\uparrow}(t) = \bar{X}_m^{r\uparrow}(\tau_m^r) \ , \ \bar{X}_m^{r\downarrow}(t) = \bar{X}_m^{r\downarrow}(\tau_m^r) \ , \\ \bar{Y}_m^{r\uparrow}(t) = \bar{Y}_m^{r\uparrow}(\tau_m^r) \ , \ \bar{Y}_m^{r\downarrow}(t) = \bar{Y}_m^{r\downarrow}(\tau_m^r) \ , \\ \bar{V}_m^{r\uparrow}(t) = \bar{V}_m^{r\uparrow}(\tau_m^r) \ , \ \bar{V}_m^{r\downarrow}(t) = \bar{V}_m^{r\downarrow}(\tau_m^r). \end{cases} \tag{1.31}$$

Using the relations (1.20)-(1.23) and the fact that for $0 \leq t \leq \tau_m^r$ the original process $(\bar{X}^r, \bar{Y}^r, \bar{V}^r)$ and the modified process $(\bar{X}_m^r, \bar{Y}_m^r, \bar{V}_m^r)$ coincide, we have for all $t \geq 0$,

$$\bar{X}_m^r(t) = \bar{G}_m^r(t) + \left(-\lambda(1-\alpha)/\beta - \min_{0 \leq s \leq t} \bar{G}_m^r(s) \right) \vee 0, \tag{1.32}$$

$$\bar{G}_m^r(t) = \bar{X}^r(0) + \bar{X}_m^{r\uparrow}(t) - \bar{X}_m^{r\downarrow}(t), \tag{1.33}$$

$$\bar{Y}_m^r(t) = \bar{Y}^r(0) + \bar{Y}_m^{r\uparrow}(t) - \bar{Y}_m^{r\downarrow}(t), \tag{1.34}$$

$$\bar{V}_m^r(t) = \bar{V}^r(0) + \bar{V}_m^{r\uparrow}(t) - \bar{V}_m^{r\downarrow}(t). \tag{1.35}$$

The counting processes $\bar{X}_m^{r\uparrow}(\cdot)$, $\bar{X}_m^{r\downarrow}(\cdot)$, $\bar{Y}_m^{r\uparrow}(\cdot)$, $\bar{Y}_m^{r\downarrow}(\cdot)$, $\bar{V}_m^{r\uparrow}(\cdot)$, $\bar{V}_m^{r\downarrow}(\cdot)$ are non-decreasing. Using FSLLN (1.24) and the fact that the processes $\bar{X}_m^r(\cdot)$, $\bar{Y}_m^r(\cdot)$, and $\bar{V}_m^r(\cdot)$ are uniformly bounded by construction, we see that w.p.1. for any subsequence of r, there exists a further subsequence along which the set of trajectories $(\bar{X}_m^{r\uparrow}(\cdot), \bar{X}_m^{r\downarrow}(\cdot), \bar{Y}_m^{r\uparrow}(\cdot), \bar{Y}_m^{r\downarrow}(\cdot), \bar{V}_m^{r\uparrow}(\cdot), \bar{V}_m^{r\downarrow}(\cdot))$ converges u.o.c. to a set of non-decreasing Lipschitz continuous functions $(x_m^{\uparrow}(\cdot), x_m^{\downarrow}(\cdot), y_m^{\uparrow}(\cdot), y_m^{\downarrow}(\cdot), v_m^{\uparrow}(\cdot), v_m^{\downarrow}(\cdot))$. But then the u.o.c. convergence of $(\bar{X}_m^r(\cdot), \bar{Y}_m^r(\cdot), \bar{V}_m^r(\cdot), \bar{G}_m^r(\cdot))$ to a set of Lipschitz continuous functions $(x_m(\cdot), y_m(\cdot), v_m(\cdot), g_m(\cdot))$ holds, where

$$x_m(t) = g_m(t) + \left(-\lambda(1-\alpha)/\beta - \min_{0 \leq s \leq t} g_m(s) \right) \vee 0, \tag{1.36}$$

$$g_m(t) = x(0) + x_m^{\uparrow}(t) - x_m^{\downarrow}(t), \tag{1.37}$$

$$y_m(t) = y(0) + y_m^{\uparrow}(t) - y_m^{\downarrow}(t), \tag{1.38}$$

$$v_m(t) = v(0) + v_m^{\uparrow}(t) - v_m^{\downarrow}(t), \tag{1.39}$$

and the following holds for t before fluid trajectory hits $\|(x_m(t), y_m(t), v_m(t))\| = m$

$$x_m^{\uparrow}(t) = \gamma\lambda t + \gamma\theta \int_0^t y_m^+(s)ds + \epsilon \int_0^t y_m^-(s)ds, \tag{1.40}$$

$$x_m^{\downarrow}(t) = \gamma\beta \int_0^t \left(x_m(s) + \frac{\lambda(1-\alpha)}{\beta} \right) ds + \gamma\alpha\mu \int_0^t \left(v_m(s) + \frac{\lambda}{\mu} - y_m^+(s) \right) ds + \\ + \gamma\delta \int_0^t y_m^-(s)ds + \epsilon \int_0^t y_m^+(s)ds, \tag{1.41}$$

$$y_m^{\uparrow}(t) = \beta \int_0^t \left(x_m(s) + \frac{\lambda(1-\alpha)}{\beta} \right) ds + \alpha\mu \int_0^t \left(v_m(s) + \frac{\lambda}{\mu} - y_m^+(s) \right) ds + \delta \int_0^t y_m^-(s)ds, \tag{1.42}$$

$$y_m^{\downarrow}(t) = \lambda t + \theta \int_0^t y_m^+(s)ds, \tag{1.43}$$

$$v_m^{\uparrow}(t) = \beta \int_0^t \left(x_m(s) + \frac{\lambda(1-\alpha)}{\beta} \right) ds, \tag{1.44}$$

$$v_m^{\downarrow}(t) = (1-\alpha)\mu \int_0^t \left(v_m(s) + \frac{\lambda}{\mu} - y_m^+(s) \right) ds + \theta \int_0^t y_m^+(s)ds. \tag{1.45}$$

Hence,

$$\begin{cases} x'_m(t) = \begin{cases} -\gamma\beta x_m(t) - \gamma\alpha\mu(v_m(t) - y_m^+(t)) + \gamma\theta y_m^+(t) - \gamma\delta y_m^-(t) - \epsilon y_m(t), \\ \qquad\qquad \text{if } x_m(t) > -\frac{\lambda(1-\alpha)}{\beta}, \\ [-\gamma\beta x_m(t) - \gamma\alpha\mu(v_m(t) - y_m^+(t)) + \gamma\theta y_m^+(t) - \gamma\delta y_m^-(t) - \epsilon y_m(t)] \vee 0, \\ \qquad\qquad \text{if } x_m(t) = -\frac{\lambda(1-\alpha)}{\beta}, \end{cases} \\ y'_m(t) = \beta x_m(t) + \alpha\mu(v_m(t) - y_m^+(t)) + \delta y_m^-(t) - \theta y_m^+(t), \\ v'_m(t) = \beta x_m(t) - (1-\alpha)\mu(v_m(t) - y_m^+(t)) - \theta y_m^+(t). \end{cases} \tag{1.46}$$

It is easy to verify that, at any regular time $t \geq 0$ such that $\|(x_m(t), y_m(t), v_m(t))\| < m$, properties (1.7) hold for the trajectory $(x_m(\cdot), y_m(\cdot), v_m(\cdot))$. $\square$

Conclusion of the proof of Theorem 1.4.1. It is easy to see that

$$\frac{d}{dt}\|(x_m(t), y_m(t), v_m(t))\| \leq C\|(x_m(t), y_m(t), v_m(t))\| \text{ for any } m \text{ and some } C > 0. \tag{1.47}$$

From Gronwall's inequality [19], we have

$$\|(x_m(t), y_m(t), v_m(t))\| \leq \|(x(0), y(0), v(0))\| e^{Ct} \text{ for all } t \geq 0 \tag{1.48}$$

For a given $(x(0), y(0), v(0))$, let us fix $T_l > 0$ and choose $m_l > \|(x(0), y(0), v(0)\| e^{CT_l}$. For this $T_l > 0$, there exists a subsequence r^l, along which $(\bar{X}^r, \bar{Y}^r, \bar{V}^r)$ converges uniformly to $(x_{m_l}, y_{m_l}, v_{m_l})$, which satisfies properties (1.7), at any $t \in [0, T_l]$. The limit trajectory $(x_{m_l}, y_{m_l}, v_{m_l})$ does not hit m_l in $[0, T_l]$. Subsequence $r^l = \{r_1^l, r_2^l, \dots\}$ is such that, w.p.1, for all sufficiently large r along the subsequence r^l, $(\bar{X}^r(t), \bar{Y}^r(t), \bar{V}^r(t)) = (\bar{X}^r_{m_l}(t), \bar{Y}^r_{m_l}(t), \bar{V}^r_{m_l}(t))$ at any $t \in [0, T_l]$.

We consider a sequence $T_1, T_2, \dots, \to \infty$. We construct a subsequence r^* by using Cantor's diagonal procedure [67] from subsequences $r^1, r^2, \dots$ $(r^1 \supseteq r^2 \supseteq \dots)$ corresponding to $T_1, T_2, \dots$, respectively (i.e. $r_1^* = r_1^1$, $r_2^* = r_2^2, \dots$). Clearly, for this subsequence r^*, w.p.1, $(\bar{X}^r, \bar{Y}^r, \bar{V}^r)$ converges u.o.c. to (x, y, v), which satisfies properties (1.7), at any regular point $t \in [0, \infty)$. □

1.6 Proof of Theorem 1.4.2

In order to prove Theorem 1.4.2, it suffices to show that LTI systems of the switched linear system (1.8) have a CQLF.

The system (1.8) is a switched linear system with $m = 2$. (Note that $y^+ = y$ if $y \geq 0$ and $y^+ = 0$ if $y < 0$, and $y^- = 0$ if $y \geq 0$ and $y^- = -y$ if $y < 0$.) Namely, for $y \geq 0$,

$$\begin{cases} x'(t) = (-\gamma\beta)x(t) + (\gamma\alpha\mu + \gamma\theta - \epsilon)y(t) + (-\gamma\alpha\mu)v(t) \\ y'(t) = (\beta)x(t) + (-\alpha\mu - \theta)y(t) + (\alpha\mu)v(t) \\ v'(t) = (\beta)x(t) + ((1-\alpha)\mu - \theta)y(t) + (-(1-\alpha)\mu)v(t) \end{cases} \tag{1.49}$$

and for $y < 0$,

$$\begin{cases} x'(t) = (-\gamma\beta)x(t) + (\gamma\delta - \epsilon)y(t) + (-\gamma\alpha\mu)v(t) \\ y'(t) = (\beta)x(t) + (-\delta)y(t) + (\alpha\mu)v(t) \\ v'(t) = (\beta)x(t) + (-(1-\alpha)\mu)v(t) \end{cases} \tag{1.50}$$

We can rewrite the systems above as two LTI systems $u'(t) = A_1u(t)$ and $u'(t) = A_2u(t)$, where $u(t) = (x(t), y(t), v(t))^T$ and

$$A_1 = \begin{pmatrix} -\gamma\beta & \gamma\alpha\mu + \gamma\theta - \epsilon & -\gamma\alpha\mu \\ \beta & -\alpha\mu - \theta & \alpha\mu \\ \beta & (1-\alpha)\mu - \theta & -(1-\alpha)\mu \end{pmatrix}, \; A_2 = \begin{pmatrix} -\gamma\beta & \gamma\delta - \epsilon & -\gamma\alpha\mu \\ \beta & -\delta & \alpha\mu \\ \beta & 0 & -(1-\alpha)\mu \end{pmatrix}. \tag{1.51}$$

Lemma 1.6.1. *Matrix A_1 in* (1.51) *is Hurwitz for all positive β, γ, μ, ϵ, $\delta \geq 0$, $\theta \geq 0$ and $\alpha \in [0, 1)$.*

Proof. The characteristic equation of A_1 is

$$\lambda^3 + (\beta\gamma + \mu + \theta)\lambda^2 + (\beta\epsilon + \beta\gamma\mu + \mu\theta)\lambda + \beta\epsilon\mu = 0. \tag{1.52}$$

By Proposition 1.2.1, it suffices to verify that

$$\beta\gamma + \mu + \theta > 0 \; , \; \beta\epsilon + \beta\gamma\mu + \mu\theta > 0 \; , \; \beta\epsilon\mu > 0, \text{ and} \tag{1.53}$$

$$\begin{aligned} (\beta\gamma + \mu + \theta)(\beta\epsilon + \beta\gamma\mu + \mu\theta) - \beta\epsilon\mu = \beta^2\gamma^2\mu + \beta^2\gamma\epsilon + \beta\gamma\mu^2 + \\ +2\beta\gamma\mu\theta + \beta\theta\epsilon + \mu^2\theta + \mu\theta^2 > 0. \end{aligned} \tag{1.54}$$

The conditions (1.53) and (1.6) are obviously true. □

Lemma 1.6.2. *For positive β, γ, μ, ϵ, $\delta \geq 0$, $\theta \geq 0$ and $\alpha \in [0, 1)$, matrix A_2 in* (1.51) *is Hurwitz if and only if*

$$\left(\frac{\beta\gamma + \delta}{\mu} + (1-\alpha)\right)\left(\frac{\beta\gamma\mu + \delta\mu(1-\alpha)}{\beta\epsilon} + 1\right) > 1 \tag{1.55}$$

Proof. The characteristic equation of A_2 is

$$\lambda^3 + (\beta\gamma + \mu(1-\alpha) + \delta)\lambda^2 + (\beta\epsilon + \beta\gamma\mu + \delta\mu(1-\alpha))\lambda + \beta\epsilon\mu = 0. \tag{1.56}$$

By Proposition 1.2.1, it suffices to verify that

$$\beta\gamma + \mu(1-\alpha) + \delta > 0\ ,\ \beta\epsilon + \beta\gamma\mu + \delta\mu(1-\alpha) > 0\ ,\ \beta\epsilon\mu > 0, \tag{1.57}$$

and $(\beta\gamma + \mu(1-\alpha) + \delta)(\beta\epsilon + \beta\gamma\mu + \delta\mu(1-\alpha)) - \beta\epsilon\mu > 0$, which is equivalent to (1.55) since

$$\begin{aligned}
&(\beta\gamma + \mu(1-\alpha) + \delta)(\beta\epsilon + \beta\gamma\mu + \delta\mu(1-\alpha)) - \beta\epsilon\mu > 0 \\
\Leftrightarrow\ &(\beta\gamma + \delta + \mu(1-\alpha))(\beta\gamma\mu + \delta\mu(1-\alpha) + \beta\epsilon) > \beta\epsilon\mu \\
\Leftrightarrow\ &\left(\frac{\beta\gamma+\delta}{\mu} + (1-\alpha)\right)\left(\frac{\beta\gamma\mu + \delta\mu(1-\alpha)}{\beta\epsilon} + 1\right) > 1.
\end{aligned}$$

The conditions (1.57) are obviously true. □

It is easy to see that Lemma 1.6.2 implies the following result.

Corollary 1.6.1. *Matrix A_2 in* (1.51) *is Hurwitz if*

$$\gamma > \frac{\alpha\mu - \delta}{\beta}. \tag{1.58}$$

(Note that $\gamma > 0$ by definition.)

Lemma 1.6.3. *Matrix A_2 in* (1.51) *is Hurwitz under the condition either* (1.9) *or* (1.10).

Proof. This easily follows by applying Corollary 1.6.1.

Lemma 1.6.4. *Matrix product A_1A_2 has no negative eigenvalues under the condition either* (1.9) *or* (1.10).

Proof. With the help of MATLAB symbolic calculation, it can be shown that A_1 is

non-singular and

$$A_1^{-1} = \begin{pmatrix} -\frac{\theta}{\beta\epsilon} & -\frac{(\alpha\epsilon-\epsilon+\gamma\theta)}{\beta\epsilon} & \frac{\alpha}{\beta} \\ -\frac{1}{\epsilon} & -\frac{\gamma}{\epsilon} & 0 \\ -\frac{1}{\epsilon} & \frac{(\epsilon-\gamma\mu)}{\epsilon\mu} & -\frac{1}{\mu} \end{pmatrix}. \tag{1.59}$$

By Proposition 1.2.4, to demonstrate that the product A_1A_2 has no negative eigenvalues, it will suffice to show that $[A_1^{-1} + \tau A_2]$ is non-singular for all $\tau \geq 0$. We have

$$\det[A_1^{-1} + \tau A_2] = [\beta^2\epsilon^2\mu^2\tau^3 +$$
$$+(\beta^2\epsilon^2 + \beta^2\gamma^2\mu^2 - 2\beta\epsilon\mu^2 + \delta\mu^2\theta + \alpha\beta\epsilon\mu^2 + \beta\delta\gamma\mu^2 - \alpha\delta\mu^2\theta + \beta\gamma\mu^2\theta - \alpha\beta\delta\epsilon\mu - \alpha\beta\delta\gamma\mu^2)\tau^2$$
$$+(\mu^2 - \alpha\mu^2 + \beta^2\gamma^2 - 2\beta\epsilon + \delta\theta + \beta\delta\gamma + \alpha\delta\mu + \beta\gamma\theta - \alpha\mu\theta - \alpha\beta\gamma\mu)\tau + 1]/(-\beta\epsilon\mu). \tag{1.60}$$

To show $\det[A_1^{-1} + \tau A_2] \neq 0$ for all $\tau \geq 0$, it will suffice to show that the numerator of the ratio (1.60) is strictly positive. We can represent the numerator of the ratio (1.60) as follows.

(a) Under the condition (1.9), the numerator of the ratio (1.60) is

$$\beta^2\epsilon^2\mu^2\tau^3 + (\beta\epsilon\tau - 1)^2 +$$
$$+[(\beta^2\gamma^2\mu^2 - 2\beta\epsilon\mu^2 - \alpha\beta\delta\epsilon\mu + \alpha\beta\epsilon\mu^2) + \delta\mu^2\theta(1-\alpha) + \beta\delta\gamma\mu^2(1-\alpha) + \beta\gamma\mu^2\theta]\tau^2 +$$
$$+[\mu^2(1-\alpha) + \beta\gamma(\beta\gamma - \alpha\mu + \delta) + \alpha\delta\mu + (\beta\gamma - \alpha\mu + \delta)\theta]\tau \overset{(1.62)\text{-}(1.63)}{>} 0, \tag{1.61}$$

since the condition (1.9) implies

$$\gamma > \frac{\alpha\mu - \delta}{\beta} \Rightarrow \beta\gamma - \alpha\mu + \delta > 0, \tag{1.62}$$

$$\text{and } \gamma > \sqrt{\frac{(2-\alpha)\epsilon\mu + \alpha\epsilon\delta}{\beta\mu}} \Rightarrow \beta^2\gamma^2\mu^2 - 2\beta\epsilon\mu^2 - \alpha\beta\delta\epsilon\mu + \alpha\beta\epsilon\mu^2 > 0. \tag{1.63}$$

Hence, the numerator of the ratio (1.60) is strictly greater than 0 under the condition (1.9).

(b) Under the condition (1.10), the numerator of the ratio (1.60) is

$$(\beta\epsilon\tau-1)^2\mu^2\tau+(\beta\epsilon\tau-1)^2+$$
$$+[(\beta^2\gamma^2\mu^2-\alpha\beta\delta\epsilon\mu+\alpha\beta\epsilon\mu^2)+\delta\mu^2\theta(1-\alpha)+\beta\delta\gamma\mu^2(1-\alpha)+\beta\gamma\mu^2\theta]\tau^2+$$
$$+[(\beta^2\gamma^2-\beta\gamma(\alpha\mu-\delta)-\alpha\mu^2)+\alpha\delta\mu+(\beta\gamma-\alpha\mu+\delta)\theta]\tau \overset{(1.65)\text{-}(1.66)}{>} 0, \tag{1.64}$$

since the condition (1.10) implies

$$\gamma > \frac{\alpha\mu-\delta+\sqrt{(\alpha\mu-\delta)^2+4\alpha\mu^2}}{2\beta} > \frac{\alpha\mu-\delta}{\beta}$$
$$\Rightarrow \beta^2\gamma^2-\beta\gamma(\alpha\mu-\delta)-\alpha\mu^2>0 \text{ and } \beta\gamma-\alpha\mu+\delta>0 \tag{1.65}$$
$$\text{and } \gamma > \sqrt{\max\left\{\frac{\alpha\epsilon(\delta-\mu)}{\beta\mu},0\right\}} \Rightarrow \beta^2\gamma^2\mu^2-\alpha\beta\delta\epsilon\mu+\alpha\beta\epsilon\mu^2>0. \tag{1.66}$$

Hence, the numerator of the ratio (1.60) is strictly greater than 0 under the condition (1.10).

Therefore, A_1A_2 has no negative eigenvalues under the condition either (1.9) or (1.10). □

Conclusion of the proof of Theorem 1.4.2. By Lemma 1.6.1, A_1 is Hurwitz for all positive β, γ, μ, ϵ; $\delta \geq 0$, $\theta \geq 0$; and $\alpha \in [0,1)$. By Lemma 1.6.3, A_2 is Hurwitz under the condition either (1.9) or (1.10). It is easy to verify that the difference $A_1 - A_2$ has rank one. By Lemma 1.6.4, A_1A_2 has no negative real eigenvalues under the condition either (1.9) or (1.10). Hence, by Proposition 1.2.3, two LTI systems $u'(t) = A_1u(t)$ and $u'(t) = A_2u(t)$ have a CQLF. Therefore, the system (1.8) is exponentially stable under the condition either (1.9) or (1.10). □

1.7 Numerical and Simulation Experiments and Conjectures

In this section, we present some numerical and simulation experiments. These results are for both stylized and actual schemes, and all results are for the true system which includes boundary $X \geq 0$. We also put forward some conjectures based on these experiments.

In all simulations, we always assume $r = 1000$, but specify only the actual arrival rate $\Lambda =$

λr. On the plots labeled 'fluid', $X(t), Y(y), V(t)$ are replaced by their *fluid approximations*

$$X(t) = rx(t) + \frac{\lambda r(1-\alpha)}{\beta}, \quad Y(t) = ry(t), \quad V(t) = rv(t) + \frac{\lambda r}{\mu},$$

respectively, where $(x(\cdot), y(\cdot), v(\cdot)$ is the corresponding fluid limit.

1.7.1 Stylized Scheme

Example 1.7.1. *Consider the following set of parameters, which satisfies condition (1.9):*

$$\Lambda = 2000 \ , \ \alpha = 0.5 \ , \ \beta = 3 \ , \ \mu = 2 \ , \ \gamma = 1 \ , \ \epsilon = 1.5 \ , \ \delta = 1 \ , \ \theta = 0.1$$

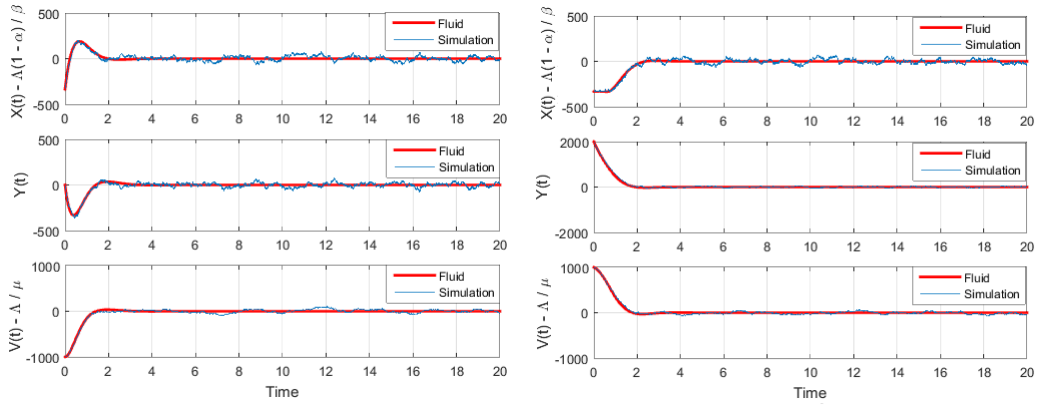
with four initial conditions: (a) $(X(0), Y(0), Z(0)) = (0, 0, 0)$; (b) $(X(0), Y(0), Z(0)) = (0, 2000, 0)$; (c) $(X(0), Y(0), Z(0)) = (2000, -2000, 1000)$; (d) $(X(0), Y(0), Z(0)) = (2000, 4000, 1000)$. The red line of the figure is the fluid approximation and the blue one is the simulation experiment. We see the converging trajectories on the Figure 1.3. Note that Figures 1.3b and 1.3d show that the trajectory hits the boundary on X. We also did the numerical/simulation experiments with many different sets of parameters satisfying the condition (1.9). All results, including those not shown on Figure 1.3, suggest the global stability of the system.

Example 1.7.2. *We use sets of parameters:*

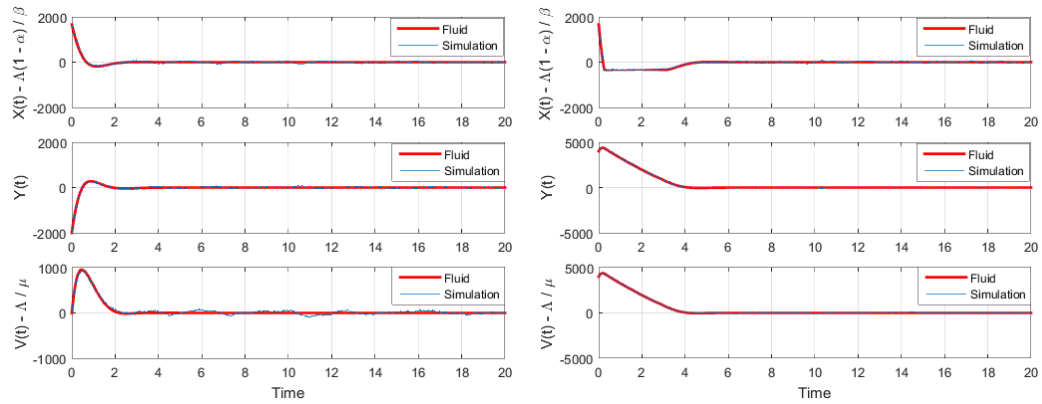
$$\Lambda = 2000 \ , \ \alpha = 0.9 \ , \ \beta = 0.05 \ , \ \mu = 0.5 \ , \ \epsilon = 1 \ , \ \delta = 0.01 \ , \ \theta = 0.01$$

with four different values of γ ($\gamma_1 = 1$, $\gamma_2 = 5$, $\gamma_3 = 10$, and $\gamma_4 = 20$) (Figure 1.4). The sets of parameters with $\gamma_1 = 1$ and $\gamma_2 = 5$ do not satisfy the condition (1.9) while the sets of parameters with $\gamma_3 = 10$ and $\gamma_4 = 20$ satisfy the condition (1.9). We consider an initial condition $(X(0), Y(0), Z(0)) = (1000, 6000, 2000)$. On the Figures 1.4b, 1.4c and 1.4d, we see that the trajectories converge. However, Figure 1.4a shows the trajectory that never converges under the set of parameters with $\gamma_1 = 1$.

With many numerical/simulation experiments, the results, including those not shown on Figure 1.4, suggest both local and global stability of the system for all sufficiently large γ.



(a) $(X(0), Y(0), Z(0)) = (0, 0, 0)$ (b) $(X(0), Y(0), Z(0)) = (0, 2000, 0)$



(c) $(X(0), Y(0), Z(0)) = (2000, -2000, 1000)$ (d) $(X(0), Y(0), Z(0)) = (2000, 4000, 1000)$

Figure 1.3: Stylized scheme: Comparison of fluid approximations with simulations in Example 1.7.1

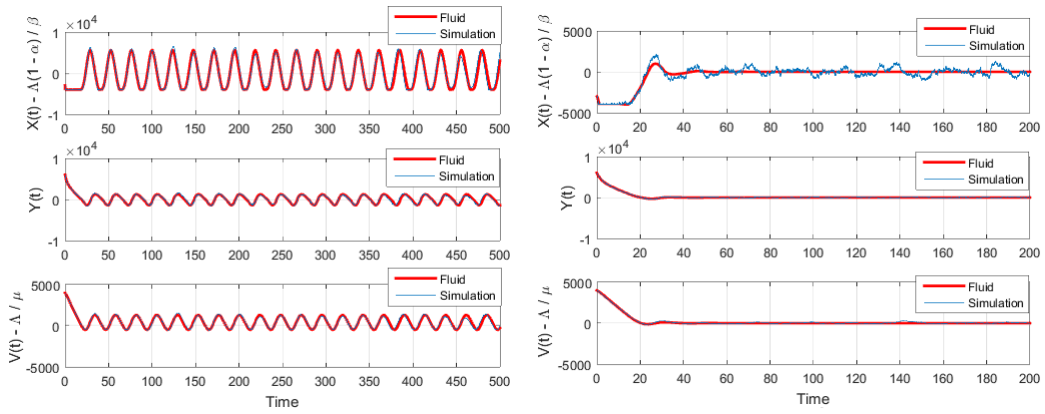
Our simulation experiments show that the fluid trajectory provides a very good approximation for the behavior of stylized scheme.

1.7.2 Actual Scheme

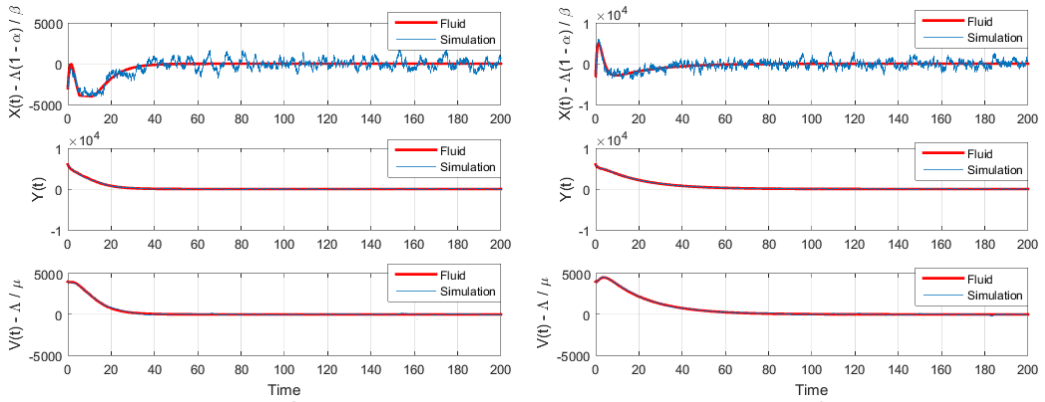
Example 1.7.3. *We conduct a simulation experiment for the actual scheme with the same set of parameters as in Example 1.7.1:*

$$\Lambda = 2000 \ , \ \alpha = 0.5 \ , \ \beta = 3 \ , \ \mu = 2 \ , \ \gamma = 1 \ , \ \epsilon = 1.5 \ , \ \delta = 1 \ , \ \theta = 0.1$$

with two initial conditions $(X(0), Y(0), Z(0), X_{target}(0)) = (0, 0, 0, 0)$ and $(X(0), Y(0), Z(0), X_{target}(0)) = (0, 0, 0, 1000)$. (Note that this set of parameters satisfies the condition (1.9).) The results are shown in Figures 1.5 and 1.6. We see that the magnitude of the difference between X_{target} and the actual number of invited agents X is very small (except



(a) $\gamma = 1$ (b) $\gamma = 5$



(c) $\gamma = 10$ (d) $\gamma = 20$

Figure 1.4: Stylized scheme: Comparison of fluid approximations with simulations in Example 1.7.2

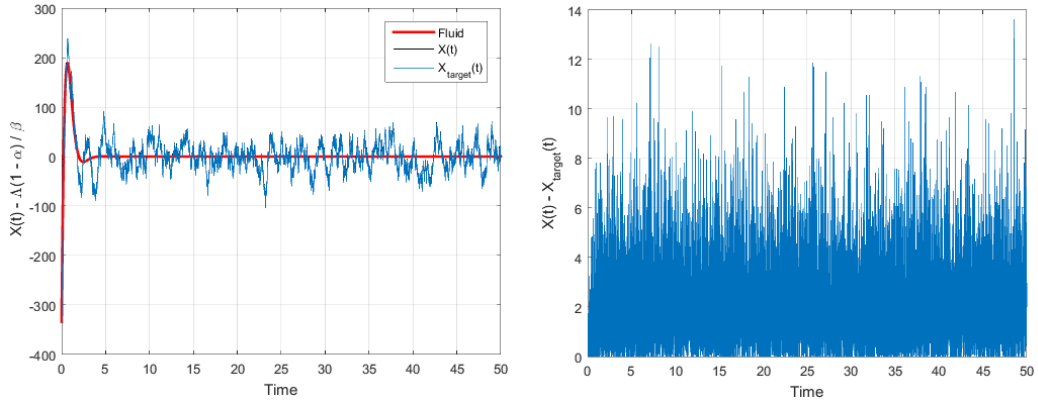
at time 0) and can be negligible compared to their values. This explains why the trajectories of X_{target} and X are well approximated by the fluid trajectory, obtained for the stylized scheme.

For the stylized scheme, the results suggest the global stability of our system for all sufficiently large γ. However, the problem with large γ is that the behavior of the stylized scheme may significantly deviate from the behavior of the actual scheme, as illustrated by the following example.

Example 1.7.4. *Consider the following set of parameters:*

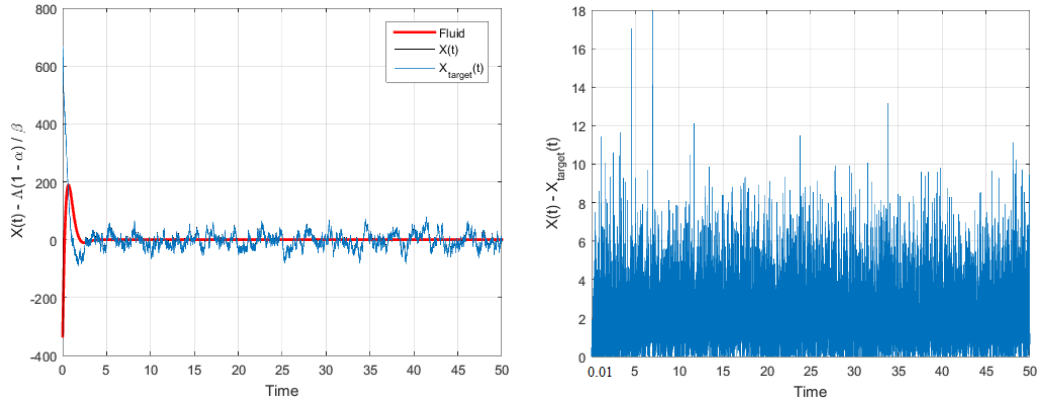
$$\Lambda = 2000 \ , \ \alpha = 0.7 \ , \ \beta = 0.5 \ , \ \mu = 3 \ , \ \epsilon = 1 \ , \ \delta = 1 \ , \ \theta = 2$$

with two values of γ ($\gamma_1 = 10$, $\gamma_2 = 20$); and an initial condition $(X(0), Y(0), Z(0), X_{target}(0)) = (0, 0, 0, 1000)$ (Figure 1.7). These results show that the behavior of the actual



(a) Fluid vs. $X(t)$ and $X_{target}(t)$ (b) $X(t) - X_{target}(t)$

Figure 1.5: Actual scheme: $(X(0), Y(0), Z(0), X_{target}(0)) = (0, 0, 0, 0)$



(a) Fluid vs. $X(t)$ and $X_{target}(t)$ (b) $X(t) - X_{target}(t)$

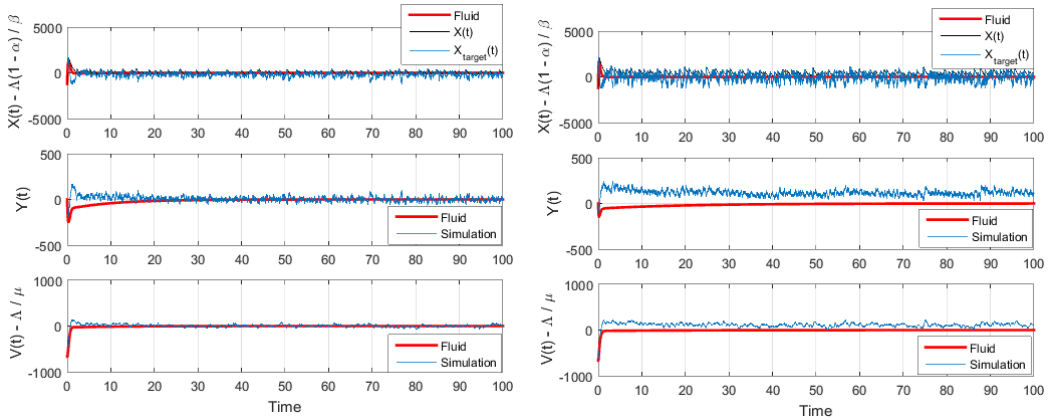
Figure 1.6: Actual scheme: $(X(0), Y(0), Z(0), X_{target}(0)) = (0, 0, 0, 1000)$

scheme deviates substantially from the behavior of the fluid trajectory with large γ.

Since $\alpha\mu > \delta$ and $\epsilon \leq \frac{(\alpha\mu-\delta)^2\mu}{(2-\alpha)\mu\beta+\alpha\delta\beta}$, then we choose $\gamma = 2.3$ such that $\gamma > \frac{\alpha\mu-\delta}{\beta}$ (Corollary 1.4.4). We can see that, with a "good" value of γ, the behavior of the actual scheme deviates negligibly from the behavior of the fluid trajectory (Figure 1.8a) and the difference between X_{target} and X is not large compared to their values (Figure 1.8b).

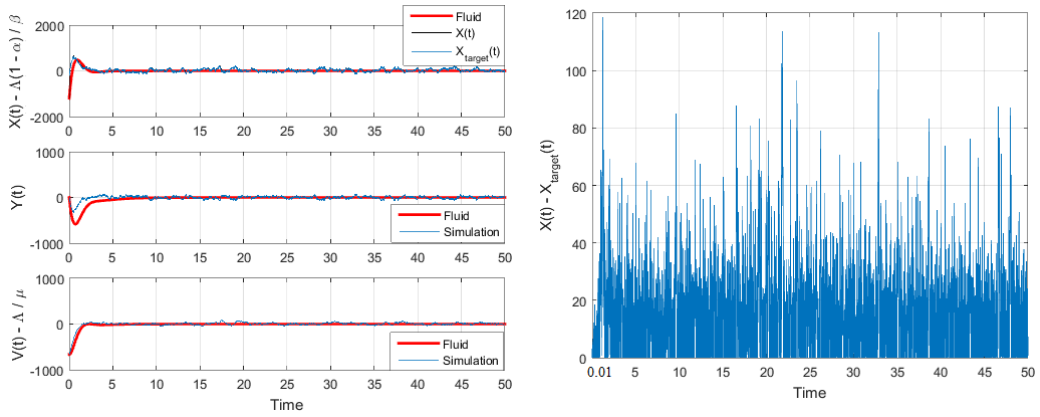
1.7.3 Global vs. Local Stability of Fluid Limits

In this chapter, we have derived some sufficient local stability conditions for the fluid limits. Based on a variety of simulation experiments above for the stylized scheme, we conjecture that local stability is sufficient for global stability of fluid limits for our model. In the next example, we compare the behavior of fluid limits for the system without boundary (given by (1.8)) with that of the system with boundary (given by (1.7)).



(a) $\gamma = 10$ (b) $\gamma = 20$

Figure 1.7: Problem with large γ of the actual scheme



(a) $\gamma = 2.3$ (b) $X(t) - X_{target}(t)$

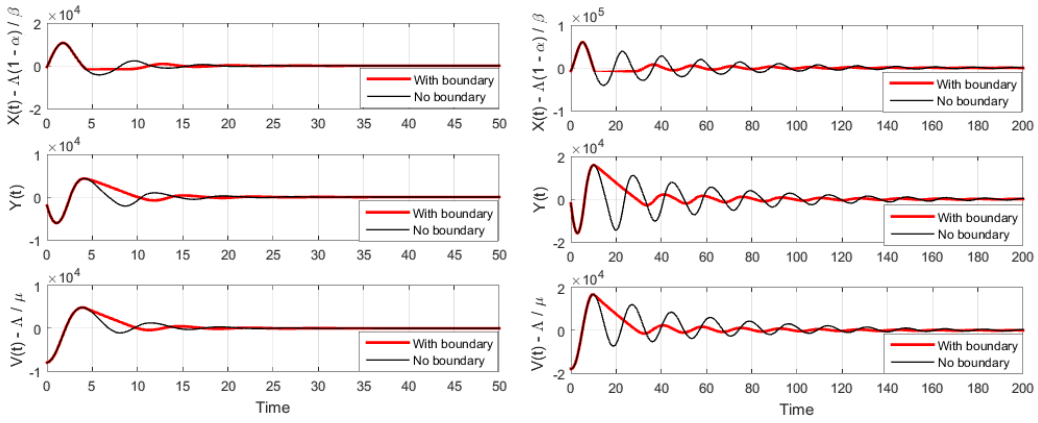
Figure 1.8: A "good" value of γ for the actual scheme

Example 1.7.5. *Consider two set of parameters, which satisfy the local stability conditions, so that the trajectory of the system (1.8) converges to the equilibrium point* $(0, 0, 0)$ *(Figure 1.9). The red line of the figure is the trajectory of the system (1.7), which may hit the boundary* $X = 0$*, and the black one is the trajectory of the system (1.8), for which there is no boundary.*

With many experiments, the results, including those not shown in Figure 1.9, further suggest the global stability of the fluid limits, when the local stability holds.

1.7.4 Summary of Conjectures, based on Numerical and Simulation Experiments.

Conjecture 1.7.1. *Our system is globally stable if it is locally stable.*



(a) Set 1 (b) Set 2

Figure 1.9: Fluid trajectories of the systems (1.7) and (1.8)

Conjecture 1.7.2. *Given all other parameters are fixed, our system is globally stable for all sufficiently large* γ.

Obviously, Conjecture 1.7.1 is stronger than Conjecture 1.7.2 because we have proved the local stability when γ is large in this chapter. We note again, however, that in a practical application the value of γ should not be made too large, because the stylized scheme behavior, which we studied in this chapter, may substantially deviate from the behavior of the actual scheme, where uninviting pending agents are not allowed.

1.8 Discussion and Further Work

In this chapter, we study a feedback-based agent invitation scheme for a model with randomly behaving agents and possible abandonment of customers and agents. This model is motivated by a variety of existing and emerging applications. The focus of the chapter is on the stability properties of the system fluid limits, arising as asymptotic limits of the system process, when the system scale (customer arrival rate) grows to infinity. The dynamic system, describing the behavior of fluid limit trajectories has a very complex structure – it is a switched linear system, which in addition has a reflecting boundary. We derived some sufficient local stability conditions, using the machinery of switched linear systems and common quadratic Lyapunov functions. Our simulation and numerical experiments show good overall performance of the feedback scheme, when the local stability conditions hold. They also suggest that, for our model, the local stability is in fact sufficient for the global

stability of fluid limits. Verifying these conjectures, as well as expanding the sufficient local stability conditions, is an interesting subject for future research. Further generalizations of the agent invitation model are also of interest from both theoretical and practical points of view.

Part II

Stochastic Gradient Algorithms

Chapter 2

When Does the Stochastic Gradient Algorithm Work Well?

In this chapter, we consider a general stochastic optimization problem which is often at the core of supervised learning, such as deep learning and linear classification. We consider a standard stochastic gradient descent (SGD) method with a fixed, large step size and propose a novel assumption on the objective function, under which this method has improved convergence rates (to a neighborhood of the optimal solutions). We then empirically demonstrate that these assumptions hold for logistic regression and standard deep neural networks on classical data sets. Thus our analysis helps to explain when efficient behavior can be expected from the SGD method in training classification models and deep neural networks.

2.1 Introduction and Motivation

In this chapter, we are interested in analyzing the behavior of the stochastic gradient algorithm when solving empirical and expected risk minimization problems. For the sake of generality we consider the following stochastic optimization problem

$$\min_{w \in \mathbb{R}^d} \{F(w) = \mathbb{E}[f(w; \xi)]\}, \tag{2.1}$$

where ξ is a random variable obeying some distribution.

In the case of empirical risk minimization with a training set $\{(x_i, y_i)\}_{i=1}^n$, ξ_i is a realization of a random variable that is defined by the i-th element of the training set. Then, by defining $f_i(w) := f(w; \xi_i)$ we write the empirical risk minimization as follows:

$$\min_{w \in \mathbb{R}^d} \left\{ F(w) = \frac{1}{n} \sum_{i=1}^{n} f_i(w) \right\}. \tag{2.2}$$

More generally ξ can be a random variable defined by a random subset of samples $\{(x_i, y_i)\}_{i \in I}$ drawn from the training set, in which case formulation (2.1) still applies to the empirical risk minimization. On the other hand, if ξ represents a sample or a set of samples drawn from the data distribution, then (2.1) represents the expected risk minimization.

Stochastic gradient descent (SGD), originally introduced in [66], has become the method of choice for solving not only (2.1) but also (2.2) when n is large. Theoretical justification for using SGD for machine learning problems is given, for example, in [12], where it is shown that, at least for convex problem, SGD is an optimal method for minimizing expected risk, which is the ultimate goal of learning. From the practical perspective SGD is often preferred to the standard gradient descent (GD) method simply because GD requires computation of a full gradient on each iteration, which, for example, in the case of deep neural networks (DNN), requires applying backpropagation for all n samples, which can be prohibitive.

Consequently, due to its simplicity in implementation and efficiency in dealing with large scale datasets, SGD has become by far the most common method for training deep neural networks and other large scale ML models. However, it is well known that SGD can be slow and unreliable in some practical applications as its behavior is strongly dependent on the chosen stepsize and on the variance of the stochastic gradients. While the method may provide fast initial improvement, it may slow down drastically after a few epochs and can even fail to move close enough to a solution for a fixed learning rate. To overcome this oscillatory behavior, several variants of SGD have been recently proposed. For example, methods such as AdaGrad [18], RMSProp [78], and Adam [30] adaptively select the stepsize for each component of w. Other techniques include diminishing stepsize scheme [13] and variance reduction methods [68, 16, 28, 51]. These latter methods reduce the variance of

the stochastic gradient estimates, by either computing a full gradient after a certain number of iterations or by storing the past gradients, both of which can be expensive. Moreover, these methods only apply to the finite sum problem (2.2) but not the general problem (2.1). On the other hand these methods enjoy faster convergence rates than that of SGD. For example, when $F(w)$ is strongly convex, convergence rates of the variance reduction methods (as well as that of GD itself) are linear, while for SGD it is only sublinear. While GD has to compute the entire gradient on *every* iteration, which makes it more expensive than the variance reduction methods, its convergence analysis allows for a much larger fixed stepsizes than those allowed in the variance reduction methods. In this chapter we are particularly interested in addressing an observation: a simple SGD with a fixed, reasonably large, step size can have a fast convergence rate to some neighborhood of the optimal solutions, without resorting to additional procedures for variance reduction.

Let us consider an example of recovering a signal $\hat{w} \in \mathbb{R}^2$ from n noisy observations $y_i = y_i^{\text{clean}} + e_i$ where $y_i^{\text{clean}} = (a_i^\mathsf{T} \hat{w})^2$. Here, a_i's are random vectors and e_i's are noise components. To recover $\hat{w}$ from the observation vector y, we solve a non-convex fourth-order polynomial minimization problem

$$\min_w \left\{ F(w) = \frac{1}{n} \sum_{i=1}^{n} (y_i - (a_i^\mathsf{T} w)^2)^2 \right\}.$$

Note that there are at least two global solutions to this problem, which we denote w_* and $-w_*$. We consider two possible scenarios:

(i) *All* of the component functions $f_i(w) = (y_i - (a_i^\mathsf{T} w)^2)^2$ have *relatively small* gradients at both of the optimal solutions w_* and $-w_*$ of the aggregate $F(w)$. In this case this means that w_* recovers a good fit for the observations y.

(ii) There are *many* indices i such that at the optimal solutions of $F(w)$, the associated gradients ∇f_i are *large*. This happens when w_* does not provide a good fit, which can happen when the noise e_i is large.

We set $n = 100$ and generate these two scenarios by setting all the noise components e_i to be small (1% of the energy of y^{clean}) for case (i) or setting only first 40 noise components to

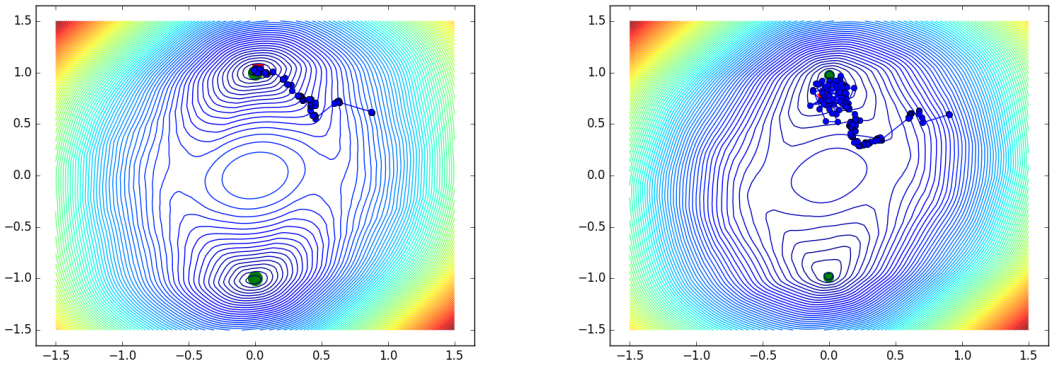


Figure 2.1: Stochastic Gradient Descent

be large (25% of the energy of y^{clean}) for case (ii). We can observe from Figure 2.1 that SGD algorithm converges to the optimal solution of $F(w)$ in case (i) depicted in the left figure; but fails to converge to the solution of F in case (ii) as shown in the right figure. The intuition behind this behavior is as follows. At every step of SGD, the iterate moves towards to the optimal solutions of the individual component function that has been randomly chosen on this iteration. If a majority of component functions f_i have their optimal solutions close to the optimum of the entire problem F, then SGD effectively acts as GD. On the other hand, if the optimal solutions of a lot of $f_i's$ are far from each other and from the overall optimum, then iterates of SGD wander randomly in the region around these individual optima, as shown on the right of Figure 2.1. Hence, SGD cannot work effectively in case (ii), unless we either reduce the learning rate or reduce the variance of the steps thus attaining more accurate gradient information.

In this chapter we generalize this result for stochastic problem (2.1) under much weaker assumptions. In particular, we do not assume that the gradients *vanish* at the solution, but that they are bounded by some small constant. Moreover, we do not impose this property on *all* stochastic gradients, but assume that it holds with suitably large probability. We then show that SGD has fast convergence rates in the strongly convex, convex and nonconvex cases, until some accuracy is reached, where this accuracy is dictated by the behavior of the stochastic gradients at the optimal solution.

We conjecture that success of SGD for training many machine learning models is the result of the associated optimization problems having this properties - most of the component

gradients are suitably small at the solution. To verify this claim, we trained linear classifiers (via logistic regression) and standard neural networks on several well-known datasets and subsequently computed the fraction of individual gradients $\nabla f_i(w_*)$ at the final solution w_* of F, that were small. The results show that more than 99% of component functions f_i have the vanishing gradient at w_*. More numerical evidence is presented in the Section 2.3.

Hence we base our analysis on the following observation.

Main observation. *For many classification problems in supervised learning, majority of component functions f_i have small gradients at the optimal solution w_* (in the convex case) or at local minima of $F(w)$ (in the nonconvex case)*

In this chapter, based on this observation, we provide theoretical analysis of SGD under the assumption on the fraction of components with small gradient at the solution. Our analysis helps explain the good performance of SGD when applied to deep learning. We summarize the key contributions of the chapter as follows.

- We conjecture that in many instances of empirical risk minimization and expected risk minimization SGD converges to a neighborhood of a stationary point of $F(w)$ such that the majority of component functions f_i have small gradients at that point. We verify numerically that this conjecture holds true for logistic regression and standard deep neural networks on a wide range of data sets.

- We formalize this conjecture as a condition under which we are able to establish improved convergence rates of SGD with fixed, large step size to a neighborhood of such stationary point when $F(w)$ is strongly convex, convex and nonconvex.

- Thus we establish that SGD converges fast to a neighborhood of the expected/empirical risk minimizer and that the size of the neighborhood is determined by some properties of the distribution of the stochastic gradient at the minimizer.

The remainder of the chapter is organized as follows. The main convergence analysis for all three cases is carried out in Section 2.2. The computational evidence is presented in Section 2.3 and implications of our analysis and findings are summarized in Section 2.4. The proofs are presented in the Appendix.

2.2 Convergence Analyses of Stochastic Gradient Algorithms

In this section, we analyze the stochastic gradient descent algorithm (SGD) under a novel condition, based on the observations of the previous section, and derive improved convergence rates for the strongly convex, convex, and non-convex cases. We present each result in the form of a general theorem with the bound on a certain optimality measure (depending on the case), followed by the corollary where we demonstrate that improved convergence rate can be observed until this optimality measure becomes small. The rate and the threshold for optimality measure are dictated by the properties of the stochastic gradient at the solution.

First we introduce the basic definition of L-smoothness.

Definition 2.2.1. *A function ϕ is L-smooth if there exists a constant $L > 0$ such that*

$$\|\nabla\phi(w) - \nabla\phi(w')\| \leq L\|w - w'\|, \ \forall \ w, w' \in \mathbb{R}^d. \tag{2.3}$$

For completeness, we state the SGD algorithm as Algorithm 1.

Algorithm 1 Stochastic Gradient Descent (SGD) Algorithm with fixed step size

Initialize w_0, **choose** stepsize $\eta > 0$, and batch size b.
for $t = 0, 1, 2, \ldots$ **do**
Generate realizations of random variables $\{\xi_{t,i}\}_{i=1}^b$ i.i.d. with $\mathbb{E}[\nabla f(w_t; \xi_{t,i})|\mathcal{F}_t] = \nabla F(w_t)$.
Compute a stochastic gradient

$$g_t = \frac{1}{b}\sum_{i=1}^{b} \nabla f(w_t; \xi_{t,i}).$$

Update the new iterate $w_{t+1} = w_t - \eta g_t$.
end for

Let $\mathcal{F}_t = \sigma(w_0, w_1, \ldots, w_t)$ be the σ-algebra generated by $w_0, w_1, \ldots, w_t$. We note that $\{\xi_{t,i}\}_{i=1}^b$ are independent of $\mathcal{F}_t$. Since $\{\xi_{t,i}\}_{i=1}^b$ are i.i.d.[1] with $\mathbb{E}[\nabla f(w_t; \xi_{t,i})|\mathcal{F}_t] = \nabla F(w_t)$, we have an unbiased estimate of gradient $\mathbb{E}[g_t|\mathcal{F}_t] = \frac{1}{b}\sum_{i=1}^b \nabla F(w_t) = \nabla F(w_t)$.

We now define the quantities that will be useful in our results.

[1]Independent and identically distributed random variables. We note from probability theory that if $X_1, \ldots, X_d$ are i.i.d. random variables then $g(X_1), \ldots, g(X_d)$ are also i.i.d. random variables if g is measurable function.

Definition 2.2.2. *Let w_* be a stationary point of the objective function $F(w)$. For any given threshold $\epsilon > 0$, define*

$$p_\epsilon := \mathbb{P}\left\{\|g_*\|^2 \leq \epsilon\right\}, \tag{2.4}$$

where $g_* = \frac{1}{b}\sum_{i=1}^{b} \nabla f(w_*; \xi_i)$, as the probability that event $\|g_*\|^2 \leq \epsilon$ happens for some i.i.d. random variables $\{\xi_i\}_{i=1}^b$. We also define

$$M_\epsilon := \mathbb{E}\left[\|g_*\|^2 \mid \|g_*\|^2 > \epsilon\right]. \tag{2.5}$$

The quantity p_ϵ measures the probability that event $\|g_*\|^2 \leq \epsilon$ happens for some realizations of random variables ξ_i, $i = 1, \ldots, b$. Clearly, p_ϵ is bounded above by 1 and monotonically increasing with respect to ϵ. Quantity M_ϵ can be interpreted as the average bound of large components $\|\nabla f(w_*; \xi)\|^2$. As we will see in our results below, quantities p_ϵ and M_ϵ appear in the convergence rate bound of the SGD algorithm. M_ϵ is also bounded above by $M_{max} = \max_\xi \|\nabla f(w_*; \xi)\|^2$, which we assume is finite, hence in all our results we can replace M_ϵ by M_{max} if we want to eliminate its dependence on ϵ. On the other hand, the dependence of quantity p_ϵ on ϵ is key for our analysis. Based on the evidence shown in Section 2.3, we expect p_ϵ to be close to 1 for all but very small values of ϵ. We will derive our convergence rate bounds in terms of $\max\{\epsilon, 1 - p_\epsilon\}$. Clearly, as ϵ decreases, $1 - p_\epsilon$ increases and vice versa, but if there exists a small ϵ for which $1 - p_\epsilon \approx \epsilon$ then our results show convergence of SGD to an $\mathcal{O}(\epsilon)$ neighborhood of the solution, at an improved rate with respect to ϵ.

2.2.1 Useful Lemmas

Let $\{\xi_i\}_{i=1}^b$ be i.i.d. random variables with $\mathbb{E}[f(w; \xi_i)] = F(w)$. From Definition 2.2.2, we have

$$\begin{aligned}\mathbb{E}\left[\|g_*\|^2\right] &= \mathbb{E}\left[\|g_*\|^2 \mid \|g_*\|^2 \leq \epsilon\right] \cdot \mathbb{P}\left\{\|g_*\|^2 \leq \epsilon\right\} + \mathbb{E}\left[\|g_*\|^2 \mid \|g_*\|^2 > \epsilon\right] \cdot \mathbb{P}\left\{g_*\|^2 > \epsilon\right\} \\ &\leq p_\epsilon \epsilon + (1 - p_\epsilon) M_\epsilon, \end{aligned} \tag{2.6}$$

where $g_* = \frac{1}{b}\sum_{i=1}^{b} \nabla f(w_*; \xi_i)$.

Lemma 2.2.1 ([49]). *Suppose that ϕ is L-smooth. Then,*

$$\phi(w) \leq \phi(w') + \nabla\phi(w')^T(w - w') + \frac{L}{2}\|w - w'\|^2, \ \forall w, w' \in \mathbb{R}^d. \tag{2.7}$$

Lemma 2.2.2 ([49]). *Suppose that ϕ is L-smooth and convex. Then,*

$$(\nabla\phi(w) - \nabla\phi(w'))^T(w - w') \geq \frac{1}{L}\|\nabla\phi(w) - \nabla\phi(w')\|^2, \ \forall w, w' \in \mathbb{R}^d. \tag{2.8}$$

Lemma 2.2.3 ([49]). *Suppose that ϕ is L-smooth and convex. Then,*

$$\|\nabla\phi(w)\|^2 \leq 2L(\phi(w) - \phi(w_*)), \forall w \in \mathbb{R}^d, \tag{2.9}$$

where $w_* = \arg\min_w \phi(w)$.

Lemma 2.2.4 ([49]). *Suppose that ϕ is μ-strongly convex. Then,*

$$2\mu[\phi(w) - \phi(w_*)] \leq \|\nabla\phi(w)\|^2 \ , \ \forall w \in \mathbb{R}^d, \tag{2.10}$$

where $w_* = \arg\min_w \phi(w)$.

Lemma 2.2.5 ([28]). *Suppose that $f(w; \xi)$ is L-smooth and convex for every realization of ξ. Then,*

$$\mathbb{E}[\|\nabla f(w; \xi) - \nabla f(w_*; \xi)\|^2] \leq 2L[F(w) - F(w_*)], \ \forall w \in \mathbb{R}^d, \tag{2.11}$$

where ξ is a random variable, and $w_* = \arg\min_w F(w)$.

Proof. Given any ξ, for all $w \in \mathbb{R}^d$, consider

$$h(w; \xi) := f(w; \xi) - f(w_*; \xi) - \nabla f(w_*; \xi)^T(w - w_*).$$

Since $h(w;\xi)$ is convex by w and $\nabla h(w_*;\xi) = 0$, we have $h(w_*;\xi) = \min_w h(w;\xi)$. Hence,

$$\begin{aligned} 0 = h(w_*;\xi) &\leq \min_{\eta} \left[h(w - \eta \nabla h(w;\xi);\xi)\right] \\ &\overset{(2.7)}{\leq} \min_{\eta} \left[h(w;\xi) - \eta \|\nabla h(w;\xi)\|^2 + \frac{L\eta^2}{2}\|\nabla h(w;\xi)\|^2\right] \\ &= h(w;\xi) - \frac{1}{2L}\|\nabla h(w;\xi)\|^2. \end{aligned}$$

Hence,

$$\|\nabla f(w;\xi) - \nabla f(w_*;\xi)\|^2 \leq 2L[f(w;\xi) - f(w_*;\xi) - \nabla f(w_*;\xi)^T(w - w_*)].$$

Taking the expectation with respect to ξ, we have

$$\mathbb{E}[\|\nabla f(w;\xi) - \nabla f(w_*;\xi)\|^2] \leq 2L[F(w) - F(w_*)].$$

□

2.2.2 Convex Objectives

In this section, we analyze the SGD method in the context of minimizing a convex objective function. We will bound the expected optimality gap at a given iterate in terms of the value of p_ϵ. First, we consider the case when F is strongly convex.

Definition 2.2.3. *A function ϕ is μ-strongly convex if there exists a constant $\mu > 0$ such that*

$$\phi(w) - \phi(w') \geq \nabla\phi(w')^T(w - w') + \frac{\mu}{2}\|w - w'\|^2, \ \forall w, w' \in \mathbb{R}^d. \tag{2.12}$$

Using this definition, we state the following result for the strongly convex case.

Theorem 2.2.1. *Suppose that $F(w)$ is μ-strongly convex and $f(w;\xi)$ is L-smooth and*

convex for every realization of ξ. Consider Algorithm 1 with $\eta \leq \frac{1}{L}$. Then, for any $\epsilon > 0$

$$\mathbb{E}[\|w_t - w_*\|^2] \leq (1 - \mu\eta(1-\eta L))^t \|w_0 - w_*\|^2 + \frac{2\eta}{\mu(1-\eta L)} p_\epsilon \epsilon + \frac{2\eta}{\mu(1-\eta L)}(1-p_\epsilon) M_\epsilon, \tag{2.13}$$

where $w_* = \arg\min_w F(w)$, and p_ϵ and M_ϵ are defined in (2.4) and (2.5), respectively.

Proof. We have

$$\begin{aligned}
&\|w_{t+1} - w_*\|^2 = \|w_t - \eta g_t - w_*\|^2 \\
&= \|w_t - w_*\|^2 - 2\eta g_t^T (w_t - w_*) + \eta^2 \|g_t\|^2 \\
&= \|w_t - w_*\|^2 - 2\eta \frac{1}{b} \sum_{i=1}^b \nabla f(w_t; \xi_{t,i})^T (w_t - w_*) + \eta^2 \left\| \frac{1}{b} \sum_{i=1}^b \nabla f(w_t; \xi_{t,i}) \right\|^2 \\
&\leq \|w_t - w_*\|^2 - 2\eta \frac{1}{b} \sum_{i=1}^b \nabla f(w_t; \xi_{t,i})^T (w_t - w_*) + 2\eta^2 \left\| \frac{1}{b} \sum_{i=1}^b (\nabla f(w_t; \xi_{t,i}) - \nabla f(w_*; \xi_{t,i})) \right\|^2 \\
&\qquad + 2\eta^2 \left\| \frac{1}{b} \sum_{i=1}^b \nabla f(w_*; \xi_{t,i}) \right\|^2 \\
&\leq \|w_t - w_*\|^2 - 2\eta \frac{1}{b} \sum_{i=1}^b \nabla f(w_t; \xi_{t,i})^T (w_t - w_*) + 2\eta^2 \frac{1}{b} \sum_{i=1}^b \|\nabla f(w_t; \xi_{t,i}) - \nabla f(w_*; \xi_{t,i})\|^2 \\
&\qquad + 2\eta^2 \left\| \frac{1}{b} \sum_{i=1}^b \nabla f(w_*; \xi_{t,i}) \right\|^2 \qquad (2.14) \\
&\overset{(2.8)}{\leq} \|w_t - w_*\|^2 - 2\eta \frac{1}{b} \sum_{i=1}^b \nabla f(w_t; \xi_{t,i})^T (w_t - w_*) \\
&\qquad + 2\eta^2 L \frac{1}{b} \sum_{i=1}^b (\nabla f(w_t; \xi_{t,i}) - \nabla f(w_*; \xi_{t,i}))^T (w_t - w_*) + 2\eta^2 \left\| \frac{1}{b} \sum_{i=1}^b \nabla f(w_*; \xi_{t,i}) \right\|^2 .
\end{aligned}$$

Hence, by taking the expectation, conditioned on $\mathcal{F}_t = \sigma(w_0, w_1, \ldots, w_t)$ (which is the σ-algebra generated by $w_0, w_1, \ldots, w_t$), we have

$$\begin{aligned}
\mathbb{E}[\|w_{t+1} - w_*\|^2 | \mathcal{F}_t] &\leq \|w_t - w_*\|^2 - 2\eta(1-\eta L) \nabla F(w_t)^T (w_t - w_*) \\
&\qquad + 2\eta^2 \mathbb{E}\left[\left\| \frac{1}{b} \sum_{i=1}^b \nabla f(w_*; \xi_{t,i}) \right\|^2 \Big| \mathcal{F}_t \right] \\
&\overset{(2.12)}{\leq} (1 - \mu\eta(1-\eta L)) \|w_t - w_*\|^2 - 2\eta(1-\eta L)[F(w_t) - F(w_*)]
\end{aligned}$$

$$+ 2\eta^2 \mathbb{E}\left[\left\|\frac{1}{b}\sum_{i=1}^{b}\nabla f(w_*;\xi_{t,i})\right\|^2\right]$$
$$\overset{\eta\leq 1/L,(2.6)}{\leq} (1-\mu\eta(1-\eta L))\|w_t - w_*\|^2 + 2\eta^2 p_\epsilon \epsilon + 2\eta^2(1-p_\epsilon)M_\epsilon.$$

The first inequality follows since

$$\mathbb{E}\left[\frac{1}{b}\sum_{i=1}^{b}\nabla f(w_t;\xi_{t,i})\Big|\mathcal{F}_t\right] = \mathbb{E}\left[\frac{1}{b}\sum_{i=1}^{b}\left(\nabla f(w_t;\xi_{t,i}) - \nabla f(w_*;\xi_{t,i})\right)\Big|\mathcal{F}_t\right] = \nabla F(w_t).$$

We note in the second equality that $\mathbb{E}\left[\left\|\frac{1}{b}\sum_{i=1}^{b}\nabla f(w_*;\xi_{t,i})\right\|^2\Big|\mathcal{F}_t\right] = \mathbb{E}\left[\left\|\frac{1}{b}\sum_{i=1}^{b}\nabla f(w_*;\xi_{t,i})\right\|^2\right]$ since $\xi_{t,i}$ is independent of $\mathcal{F}_t$. By taking the expectation for both sides of the above equation, we obtain

$$\mathbb{E}[\|w_{t+1} - w_*\|^2] \leq (1-\mu\eta(1-\eta L))\mathbb{E}[\|w_t - w_*\|^2] + 2\eta^2 p_\epsilon \epsilon + 2\eta^2(1-p_\epsilon)M_\epsilon.$$

Hence, we conclude

$$\mathbb{E}[\|w_{t+1} - w_*\|^2] \leq (1-\mu\eta(1-\eta L))^{t+1}\|w_0 - w_*\|^2 + \frac{2\eta}{\mu(1-\eta L)}p_\epsilon\epsilon + \frac{2\eta}{\mu(1-\eta L)}(1-p_\epsilon)M_\epsilon.$$

□

The main conclusion is stated in the following corollary.

Corollary 2.2.1. *For any ϵ such that $1 - p_\epsilon \leq \epsilon$, and for Algorithm 1 with $\eta \leq \frac{1}{2L}$, we have*

$$\mathbb{E}[\|w_t - w_*\|^2] \leq (1-\mu\eta)^t\|w_0 - w_*\|^2 + \frac{2\eta}{\mu}\left(1 + M_\epsilon\right)\epsilon.$$

Furthermore if $t \geq T$ for $T = \frac{1}{\mu\eta}\log\left(\frac{\mu\|w_0 - w_*\|^2}{2\eta(1+M_\epsilon)\epsilon}\right)$, then

$$\mathbb{E}[\|w_t - w_*\|^2] \leq \frac{4\eta}{\mu}\left(1 + M_\epsilon\right)\epsilon. \tag{2.15}$$

Proof. Taking the expectation, conditioning on $\mathcal{F}_t = \sigma(w_0, w_1, \ldots, w_t)$ to (2.14), we have

$$\begin{aligned}
\mathbb{E}[\|w_{t+1} - w_*\|^2 | \mathcal{F}_t] \leq \|w_t - w_*\|^2 &- 2\eta \nabla F(w_t)^T (w_t - w_*) \\
&+ 2\eta^2 \mathbb{E}[\|\nabla f(w_t; \xi_{t,1}) - \nabla f(w_*; \xi_{t,1})\|] \\
&+ 2\eta^2 \mathbb{E}\left[\left\| \frac{1}{b} \sum_{i=1}^{b} \nabla f(w_*; \xi_{t,i}) \right\|^2 \Big| \mathcal{F}_t \right] \\
\overset{(2.12),(2.11)}{\leq} &(1 - \mu\eta)\|w_t - w_*\|^2 - 2\eta(1 - 2\eta L)[F(w_t) - F(w_*)] \\
&+ 2\eta^2 \mathbb{E}\left[\left\| \frac{1}{b} \sum_{i=1}^{b} \nabla f(w_*; \xi_{t,i}) \right\|^2 \right] \\
\overset{\eta \leq \frac{1}{2L},(2.6)}{\leq} &(1 - \mu\eta)\|w_t - w_*\|^2 + 2\eta^2 p_\epsilon \epsilon + 2\eta^2 (1 - p_\epsilon) M_\epsilon.
\end{aligned}$$

The first inequality follows since $\{\xi_{i,i}\}_{i=1}^b$ are i.i.d. random variables. Hence, we have

$$\mathbb{E}[\|w_{t+1} - w_*\|^2] \leq (1 - \mu\eta)^{t+1} \|w_0 - w_*\|^2 + \frac{2\eta}{\mu} p_\epsilon \epsilon + \frac{2\eta}{\mu}(1 - p_\epsilon) M_\epsilon.$$

Therefore,

$$\begin{aligned}
\mathbb{E}[\|w_t - w_*\|^2] &\leq (1 - \mu\eta)^t \|w_0 - w_*\|^2 + \frac{2\eta}{\mu} p_\epsilon \epsilon + \frac{2\eta}{\mu}(1 - p_\epsilon) M_\epsilon \\
&\leq (1 - \mu\eta)^t \|w_0 - w_*\|^2 + \frac{2\eta}{\mu}(1 + M_\epsilon)\epsilon,
\end{aligned}$$

where the last inequality follows since $1 - p_\epsilon \leq \epsilon$.

First, we would like to find a T such that

$$(1 - \mu\eta)^T \|w_0 - w_*\|^2 = \frac{2\eta}{\mu} (1 + M_\epsilon) \epsilon.$$

Taking log for both sides, we have

$$T \log(1 - \mu\eta) + \log\left(\|w_0 - w_*\|^2\right) = \log\left(\frac{2\eta}{\mu} (1 + M_\epsilon) \epsilon \right).$$

Hence,

$$T = -\frac{1}{\log(1-\mu\eta)} \log\left(\frac{\mu\|w_0 - w_*\|^2}{2\eta\,(1+M_\epsilon)\,\epsilon}\right) \leq \frac{1}{\mu\eta} \log\left(\frac{\mu\|w_0 - w_*\|^2}{2\eta\,(1+M_\epsilon)\,\epsilon}\right),$$

where the last inequality follows since $-1/\log(1-x) \leq 1/x$ for $0 < x \leq 1$. Hence, if $t \geq T$ for $T = \frac{1}{\mu\eta} \log\left(\frac{\mu\|w_0 - w_*\|^2}{2\eta(1+M_\epsilon)\epsilon}\right)$, then

$$\mathbb{E}[\|w_t - w_*\|^2] \leq \frac{2\eta}{\mu}\,(1+M_\epsilon)\,\epsilon + \frac{2\eta}{\mu}\,(1+M_\epsilon)\,\epsilon = \frac{4\eta}{\mu}\,(1+M_\epsilon)\,\epsilon.$$

□

Note that in Corollary 2.2.1 we assume that $\eta \leq \frac{1}{2L}$ instead of $\eta \leq \frac{1}{L}$ only to simplify the expressions. (The proof in detail is in the Appendix.) We conclude that under the assumption $1 - p_\epsilon \leq \epsilon$, Algorithm 1 has linear convergence rate in terms of any such ϵ.

The following theorem establishes convergence rate bound for Algorithm 1 when the strong convexity assumption on $F(w)$ is relaxed.

Theorem 2.2.2. *Suppose that $f(w;\xi)$ is L-smooth and convex for every realization of ξ. Consider Algorithm 1 with $\eta < \frac{1}{L}$. Then for any $\epsilon > 0$, we have*

$$\mathbb{E}[F(w_t) - F(w_*)] \leq \frac{\|w_0 - w_*\|^2}{2\eta(1-\eta L)t} + \frac{\eta}{(1-\eta L)} p_\epsilon \epsilon + \frac{\eta M_\epsilon}{(1-\eta L)}(1-p_\epsilon), \tag{2.16}$$

where w_* is any optimal solution of $F(w)$, and p_ϵ and M_ϵ are defined in (2.4) *and* (2.5)*, respectively.*

Proof. If ϕ is convex, then

$$\phi(w) - \phi(w') \geq \nabla\phi(w')^T(w - w'),\ \forall w, w' \in \mathbb{R}^d. \tag{2.17}$$

From the proof of Theorem 2.2.1, we could have

$$\mathbb{E}[\|w_{t+1} - w_*\|^2 | \mathcal{F}_t] \leq \|w_t - w_*\|^2 - 2\eta(1-\eta L)\nabla F(w_t)^T(w_t - w_*)$$

$$
\begin{aligned}
&+ 2\eta^2 \mathbb{E}\left[\left\|\frac{1}{b}\sum_{i=1}^{b}\nabla f(w_*;\xi_{t,i})\right\|^2 \Big| \mathcal{F}_t\right] \\
&\overset{(2.17),(2.24),(2.6)}{\leq} \|w_t - w_*\|^2 - 2\eta(1-\eta L)[F(w_t) - F(w_*)] + 2\eta^2 p_\epsilon \epsilon \\
&+ 2\eta^2(1-p_\epsilon)M_\epsilon.
\end{aligned}
$$

Taking the expectation for both sides of the above equation yields

$$
\mathbb{E}[\|w_{t+1} - w_*\|^2] \leq \mathbb{E}[\|w_t - w_*\|^2] - 2\eta(1-\eta L)\mathbb{E}[F(w_t) - F(w_*)] + 2\eta^2 p_\epsilon \epsilon + 2\eta^2(1-p_\epsilon)M_\epsilon.
$$

With $\eta < \frac{1}{L}$, one obtains

$$
\begin{aligned}
\mathbb{E}[F(w_t) - F(w_*)] \leq \frac{1}{2\eta(1-\eta L)}\Big(\mathbb{E}[\|w_t - w_*\|^2] - \mathbb{E}[\|w_{t+1} - w_*\|^2]\Big) \\
+ \frac{\eta}{(1-\eta L)} p_\epsilon \epsilon + \frac{\eta M_\epsilon}{(1-\eta L)}(1-p_\epsilon).
\end{aligned}
$$

By summing from $k = 0, \ldots, t$ and averaging, we have

$$
\begin{aligned}
\frac{1}{t+1}\sum_{k=0}^{t}\mathbb{E}[F(w_k) - F(w_*)] \leq \frac{1}{2\eta(1-\eta L)(t+1)}\|w_0 - w_*\|^2 \\
+ \frac{\eta}{(1-\eta L)} p_\epsilon \epsilon + \frac{\eta M_\epsilon}{(1-\eta L)}(1-p_\epsilon).
\end{aligned}
$$

Since $\mathbb{E}[F(w_k)]$ is a non-increasing function on k, the sum on the left hand side is larger than $t+1$ times its last element. Hence,

$$
\begin{aligned}
\mathbb{E}[F(w_{t+1}) - F(w_*)] &\leq \frac{1}{t+1}\sum_{k=0}^{t}\mathbb{E}[F(w_k) - F(w_*)] \\
&\leq \frac{1}{2\eta(1-\eta L)(t+1)}\|w_0 - w_*\|^2 + \frac{\eta}{(1-\eta L)} p_\epsilon \epsilon + \frac{\eta M_\epsilon}{(1-\eta L)}(1-p_\epsilon).
\end{aligned}
$$

□

Again, the convergence rate of SGD is governed by the initial solution and quantities p_ϵ and M_ϵ. Hence we have the following corollary.

Corollary 2.2.2. *If $f(w;\xi)$ is L-smooth and convex for every realization of ξ, then for any*

ϵ such that $1 - p_\epsilon \leq \epsilon$, and $\eta \leq \frac{1}{2L}$, it holds that

$$\mathbb{E}[F(w_t) - F(w_*)] \leq \frac{\|w_0 - w_*\|^2}{\eta t} + 2\eta\left(1 + M_\epsilon\right)\epsilon.$$

Hence, if $t \geq T$ for $T = \frac{\|w_0 - w_*\|^2}{(2\eta^2)(1+M_\epsilon)\epsilon}$, we have

$$\mathbb{E}[F(w_t) - F(w_*)] \leq 4\eta\left(1 + M_\epsilon\right)\epsilon. \tag{2.18}$$

Proof. By Theorem 2.2.2, with $\eta \leq \frac{1}{2L}$, we have

$$\begin{aligned}
\mathbb{E}[F(w_t) - F(w_*)] &\leq \frac{\|w_0 - w_*\|^2}{2\eta(1-\eta L)t} + \frac{\eta}{(1-\eta L)}p_\epsilon\epsilon + \frac{\eta M_\epsilon}{(1-\eta L)}(1-p_\epsilon) \\
&\leq \frac{2\|w_0 - w_*\|^2}{2\eta t} + 2\eta p_\epsilon\epsilon + 2\eta M_\epsilon(1-p_\epsilon) \\
&\leq \frac{\|w_0 - w_*\|^2}{\eta t} + 2\eta(1+M_\epsilon)\epsilon.
\end{aligned}$$

Similar to the proof of Corollary 2.2.1, we want to find a T such that

$$\frac{\|w_0 - w_*\|^2}{\eta T} = 2\eta(1+M_\epsilon)\epsilon.$$

It is easy to see that if $t \geq T$ for $T = \frac{\|w_0 - w_*\|^2}{(2\eta^2)(1+M_\epsilon)\epsilon}$, then

$$\mathbb{E}[F(w_t) - F(w_*)] \leq 2\eta(1+M_\epsilon)\epsilon + 2\eta(1+M_\epsilon)\epsilon = 4\eta(1+M_\epsilon)\epsilon.$$

□

Similarly to the strongly convex case, under the key assumption that $1 - p_\epsilon \leq \epsilon$, we show that Algorithm 1 achieves $\mathcal{O}(\epsilon)$ optimality gap, in expectation, in $\mathcal{O}(1/\epsilon)$ iterations. In Corollary 2.2.2 we again assume that $\eta \leq \frac{1}{2L}$ instead of $\eta < \frac{1}{L}$ only to simplify the expressions and to replace $\frac{1}{1-\eta L}$ term with 2 in the complexity bound.

2.2.3 Nonconvex Objectives

In this section, we establish expected complexity bound for Algorithm 1 when applied to nonconvex objective functions. This setting includes deep neural networks in which the cost function is a sum of nonconvex function components. Despite the nonconvexity of the objective, it has been observed that deep neural networks can be trained fairly quickly by SGD algorithms. It has also been observed that after reaching certain accuracy, the SGD algorithm may slow down dramatically.

For the analysis of the nonconvex case, we need to make an assumption on the rate of change in the gradients near all local solutions, or at least those to which iterates w_t generated by the algorithm may converge.

Assumption 2.2.1. *We assume that there exists a constant $N > 0$, such that for any sequence of iterates w_0, w_1, ..., w_t of any realization of Algorithm 1, there exists a stationary point w_* of $F(w)$ (possibly dependent on that sequence) such that*

$$\frac{1}{t+1}\sum_{k=0}^{t}\left(\mathbb{E}\left[\left\|\frac{1}{b}\sum_{i=1}^{b}\nabla f(w_k;\xi_{k,i})-\frac{1}{b}\sum_{i=1}^{b}\nabla f(w_*;\xi_{k,i})\right\|^2\Big|\mathcal{F}_k\right]\right)\leq N\frac{1}{t+1}\sum_{k=0}^{t}\|\nabla F(w_k)\|^2, \tag{2.19}$$

where the expectation is taken over random variables $\xi_{k,i}$ conditioned on $\mathcal{F}_t = \sigma(w_0, w_1, \ldots, w_t)$, which is the σ-algebra generated by $w_0, w_1, \ldots, w_t$. Let $\mathcal{W}_*$ denote the set of all such stationary points w_*, determined by the constant N and by realizations w_0, w_1, ..., w_t.

This assumption is made for any realization w_0, w_1, ..., w_t and states that the average squared norm of the difference between the stochastic gradient directions computed by Algorithm 1 at w_t and the same stochastic gradient computed at w_*, over any t iterations, is proportional to the average true squared gradient norm. If w_* is a stationary point for all $f(w;\xi_{k,i})$, in other words, $\nabla f(w_*;\xi_{k,i}) = 0$ for all realizations of $\xi_{k,i}$, then Assumption 2.2.1 simply states that all stochastic gradients have the same average expected rate of growth as the true gradient, as the iterates get further away from w_*. Notice that w_* may not be a stationary point for all $f(w;\xi_{k,i})$, hence Assumption 2.2.1 bounds the average expected rate of *change* of the stochastic gradients in terms of the rate of change of the true gradient. In

the next section we will demonstrate numerically that Assumption 2.2.1 holds for problems of training deep neural networks.

We also need to slightly modify Definition 2.2.2.

Definition 2.2.4. *Let* $g_* = \frac{1}{b}\sum_{i=1}^{b} \nabla f(w_*; \xi_i)$ *for some i.i.d. random variables* $\{\xi_i\}_{i=1}^{b}$. *For any given threshold* $\epsilon > 0$, *define*

$$p_\epsilon := \inf_{\mathbf{w}_* \in \mathcal{W}_*} \mathbb{P}\left\{ \|g_*\|^2 \leq \epsilon \right\}, \tag{2.20}$$

where the infimum is taken over the set $\mathcal{W}_*$ *defined in Assumption 2.2.1. Similarly, we also define*

$$M_\epsilon := \sup_{\mathbf{w}_* \in \mathcal{W}_*} \mathbb{E}\left[\|g_*\|^2 \mid \|g_*\|^2 > \epsilon \right]. \tag{2.21}$$

We know that p_ϵ and M_ϵ defined as above exist since $p_\epsilon \geq 0$ and $M_\epsilon \leq M_{max}$. This time, if we assume that $1 - p_\epsilon \leq \epsilon$ for some reasonably small ϵ, this implies that for *all* stationary points of $F(w)$ that appear in Assumption 2.2.1 a large fraction of stochastic gradients have small norm at those points. Essentially, $\mathcal{W}_*$ consists of stationary points to which different realization of SGD iterates converge.

Theorem 2.2.3. *Let Assumption 2.2.1 hold for some* $N > 0$. *Suppose that* F *is* L*-smooth. Consider Algorithm 1 with* $\eta < \frac{1}{LN}$. *Then, for any* $\epsilon > 0$, *we have*

$$\frac{1}{t+1}\sum_{k=0}^{t} \mathbb{E}[\|\nabla F(w_k)\|^2] \leq \frac{[F(w_0) - F^*]}{\eta\,(1 - L\eta N)\,(t+1)} + \frac{L\eta}{(1 - L\eta N)}\epsilon + \frac{L\eta M_\epsilon}{(1 - L\eta N)}(1 - p_\epsilon),$$

where F^* *is any lower bound of* F; *and* p_ϵ *and* M_ϵ *are defined in* (2.20) *and* (2.21) *respectively.*

Proof. Let us assume that, there exists a local minima w_* of $F(w)$. We have

$$\begin{aligned}
\mathbb{E}[F(w_{t+1})|\mathcal{F}_t] &= \mathbb{E}[F(w_t - \eta g_t)|\mathcal{F}_t] \\
&\overset{(2.7)}{\leq} F(w_t) - \eta\|\nabla F(w_t)\|^2 + \frac{L\eta^2}{2}\mathbb{E}\left[\left\| \frac{1}{b}\sum_{i=1}^{b} \nabla f(w_t; \xi_{t,i}) \right\|^2 \Big| \mathcal{F}_t \right]
\end{aligned}$$

$$
\begin{aligned}
&\leq F(w_t) - \eta\|\nabla F(w_t)\|^2 + L\eta^2\mathbb{E}\left[\left\|\frac{1}{b}\sum_{i=1}^{b}(\nabla f(w_t;\xi_{t,i}) - \nabla f(w_*;\xi_{t,i}))\right\|^2 \Big| \mathcal{F}_t\right] \\
&\quad + L\eta^2\mathbb{E}\left[\left\|\frac{1}{b}\sum_{i=1}^{b}\nabla f(w_*;\xi_{t,i})\right\|^2 \Big| \mathcal{F}_t\right] \\
&\leq F(w_t) - \eta\|\nabla F(w_t)\|^2 + L\eta^2\mathbb{E}\left[\left\|\frac{1}{b}\sum_{i=1}^{b}(\nabla f(w_t;\xi_{t,i}) - \nabla f(w_*;\xi_{t,i}))\right\|^2 \Big| \mathcal{F}_t\right] \\
&\quad + L\eta^2\epsilon + L\eta^2(1-p_\epsilon)M_\epsilon.
\end{aligned}
$$

By summing from $k = 0, \ldots, t$ and averaging, we have

$$
\begin{aligned}
\frac{1}{t+1}\sum_{k=0}^{t}\mathbb{E}[F(w_{k+1})|\mathcal{F}_k] &\leq \frac{1}{t+1}\sum_{k=0}^{t}F(w_k) - \eta\frac{1}{t+1}\sum_{k=0}^{t}\|\nabla F(w_k)\|^2 \\
&\quad + L\eta^2\frac{1}{t+1}\sum_{k=0}^{t}\left(\mathbb{E}\left[\left\|\frac{1}{b}\sum_{i=1}^{b}(\nabla f(w_k;\xi_{k,i}) - \nabla f(w_*;\xi_{k,i}))\right\|^2 \Big| \mathcal{F}_k\right]\right) \\
&\quad + L\eta^2\epsilon + L\eta^2(1-p_\epsilon)M_\epsilon \\
&\overset{(2.19)}{\leq} \frac{1}{t+1}\sum_{k=0}^{t}F(w_k) - \eta\left(1 - L\eta N\right)\frac{1}{t+1}\sum_{k=0}^{t}\|\nabla F(w_k)\|^2 \\
&\quad + L\eta^2\epsilon + L\eta^2(1-p_\epsilon)M_\epsilon.
\end{aligned}
$$

Taking the expectation for the above equation, we have

$$
\begin{aligned}
\frac{1}{t+1}\sum_{k=0}^{t}\mathbb{E}[F(w_{k+1})] &\leq \frac{1}{t+1}\sum_{k=0}^{t}\mathbb{E}[F(w_k)] - \eta\left(1 - L\eta N\right)\frac{1}{t+1}\sum_{k=0}^{t}\mathbb{E}[\|\nabla F(w_k)\|^2] \\
&\quad + L\eta^2\epsilon + L\eta^2(1-p_\epsilon)M_\epsilon.
\end{aligned}
$$

Hence, with $\eta < \frac{1}{LN}$, we have

$$
\begin{aligned}
\frac{1}{t+1}\sum_{k=0}^{t}\mathbb{E}[\|\nabla F(w_k)\|^2] &\leq \frac{[\mathbb{E}[F(w_0)] - \mathbb{E}[F(w_{t+1})]]}{\eta\left(1 - L\eta N\right)(t+1)} + \frac{L\eta}{(1 - L\eta N)}\epsilon + \frac{L\eta M_\epsilon}{(1 - L\eta N)}(1-p_\epsilon) \\
&\leq \frac{[F(w_0) - F^*]}{\eta\left(1 - L\eta N\right)(t+1)} + \frac{L\eta}{(1 - L\eta N)}\epsilon + \frac{L\eta M_\epsilon}{(1 - L\eta N)}(1-p_\epsilon),
\end{aligned}
$$

where F^* is any lower bound of F. □

Corollary 2.2.3. *Let Assumption 2.2.1 hold and p_ϵ and M_ϵ be defined as in* (2.20) *and* (2.21)*. For any ϵ such that $1 - p_\epsilon \leq \epsilon$, and for $\eta \leq \frac{1}{2LN}$, we have*

$$\frac{1}{t+1}\sum_{k=0}^{t}\mathbb{E}[\|\nabla F(w_k)\|^2] \leq \frac{2[F(w_0) - F^*]}{\eta(t+1)} + 2L\eta(1 + M_\epsilon)\epsilon.$$

Hence, if $t \geq T$ for $T = \frac{[F(w_0)-F^*]}{(L\eta^2)(1+M_\epsilon)\epsilon}$, we have

$$\frac{1}{t+1}\sum_{k=0}^{t}\mathbb{E}[\|\nabla F(w_k)\|^2] \leq 4L\eta(1 + M_\epsilon)\epsilon.$$

Proof. By Theorem 2.2.3, with $\eta \leq \frac{1}{2LN}$, we have

$$\begin{aligned}\frac{1}{t+1}\sum_{k=0}^{t}\mathbb{E}[\|\nabla F(w_k)\|^2] &\leq \frac{[F(w_0) - F^*]}{\eta\,(1 - L\eta N)\,(t+1)} + \frac{L\eta}{(1 - L\eta N)}\epsilon + \frac{L\eta M_\epsilon}{(1 - L\eta N)}(1 - p_\epsilon) \\ &\leq \frac{2[F(w_0) - F^*]}{\eta(t+1)} + 2L\eta\epsilon + 2L\eta M_\epsilon(1 - p_\epsilon) \\ &\leq \frac{2[F(w_0) - F^*]}{\eta(t+1)} + 2L\eta(1 + M_\epsilon)\epsilon.\end{aligned}$$

Similar to the proof of Corollaries 2.2.1 and 2.2.2, we want to find a T such that

$$\frac{2[F(w_0) - F^*]}{\eta T} = 2L\eta(1 + M_\epsilon)\epsilon.$$

It is easy to see that if $t \geq T$ for $T = \frac{[F(w_0)-F^*]}{(L\eta^2)(1+M_\epsilon)\epsilon}$, then

$$\frac{1}{t+1}\sum_{k=0}^{t}\mathbb{E}[\|\nabla F(w_k)\|^2] \leq 2L\eta(1 + M_\epsilon)\epsilon + 2L\eta(1 + M_\epsilon)\epsilon = 4L\eta(1 + M_\epsilon)\epsilon.$$

□

2.3 Numerical Experiments

The purpose of this section is to numerically validate our assumptions on p_ϵ as defined in Definition 2.2.2. We wish to show that there exists a small ϵ satisfying

$$1 - p_\epsilon \approx \epsilon. \tag{2.22}$$

For our numerical experiments, we consider the finite sum minimization problem

$$\min_{w \in \mathbb{R}^d} \left\{ F(w) = \frac{1}{n} \sum_{i=1}^{n} f_i(w) \right\}. \tag{2.23}$$

Definition 2.3.1. *Let w_* be a stationary point of the objective function $F(w)$. For any given threshold $\epsilon > 0$, define the set $\mathcal{S}_\epsilon$ and its complement $\mathcal{B}_\epsilon$*

$$\mathcal{S}_\epsilon := \left\{ i : \|\nabla f_i(w_*)\|^2 \leq \epsilon \right\} \qquad \text{and} \qquad \mathcal{B}_\epsilon := [n] \backslash \mathcal{S}_\epsilon.$$

We also define the quantity $p_\epsilon := \frac{|\mathcal{S}_\epsilon|}{n}$ that measures the size of the set $\mathcal{S}_\epsilon$ and the upper bound M_ϵ

$$\frac{1}{|\mathcal{B}_\epsilon|} \sum_{i \in \mathcal{B}_\epsilon} \|\nabla f_i(w_*)\|^2 \leq M_\epsilon.$$

2.3.1 Logistic Regression for Convex Case

We consider ℓ_2-regularized logistic regression problems with

$$f_i(w) = \log(1 + \exp(-y_i x_i^T w)) + \frac{\lambda}{2}\|w\|^2,$$

where the penalty parameter λ is set to $1/n$, a widely-used value in the literature [51]. We conducted experiments on popular datasets `covtype, ijcnn1, w8a, a9a, mushrooms, phishing, skin_nonskin` from the LIBSVM website [2] and `ijcnn2` [3]. The optimal solution w_* of the convex problem (2.23) is found by using the full-batch `L-BFGS` method [40] with the stopping criterion $\|\nabla F(w_*)\|_2 \leq 10^{-12}$. We then ran Algorithm 1 using the learning

[2] http://www.csie.ntu.edu.tw/∼cjlin/libsvmtools/datasets/
[3] http://mlbench.org/repository/data/viewslug/ijcnn1/

rate $\eta = 10^{-1}$ and the batch-size $b = 1$ and 100 epochs. The final solution given by the SGD algorithm is denoted by w_{SGD}. We report the value of p_ϵ defined in Definition 2.3.1 expressed in percentage form for different values of ϵ.

As we can see from Table 2.1 that $\epsilon = 10^{-3}$ satisfies (2.22) for all cases. For datasets `covtype, ijcnn1, ijcnn2,phishing` and `skin_nonskin`, ϵ can take a smaller value 10^{-4}. The small value for ϵ indicates that SGD with a fixed step size can converge to a small neighborhood of the optimal solution of F. The success of using SGD is illustrated, optimality gaps $F(w_{SGD}) - F(w_*)$ are small in our experiments.

Table 2.1: Percentage of f_i with small gradient value for different threshold ϵ (Logistic Regression) (Opt. $= F(w_{SGD}) - F(w_*)$)

Datasets	Opt.	$\epsilon = 10^{-2}$	$\epsilon = 10^{-3}$	$\epsilon = 10^{-4}$	$\epsilon = 10^{-5}$	$\epsilon = 10^{-6}$	Train accuracy
covtype	$5 \cdot 10^{-4}$	100%	100%	100%	99.9995%	54.9340%	0.7562
ijcnn1	$1 \cdot 10^{-4}$	100%	100%	100%	96.8201%	89.0197%	0.9219
ijcnn2	$2 \cdot 10^{-4}$	100%	100%	100%	99.2874%	90.4565%	0.9228
w8a	$8 \cdot 10^{-5}$	100%	99.9899%	99.4231%	98.3557%	92.7818%	0.9839
a9a	$4 \cdot 10^{-3}$	100%	100%	84.0945%	58.5824%	40.0909%	0.8491
mushrooms	$3 \cdot 10^{-5}$	100%	100%	99.9261%	98.7568%	94.4239%	1.0000
phishing	$2 \cdot 10^{-4}$	100%	100%	100%	89.9231%	73.8128%	0.9389
skin_nonskin	$4 \cdot 10^{-5}$	100%	100%	100%	99.6331%	91.3730%	0.9076

We compare convergence rates of SGD (learning rate $\eta = 0.1 < \frac{1}{2L}$) with `SVRG` [28] and `L-BFGS` [40] as shown in Figure 2.2. We can observe that SGD has better performance than SVRG and L-BFGS in the beginning until it achieves $\mathcal{O}(\epsilon)$ accuracy, for the value of ϵ consistent to what is indicated in Table 2.1. We note that the values of M_ϵ for all datasets should not exceed 10^{-2} according to Table 2.1.

2.3.2 Neural Networks for Nonconvex Case

For experiments with nonconvex problems we train DNNs using two standard network architectures: feed forward network (FFN) and convolutional neural network (CNN). Configuration of FNN includes 2 dense layers each containing 256 neurons followed by a ReLU activation. The output layer consists of c neurons with the softmax activation where c is the number of classes. For CNN, we configure the network to have 2 convolutional layers followed by 2 dense layers. Convolutional layers contain a convolutional operator followed

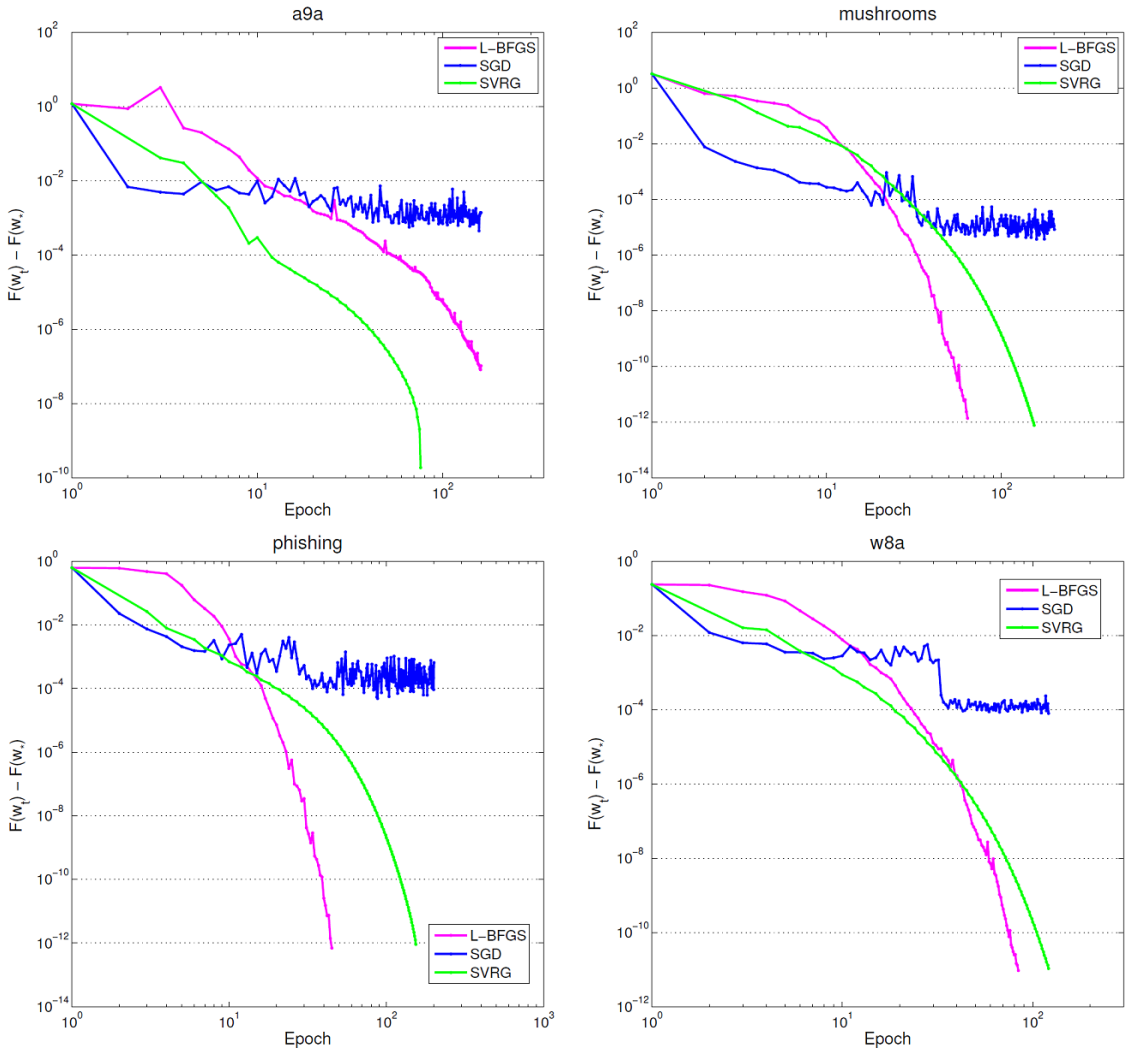


Figure 2.2: The convergence comparisons of SGD, SVRG, and L-BFGS

by a ReLU activation and then a max pooling. The number of filters of both the convolutional operators are set to 64 and the associated filter sizes are 5×5. Number of neurons in dense layers are 384 and 192, respectively, and the activation used in these layers is again ReLU. Throughout the simulations, we use popular datasets which include `MNIST` [4] (60000 training data images of size 28×28 contained in 10 classes), `SVHN` [5] (73257 training images of size 32×32 contained in 10 classes), `CIFAR10` (50000 training color images of size 32×32 contained in 10 classes), and `CIFAR100` [6] (50000 training color images of size 32×32 contained in 100 classes).

[4] http://yann.lecun.com/exdb/mnist/
[5] http://ufldl.stanford.edu/housenumbers/
[6] https://www.cs.toronto.edu/ kriz/cifar.html

We trained the networks by the popular Adam algorithm with a minibatch of size 32 and reported the values of p_ϵ at the last iteration w_{Adam}. In all our experiments, we did not apply batch normalization and dropout techniques during the training. Since the problem of interest is nonconvex, multiple local minima could exist. We experimented with 10 seeds and reported the minimum result (minimum of the percentage of component functions with small gradient value). Table 2.2 shows the values of p_ϵ in terms of percentage for different thresholds ϵ. As is clear from the table, p_ϵ is close to 1 for a sufficiently small ϵ. It confirms that the majority of component functions f_i has negligible gradients at the final solution of F.

Table 2.2: Percentage of f_i with small gradient value for different threshold ϵ (Neural Networks) (Opt. $= \|\nabla F(w_{Adam})\|^2$)

Datasets	Opt.	$\epsilon = 10^{-3}$	$\epsilon = 10^{-5}$	$\epsilon = 10^{-7}$	Train accuracy	N	M
MNIST (FFN)	$1.3 \cdot 10^{-15}$	100%	100%	99.99%	1.0000	6500	$2.1 \cdot 10^{-8}$
SVHN (FFN)	$3.5 \cdot 10^{-3}$	99.94%	99.92%	99.91%	0.9997	12000	500
MNIST (CNN)	$1.6 \cdot 10^{-17}$	100%	100%	100%	1.0000	6083	$6.4 \cdot 10^{-8}$
SVHN (CNN)	$8.1 \cdot 10^{-7}$	99.99%	99.98%	99.96%	0.9999	8068	0.18
CIFAR10 (CNN)	$5.1 \cdot 10^{-20}$	100%	100%	100%	1.0000	1205	$8.7 \cdot 10^{-14}$
CIFAR100 (CNN)	$5.5 \cdot 10^{-2}$	99.50%	99.45%	99.42%	0.9988	984	3000

The value of N is the estimation of N in (2.19), which is shown in Section 2.3.3. We note that for some datasets and network structures, Adam did not converge to a real local solution (SVHN-FFN and CIFAR100-CNN) and Table 2.2 shows only an approximation of the behavior at the local solution.

2.3.3 Nonconvex Assumption Verification

This section shows how to estimate N. We are proving some numerical experiments to verify Assumption 2.2.1. Let us define

$$r_t = \frac{\frac{1}{t+1}\sum_{k=0}^{t}\left(\frac{1}{n}\sum_{i=1}^{n}\|\nabla f_i(w_k) - \nabla f_i(w_*)\|^2\right)}{\frac{1}{t+1}\sum_{k=0}^{t}\|F(w_k)\|^2}$$

We show two plots to see behaviors of r_t for MNIST (FFN) and CIFAR10 (CNN) (others are reported in Table 2.2. We can observe from Figure 2.3 that r_t is bounded above by a constant. (Note that $r_t \leq N$.)

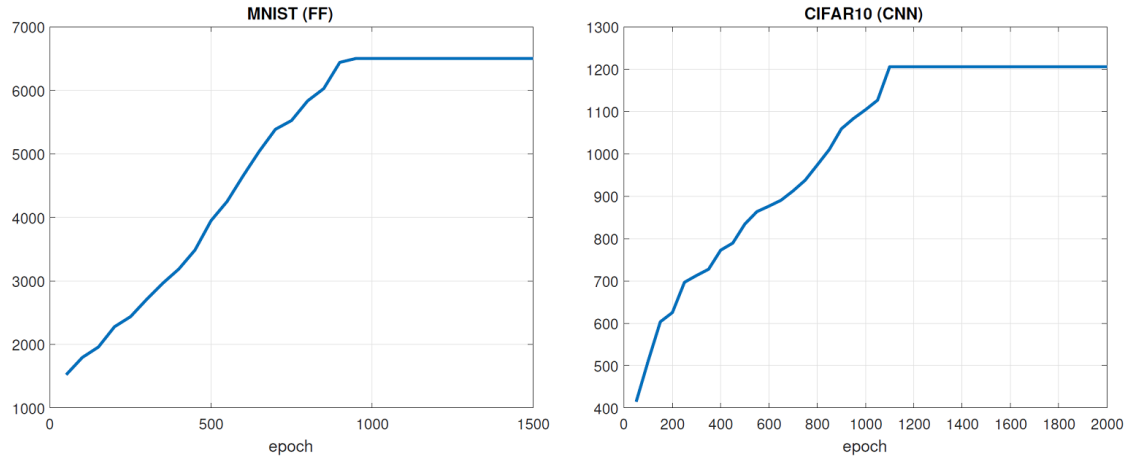


Figure 2.3: The behaviors of r_t

2.4 Conclusions

We have demonstrated that based on the behavior of the stochastic gradient estimates at or near the stationary points, SGD with fixed step size converges with the same rate as full gradient descent of the variance reduction methods, until it reaches the accuracy where the variance in the stochastic gradient estimates starts to dominate and prevents further convergence. In particular out assumption is that $1-\epsilon$ fraction of the stochastic gradient estimates have squared norm below ϵ at the solution. Note ϵ can be made arbitrarily small by increasing the minibatch size b. Indeed we have the following lemma

Lemma 2.4.1. *Let $\xi_1, \ldots, \xi_b$ be i.i.d. with $\mathbb{E}[\nabla f(w;\xi_i)] = \nabla F(w)$, $i = 1, \ldots, b$, for all $w \in \mathbb{R}^d$. Then,*

$$\mathbb{E}\left[\left\|\frac{1}{b}\sum_{i=1}^{b}\nabla f(w;\xi_i) - \nabla F(w)\right\|^2\right] = \frac{\mathbb{E}[\|\nabla f(w;\xi_1)\|^2] - \|\nabla F(w)\|^2}{b}. \tag{2.24}$$

Proof. We are going to use mathematical induction to prove the result. With $b = 1$, it is easy to see

$$\begin{aligned}\mathbb{E}\left[\|\nabla f(w;\xi_1) - \nabla F(w)\|^2\right] &= \mathbb{E}[\|\nabla f(w;\xi_1)\|^2] - 2\|\nabla F(w)\|^2 + \|\nabla F(w)\|^2 \\ &= \mathbb{E}[\|\nabla f(w;\xi_1)\|^2] - \|\nabla F(w)\|^2.\end{aligned}$$

Let assume that it is true with $b = m - 1$, we are going to show it is also true with

$b = m$. We have

$$
\begin{aligned}
&\mathbb{E}\left[\left\|\frac{1}{m}\sum_{i=1}^{m}\nabla f(w;\xi_i) - \nabla F(w)\right\|^2\right] \\
&= \mathbb{E}\left[\left\|\frac{\sum_{i=1}^{m-1}\nabla f(w;\xi_i) - (m-1)\nabla F(w) + (\nabla f(w;\xi_m) - \nabla F(w))}{m}\right\|^2\right] \\
&= \frac{1}{m^2}\left(\mathbb{E}\left[\left\|\sum_{i=1}^{m-1}\nabla f(w;\xi_i) - (m-1)\nabla F(w)\right\|^2\right] + \mathbb{E}\left[\|\nabla f(w;\xi_m) - \nabla F(w)\|^2\right]\right) \\
&\quad + \frac{1}{m}\mathbb{E}\left[2\left(\sum_{i=1}^{m-1}\nabla f(w;\xi_i) - (m-1)\nabla F(w)\right)^T(\nabla f(w;\xi_m) - \nabla F(w))\right] \\
&= \frac{1}{m^2}\left(\mathbb{E}\left[\left\|\sum_{i=1}^{m-1}\nabla f(w;\xi_i) - (m-1)\nabla F(w)\right\|^2\right] + \mathbb{E}\left[\|\nabla f(w;\xi_m) - \nabla F(w)\|^2\right]\right) \\
&= \frac{1}{m^2}\left((m-1)\mathbb{E}[\|\nabla f(w;\xi_1)\|^2] - (m-1)\|\nabla F(w)\|^2 + \mathbb{E}[\|\nabla f(w;\xi_m)\|^2] - \|\nabla F(w)\|^2\right) \\
&= \frac{1}{m}\left(\mathbb{E}[\|\nabla f(w;\xi_1)\|^2] - \|\nabla F(w)\|^2\right).
\end{aligned}
$$

The third and the last equalities follow since $\xi_1, \ldots, \xi_b$ be i.i.d. with $\mathbb{E}[\nabla f(w;\xi_i)] = \nabla F(w)$. Therefore, the desired result is achieved. □

It is easy to see that by choosing large b the relation $1 - p_\epsilon \leq \epsilon$ can be achieved for smaller values of ϵ. In the limit for arbitrarily small ϵ we recover full gradient method and its convergence behavior.

Chapter 3

SGD and Hogwild!

Stochastic gradient descent (SGD) is the optimization algorithm of choice in many machine learning applications such as regularized empirical risk minimization and training deep neural networks. The classical convergence analysis of SGD is carried out under the assumption that the norm of the stochastic gradient is uniformly bounded. While this might hold for some loss functions, it is always violated for cases where the objective function is strongly convex. In [13], a new analysis of convergence of SGD is performed under the assumption that stochastic gradients are bounded with respect to the true gradient norm. Here we show that for stochastic problems arising in machine learning such a bound always holds; and we also propose an alternative convergence analysis of SGD within a diminishing learning rate regime, which results in more relaxed conditions than those in [13]. We then move on the asynchronous parallel setting, and prove convergence of the Hogwild! algorithm in the same regime, obtaining the first convergence results for this method in the case of a diminished learning rate.

3.1 Introduction

We are interested in solving the following stochastic optimization problem

$$\min_{w \in \mathbb{R}^d} \left\{ F(w) = \mathbb{E}[f(w;\xi)] \right\}, \tag{3.1}$$

where ξ is a random variable obeying some distribution.

In the case of empirical risk minimization with a training set $\{(x_i, y_i)\}_{i=1}^n$, ξ_i is a realization of a random variable that is defined by the i-th element of the training set. Then, by defining $f_i(w) := f(w; \xi_i)$, empirical risk minimization reduces to

$$\min_{w \in \mathbb{R}^d} \left\{ F(w) = \frac{1}{n} \sum_{i=1}^{n} f_i(w) \right\}. \tag{3.2}$$

Problem (3.2) arises frequently in supervised learning applications [25]. For a wide range of applications, such as linear regression and logistic regression, the objective function F is strongly convex and each f_i, $i \in [n]$, is convex and has Lipschitz continuous gradients (with Lipschitz constant L). Given a training set $\{(x_i, y_i)\}_{i=1}^n$ with $x_i \in \mathbb{R}^d, y_i \in \mathbb{R}$, the ℓ_2-regularized least squares regression model, for example, is written as (3.2) with $f_i(w) \overset{\text{def}}{=} (\langle x_i, w \rangle - y_i)^2 + \frac{\lambda}{2}\|w\|^2$. The ℓ_2-regularized logistic regression for binary classification is written with $f_i(w) \overset{\text{def}}{=} \log(1 + \exp(-y_i \langle x_i, w \rangle)) + \frac{\lambda}{2}\|w\|^2$, $y_i \in \{-1, 1\}$. It is well established by now that solving this type of problem by gradient descent (GD) [49, 57] may be prohibitively expensive and stochastic gradient descent (SGD) is thus preferable. Recently, a class of variance reduction methods [34, 16, 28, 51] has been proposed in order to reduce the computational cost. All these methods explicitly exploit the finite sum form of (3.2) and thus they have some disadvantages for very large scale machine learning problems and are not applicable to (3.1).

To apply SGD to the general form (3.1) one needs to assume existence of unbiased gradient estimators. This is usually defined as follows:

$$\mathbb{E}_\xi[\nabla f(w; \xi)] = \nabla F(w),$$

for any fixed w. Here we make an important observation: if we view (3.1) not as a general stochastic problem but as the expected risk minimization problem, where ξ corresponds to a random data sample pulled from a distribution, then (3.1) has an additional key property: for each realization of the random variable ξ, $f(w; \xi)$ is a convex function with Lipschitz continuous gradients. Notice that traditional analysis of SGD for general stochastic problem

of the form (3.1) does not make any assumptions on individual function realizations. In this chapter, we derive convergence properties for SGD applied to (3.1) with these additional assumptions on $f(w;\xi)$ and also extend to the case when $f(w;\xi)$ are not necessarily convex.

Regardless of the properties of $f(w;\xi)$ we assume that F in (3.1) is strongly convex. We define the (unique) optimal solution of F as w_*.

Assumption 3.1.1 (μ-strongly convex)**.** *The objective function $F : \mathbb{R}^d \to \mathbb{R}$ is a μ-strongly convex, i.e., there exists a constant $\mu > 0$ such that $\forall w, w' \in \mathbb{R}^d$,*

$$F(w) - F(w') \geq \langle \nabla F(w'), (w - w') \rangle + \frac{\mu}{2}\|w - w'\|^2. \tag{3.3}$$

It is well-known in literature [49, 13] that Assumption 3.1.1 implies

$$2\mu[F(w) - F(w_*)] \leq \|\nabla F(w)\|^2 \ , \ \forall w \in \mathbb{R}^d. \tag{3.4}$$

The classical theoretical analysis of SGD assumes that the *stochastic gradients are uniformly bounded*, i.e. there exists a finite (fixed) constant $\sigma < \infty$, such that

$$\mathbb{E}[\|\nabla f(w;\xi)\|^2] \leq \sigma^2 \ , \ \forall w \in \mathbb{R}^d \tag{3.5}$$

(see e.g. [69, 48, 64, 26, 63], etc.). However, this assumption is clearly false if F is strongly convex. Specifically, under this assumption together with strong convexity, $\forall w \in \mathbb{R}^d$, we have

$$\begin{aligned} 2\mu[F(w) - F(w_*)] &\overset{(3.4)}{\leq} \|\nabla F(w)\|^2 = \|\mathbb{E}[\nabla f(w;\xi)]\|^2 \\ &\leq \mathbb{E}[\|\nabla f(w;\xi)\|^2] \overset{(3.5)}{\leq} \sigma^2. \end{aligned}$$

Hence,

$$F(w) \leq \frac{\sigma^2}{2\mu} + F(w_*) \ , \ \forall w \in \mathbb{R}^d.$$

On the other hand strong convexity and $\nabla F(w_*) = 0$ imply

$$F(w) \geq \mu \|w - w_*\|^2 + F(w_*) \ , \ \forall w \in \mathbb{R}^d.$$

The last two inequalities are clearly in contradiction with each other for sufficiently large $\|w - w_*\|^2$.

Let us consider the following example: $f_1(w) = \frac{1}{2}w^2$ and $f_2(w) = w$ with $F(w) = \frac{1}{2}(f_1(w) + f_2(w))$. Note that F is strongly convex, while individual realizations are not necessarily so. Let $w_0 = 0$, for any number $t \geq 0$, with probability $\frac{1}{2^t}$ the steps of SGD algorithm for all $i < t$ are $w_{i+1} = w_i - \eta_i$. This implies that $w_t = -\sum_{i=1}^{t} \eta_i$ and since $\sum_{i=1}^{\infty} \eta_i = \infty$ then $|w_t|$ can be arbitrarily large for large enough t with probability $\frac{1}{2^t}$. Noting that for this example, $\mathbb{E}[\|\nabla f(w_t; \xi)\|^2] = \frac{1}{2}w_t^2 + \frac{1}{2}$, we see that $\mathbb{E}[\|\nabla f(w_t; \xi)\|^2]$ can also be arbitrarily large.

Recently, in the review paper [13], convergence of SGD for general stochastic optimization problem was analyzed under the following assumption: there exist constants M and N such that $\mathbb{E}[\|\nabla f(w_t; \xi_t)\|^2] \leq M\|\nabla F(w_t)\|^2 + N$, where w_t, $t \geq 0$, are generated by the algorithm. This assumption does not contradict strong convexity, however, in general, constants M and N are unknown, while M is used to determine the learning rate η_t [13]. In addition, the rate of convergence of the SGD algorithm depends on M and N. In this chapter, we show that under the smoothness assumption on individual realizations $f(w, \xi)$ it is possible to derive the bound $\mathbb{E}[\|\nabla f(w; \xi)\|^2] \leq M_0[F(w) - F(w_*)] + N$ with specific values of M_0, and N for $\forall w \in \mathbb{R}^d$, which in turn implies the bound $\mathbb{E}[\|\nabla f(w; \xi)\|^2] \leq M\|\nabla F(w)\|^2 + N$ with specific M, by strong convexity of F. We also note that, in [47], the convergence of SGD without bounded gradient assumption is studied. We then provide an alternative convergence analysis for SGD which shows convergence in expectation with a bound on learning rate which is larger than that in [13, 47] by a factor of L/μ. We then use the new framework for the convergence analysis of SGD to analyze an asynchronous stochastic gradient method.

In [64], an asynchronous stochastic optimization method called Hogwild! was proposed. Hogwild! algorithm is a parallel version of SGD, where each processor applies SGD steps

independently of the other processors to the solution w which is shared by all processors. Thus, each processor computes a stochastic gradient and updates w without "locking" the memory containing w, meaning that multiple processors are able to update w at the same time. This approach leads to much better scaling of parallel SGD algorithm than a synchoronous version, but the analysis of this method is more complex. In [64, 44, 15] various variants of Hogwild! with a fixed step size are analyzed under the assumption that the gradients are bounded as in (3.5). In this chapter, we extend our analysis of SGD to provide analysis of Hogwild! with diminishing step sizes and without the assumption on bounded gradients.

In a recent technical report [35] Hogwild! with fixed step size is analyzed without the bounded gradient assumption. We note that SGD with fixed step size only converges to a neighborhood of the optimal solution, while by analyzing the diminishing step size variant we are able to show convergence to the *optimal solution* with probability one. Both in [35] and in this chapter, the version of Hogwild! with inconsistent reads and writes is considered.

3.1.1 Contribution

We provide a new framework for the analysis of stochastic gradient algorithms in the strongly convex case under the condition of Lipschitz continuity of the individual function realizations, but **without requiring any bounds on the stochastic gradients**. Within this framework we have the following contributions:

- We prove the almost sure (w.p.1) convergence of SGD with diminishing step size. Our analysis provides a larger bound on the possible initial step size when compared to any previous analysis of convergence in expectation for SGD.

- We introduce a general recurrence for vector updates which has as its special cases (a) Hogwild! algorithm with diminishing step sizes, where each update involves all non-zero entries of the computed gradient, and (b) a position-based updating algorithm where each update corresponds to only one uniformly selected non-zero entry of the computed gradient.

- We analyze this general recurrence under inconsistent vector reads from and vector

writes to shared memory (where individual vector entry reads and writes are atomic in that they cannot be interrupted by writes to the same entry) assuming that there exists a delay τ such that during the $(t+1)$-th iteration a gradient of a read vector w is computed which includes the aggregate of all the updates up to and including those made during the $(t-\tau)$-th iteration. In other words, τ controls to what extend past updates influence the shared memory.

  - Our upper bound for the expected convergence rate is sublinear, i.e., $O(1/t)$, and its precise expression allows comparison of algorithms (a) and (b) described above.
  - For SGD we can improve this upper bound by a factor 2 and also show that its initial step size can be larger.
  - We show that τ can be a function of t as large as $\approx \sqrt{t/\ln t}$ without affecting the asymptotic behavior of the upper bound; we also determine a constant T_0 with the property that, for $t \geq T_0$, higher order terms containing parameter τ are smaller than the leading $O(1/t)$ term. We give intuition explaining why the expected convergence rate is not more affected by τ. Our experiments confirm our analysis.
  - We determine a constant T_1 with the property that, for $t \geq T_1$, the higher order term containing parameter $\|w_0 - w_*\|^2$ is smaller than the leading $O(1/t)$ term.

- All the above contributions generalize to the non-convex setting where we do not need to assume that the component functions $f(w;\xi)$ are convex in w.

3.1.2 Organization

We analyse the convergence rate of SGD in Section 3.2 and introduce the general recursion and its analysis in Section 3.3. Experiments are reported in Section 3.5.

3.2 New Framework for Convergence Analysis of SGD

We introduce SGD algorithm in Algorithm 2.

Algorithm 2 Stochastic Gradient Descent (SGD) Method

Initialize: w_0
Iterate:
for $t = 0, 1, 2, \ldots$ **do**
  Choose a step size (i.e., learning rate) $\eta_t > 0$.
  Generate a realization of the random variable ξ_t.
  Compute a stochastic gradient $\nabla f(w_t; \xi_t)$.
  Update the new iterate $w_{t+1} = w_t - \eta_t \nabla f(w_t; \xi_t)$.
end for

The sequence of random variables $\{\xi_t\}_{t \geq 0}$ is assumed to be i.i.d.[1] Let us introduce our key assumption that each realization $\nabla f(w; \xi)$ is an L-smooth function.

Assumption 3.2.1 (L-smooth)**.** *$f(w; \xi)$ is L-smooth for every realization of ξ, i.e., there exists a constant $L > 0$ such that, $\forall w, w' \in \mathbb{R}^d$,*

$$\|\nabla f(w; \xi) - \nabla f(w'; \xi)\| \leq L \|w - w'\|. \tag{3.6}$$

Assumption 3.2.1 implies that F is also L-smooth. Then, by the property of L-smooth function (in [49]), we have, $\forall w, w' \in \mathbb{R}^d$,

$$F(w) \leq F(w') + \langle \nabla F(w'), (w - w') \rangle + \frac{L}{2} \|w - w'\|^2. \tag{3.7}$$

The following additional convexity assumption can be made, as it holds for many problems arising in machine learning.

Assumption 3.2.2. *$f(w; \xi)$ is convex for every realization of ξ, i.e., $\forall w, w' \in \mathbb{R}^d$,*

$$f(w; \xi) - f(w'; \xi) \geq \langle \nabla f(w'; \xi), (w - w') \rangle.$$

We first derive our analysis under Assumptions 3.2.1, and 3.2.2 and then we derive weaker results under only Assumption 3.2.1. First, we introduce some useful lemmas as follows.

Lemma 3.2.1 (Generalization of the result in [28])**.** *Let Assumptions 3.2.1 and 3.2.2 hold.*

[1] Independent and identically distributed.

Then, $\forall w \in \mathbb{R}^d$,

$$\mathbb{E}[\|\nabla f(w;\xi) - \nabla f(w_*;\xi)\|^2] \leq 2L[F(w) - F(w_*)], \tag{3.8}$$

where ξ *is a random variable, and* $w_* = \arg\min_w F(w)$.

Lemma 3.2.2 ([10])**.** *Let* Y_k, Z_k, *and* W_k, $k = 0, 1, \ldots$, *be three sequences of random variables and let* $\{\mathcal{F}_k\}_{k\geq 0}$ *be a filtration, that is,* σ*-algebras such that* $\mathcal{F}_k \subset \mathcal{F}_{k+1}$ *for all* k. *Suppose that:*

- *The random variables* Y_k, Z_k, *and* W_k *are nonnegative, and* $\mathcal{F}_k$*-measurable.*
- *For each* k, *we have* $\mathbb{E}[Y_{k+1}|\mathcal{F}_k] \leq Y_k - Z_k + W_k$.
- *There holds, w.p.1,*

$$\sum_{k=0}^{\infty} W_k < \infty.$$

Then, we have, w.p.1,

$$\sum_{k=0}^{\infty} Z_k < \infty \text{ and } Y_k \to Y \geq 0.$$

3.2.1 Convergence With Probability One

As discussed in the introduction, under Assumptions 3.2.1 and 3.2.2 we can now derive a bound on $\mathbb{E}\|\nabla f(w;\xi)\|^2$.

Lemma 3.2.3. *Let Assumptions 3.2.1 and 3.2.2 hold. Then, for* $\forall w \in \mathbb{R}^d$,

$$\mathbb{E}[\|\nabla f(w;\xi)\|^2] \leq 4L[F(w) - F(w_*)] + N, \tag{3.9}$$

where $N = 2\mathbb{E}[\|\nabla f(w_*;\xi)\|^2]$; ξ *is a random variable, and* $w_* = \arg\min_w F(w)$.

Proof. Note that

$$\|a\|^2 = \|a - b + b\|^2 \leq 2\|a - b\|^2 + 2\|b\|^2, \tag{3.10}$$

$$\Rightarrow \frac{1}{2}\|a\|^2 - \|b\|^2 \leq \|a-b\|^2. \tag{3.11}$$

Hence,

$$\begin{aligned}\frac{1}{2}\mathbb{E}[\|\nabla f(w;\xi)\|^2] - \mathbb{E}[\|\nabla f(w_*;\xi)\|^2] &= \mathbb{E}\left[\frac{1}{2}\|\nabla f(w;\xi)\|^2 - \|\nabla f(w_*;\xi)\|^2\right] \\ &\overset{(3.11)}{\leq} \mathbb{E}[\|\nabla f(w;\xi) - \nabla f(w_*;\xi)\|^2] \\ &\overset{(6.21)}{\leq} 2L[F(w) - F(w_*)] \end{aligned} \tag{3.12}$$

Therefore,

$$\mathbb{E}[\|\nabla f(w;\xi)\|^2] \overset{(3.10)(3.12)}{\leq} 4L[F(w) - F(w_*)] + 2\mathbb{E}[\|\nabla f(w_*;\xi)\|^2]. \qquad \square$$

Using Lemma 3.2.3 and Super Martingale Convergence Theorem [10] (Lemma 3.2.2), we can provide the sufficient condition for almost sure convergence of Algorithm 2 in the strongly convex case without assuming any bounded gradients.

We note that if $\{\xi_i\}_{i\geq 0}$ are i.i.d. random variables, then $\mathbb{E}[\|\nabla f(w_*;\xi_0)\|^2] = \cdots = \mathbb{E}[\|\nabla f(w_*;\xi_t)\|^2]$. We have the following results for Algorithm 2.

Theorem 3.2.1 (Sufficient conditions for almost sure convergence)**.** *Let Assumptions 3.1.1, 3.2.1 and 3.2.2 hold. Consider Algorithm 2 with a stepsize sequence such that*

$$0 < \eta_t \leq \frac{1}{2L} \ , \ \sum_{t=0}^{\infty} \eta_t = \infty \ \textit{and} \ \sum_{t=0}^{\infty} \eta_t^2 < \infty.$$

Then, the following holds w.p.1 (almost surely)

$$\|w_t - w_*\|^2 \to 0.$$

Proof. Let $\mathcal{F}_t = \sigma(w_0, \xi_0, \ldots, \xi_{t-1})$ be the σ-algebra generated by $w_0, \xi_0, \ldots, \xi_{t-1}$, i.e., $\mathcal{F}_t$ contains all the information of $w_0, \ldots, w_t$. Note that $\mathbb{E}[\nabla f(w_t;\xi_t)|\mathcal{F}_t] = \nabla F(w_t)$. By

Lemma 3.2.3, we have

$$\mathbb{E}[\|\nabla f(w_t;\xi_t)\|^2|\mathcal{F}_t] \leq 4L[F(w_t) - F(w_*)] + N, \tag{3.13}$$

where $N = 2\mathbb{E}[\|\nabla f(w_*;\xi_0)\|^2] = \cdots = 2\mathbb{E}[\|\nabla f(w_*;\xi_t)\|^2]$ since $\{\xi_i\}_{i\geq 0}$ are i.i.d. random variables. Note that $w_{t+1} = w_t - \eta_t \nabla f(w_t;\xi_t)$. Hence,

$$\begin{aligned}
\mathbb{E}[\|w_{t+1} - w_*\|^2|\mathcal{F}_t] &= \mathbb{E}[\|w_t - \eta_t\nabla f(w_t;\xi_t) - w_*\|^2|\mathcal{F}_t] \\
&= \|w_t - w_*\|^2 - 2\eta_t\langle \nabla F(w_t), (w_t - w_*)\rangle + \eta_t^2\mathbb{E}[\|\nabla f(w_t;\xi_t)\|^2|\mathcal{F}_t] \\
&\overset{(3.3)(3.13)}{\leq} \|w_t - w_*\|^2 - \mu\eta_t\|w_t - w_*\|^2 - 2\eta_t[F(w_t) - F(w_*)] \\
&\quad + 4L\eta_t^2[F(w_t) - F(w_*)] + \eta_t^2 N \\
&= \|w_t - w_*\|^2 - \mu\eta_t\|w_t - w_*\|^2 - 2\eta_t(1 - 2L\eta_t)[F(w_t) - F(w_*)] + \eta_t^2 N \\
&\leq \|w_t - w_*\|^2 - \mu\eta_t\|w_t - w_*\|^2 + \eta_t^2 N.
\end{aligned}$$

The last inequality follows since $0 < \eta_t \leq \frac{1}{2L}$. Therefore,

$$\mathbb{E}[\|w_{t+1} - w_*\|^2|\mathcal{F}_t] \leq \|w_t - w_*\|^2 - \mu\eta_t\|w_t - w_*\|^2 + \eta_t^2 N. \tag{3.14}$$

Since $\sum_{t=0}^{\infty}\eta_t^2 N < \infty$, we could apply Lemma 3.2.2. Then, we have w.p.1,

$$\begin{aligned}
&\|w_t - w_*\|^2 \to W \geq 0, \\
\text{and } &\sum_{t=0}^{\infty}\mu\eta_t\|w_t - w_*\|^2 < \infty.
\end{aligned}$$

We want to show that $\|w_t - w_*\|^2 \to 0$, w.p.1. Proving by contradiction, we assume that there exist $\epsilon > 0$ and t_0, s.t. $\|w_t - w_*\|^2 \geq \epsilon$ for $\forall t \geq t_0$. Hence,

$$\sum_{t=0}^{\infty}\mu\eta_t\|w_t - w_*\|^2 \geq \mu\epsilon\sum_{t=0}^{\infty}\eta_t = \infty.$$

This is a contradiction. Therefore, $\|w_t - w_*\|^2 \to 0$ w.p.1. □

Note that the classical SGD proposed in [66] has learning rate satisfying conditions

$$\sum_{t=0}^{\infty} \eta_t = \infty \text{ and } \sum_{t=0}^{\infty} \eta_t^2 < \infty$$

However, the original analysis is performed under the bounded gradient assumption, as in (3.5). In Theorem 3.2.1, on the other hand, we do not use this assumption, but instead assume Lipschitz smoothness and convexity of the function realizations, which does not contradict the strong convexity of $F(w)$.

Theorem 3.2.2. *Let Assumptions 3.1.1, 3.2.1 and 3.2.2 hold. Let* $E = \frac{2\alpha L}{\mu}$ *with* $\alpha = 2$. *Consider Algorithm 2 with a stepsize sequence such that* $\eta_t = \frac{\alpha}{\mu(t+E)} \leq \eta_0 = \frac{1}{2L}$. *The expectation* $\mathbb{E}[\|w_t - w_*\|^2]$ *is at most*

$$\frac{4\alpha^2 N}{\mu^2} \frac{1}{(t - T + E)}$$

for

$$t \geq T = \frac{4L}{\mu} \max\{\frac{L\mu}{N}\|w_0 - w_*\|^2, 1\} - \frac{4L}{\mu}.$$

Proof. Using the beginning of the proof of Theorem 3.2.1, taking the expectation to (3.14), with $0 < \eta_t \leq \frac{1}{2L}$, we have

$$\mathbb{E}[\|w_{t+1} - w_*\|^2] \leq (1 - \mu\eta_t)\mathbb{E}[\|w_t - w_*\|^2] + \eta_t^2 N.$$

We first show that

$$\mathbb{E}[\|w_t - w_*\|^2] \leq \frac{N}{\mu^2} G \frac{1}{(t + E)}, \tag{3.15}$$

where $G = \max\{I, J\}$, and

$$I = \frac{E\mu^2}{N}\mathbb{E}[\|w_0 - w_*\|^2] > 0,$$
$$J = \frac{\alpha^2}{\alpha - 1} > 0.$$

We use mathematical induction to prove (3.15) (this trick is based on the idea from [13]).

Let $t = 0$, we have

$$\mathbb{E}[\|w_0 - w_*\|^2] \leq \frac{NG}{\mu^2 E},$$

which is obviously true since $G \geq \frac{E\mu^2}{N}\|w_0 - w_*\|^2$.

Suppose it is true for t, we need to show that it is also true for $t+1$. We have

$$\begin{aligned}
\mathbb{E}[\|w_{t+1} - w_*\|^2] &\leq \left(1 - \frac{\alpha}{t+E}\right)\frac{NG}{\mu^2(t+E)} + \frac{\alpha^2 N}{\mu^2(t+E)^2} \\
&= \left(\frac{t+E-\alpha}{\mu^2(t+E)^2}\right)NG + \frac{\alpha^2 N}{\mu^2(t+E)^2} \\
&= \left(\frac{t+E-1}{\mu^2(t+E)^2}\right)NG - \left(\frac{\alpha-1}{\mu^2(t+E)^2}\right)NG + \frac{\alpha^2 N}{\mu^2(t+E)^2}.
\end{aligned}$$

Since $G \geq \frac{\alpha^2}{\alpha-1}$,

$$-\left(\frac{\alpha-1}{\mu^2(t+E)^2}\right)NG + \frac{\alpha^2 N}{\mu^2(t+E)^2} \leq 0.$$

This implies

$$\begin{aligned}
\mathbb{E}[\|w_{t+1} - w_*\|^2] &\leq \left(\frac{t+E-1}{\mu^2(t+E)^2}\right)NG \\
&= \left(\frac{(t+E)^2-1}{(t+E)^2}\right)\frac{NG}{\mu^2(t+E+1)} \\
&\leq \frac{NG}{\mu^2(t+E+1)}.
\end{aligned}$$

This proves (3.15) by induction in t.

Notice that the induction proof of (3.15) holds more generally for $E \geq \frac{2\alpha L}{\mu}$ with $\alpha > 1$ (this is sufficient for showing $\eta_t \leq \frac{1}{2L}$. In this more general interpretation we can see that the convergence rate is minimized for I minimal, i.e., $E = \frac{2\alpha L}{\mu}$ and for this reason we have fixed E as such in the theorem statement.

Notice that

$$G = \max\{I, J\} = \max\{\frac{2\alpha L\mu}{N}\mathbb{E}[\|w_0 - w_*\|^2], \frac{\alpha^2}{\alpha-1}\}.$$

We choose $\alpha = 2$ such that η_t only depends on known parameters μ and L. For this α we

obtain

$$G = 4\max\{\frac{L\mu}{N}\mathbb{E}[\|w_0 - w_*\|^2], 1\}.$$

For $T = \frac{4L}{\mu}\max\{\frac{L\mu}{N}\mathbb{E}[\|w_0 - w_*\|^2], 1\} - \frac{4L}{\mu}$, we have that according to (3.15)

$$\begin{aligned}\frac{L\mu}{N}\mathbb{E}[\|w_T - w_*\|^2] &\leq \frac{L\mu}{N}\frac{N}{\mu^2}\frac{G}{(T+E)} \\ &= \frac{L}{\mu}\frac{4\max\{\frac{L\mu}{N}\mathbb{E}[\|w_0 - w_*\|^2], 1\}}{\frac{4L}{\mu}\max\{\frac{L\mu}{N}\mathbb{E}[\|w_0 - w_*\|^2], 1\}} = 1. \end{aligned} \tag{3.16}$$

Applying (3.15)with w_T as starting point rather than w_0 gives, for $t \geq \max\{T, 0\}$,

$$\mathbb{E}[\|w_t - w_*\|^2] \leq \frac{N}{\mu^2}G\frac{1}{(t - T + E)},$$

where G is now equal to

$$4\max\{\frac{L\mu}{N}\mathbb{E}[\|w_T - w_*\|^2], 1\},$$

which equals 4, see (3.16). For any given w_0, we prove the theorem. □

3.2.2 Convergence Analysis without Convexity

In this section, we provide the analysis of Algorithm 2 without using Assumption 3.2.2, that is, $f(w;\xi)$ is not necessarily convex. We still do not need to impose the bounded stochastic gradient assumption, since we can derive an analogue of Lemma 3.2.3, albeit with worse constant in the bound.

Lemma 3.2.4. *Let Assumptions 3.1.1 and 3.2.1 hold. Then, for* $\forall w \in \mathbb{R}^d$,

$$\mathbb{E}[\|\nabla f(w;\xi)\|^2] \leq 4L\kappa[F(w) - F(w_*)] + N, \tag{3.17}$$

where $\kappa = \frac{L}{\mu}$ *and* $N = 2\mathbb{E}[\|\nabla f(w_*;\xi)\|^2]$*;* ξ *is a random variable, and* $w_* = \arg\min_w F(w)$.

Proof. Analogous to the proof of Lemma 3.2.3, we have

Hence,

$$\frac{1}{2}\mathbb{E}[\|\nabla f(w;\xi)\|^2] - \mathbb{E}[\|\nabla f(w_*;\xi)\|^2] = \mathbb{E}\left[\frac{1}{2}\|\nabla f(w;\xi)\|^2 - \|\nabla f(w_*;\xi)\|^2\right]$$

$$\begin{aligned}
&\overset{(3.11)}{\leq} \mathbb{E}[\|\nabla f(w;\xi) - \nabla f(w_*;\xi)\|^2] \\
&\overset{(5.5)}{\leq} L^2\|w - w_*\|^2 \\
&\overset{(3.3)}{\leq} \frac{2L^2}{\mu}[F(w) - F(w_*)] = 2L\kappa[F(w) - F(w_*)].
\end{aligned} \tag{3.18}$$

Therefore,

$$\mathbb{E}[\|\nabla f(w;\xi)\|^2] \overset{(3.10)(3.18)}{\leq} 4L\kappa[F(w) - F(w_*)] + 2\mathbb{E}[\|\nabla f(w_*;\xi)\|^2].$$

□

Based on the proofs of Theorems 3.2.1 and 3.2.2, we can easily have the following two results (Theorems 3.2.3 and 3.2.4).

Theorem 3.2.3 (Sufficient conditions for almost sure convergence)**.** *Let Assumptions 3.1.1 and 3.2.1 hold. Then, we can conclude the statement of Theorem 3.2.1 with the definition of the step size replaced by* $0 < \eta_t \leq \frac{1}{2L\kappa}$ *with* $\kappa = \frac{L}{\mu}$.

Theorem 3.2.4. *Let Assumptions 3.1.1 and 3.2.1 hold. Then, we can conclude the statement of Theorem 3.2.2 with the definition of the step size replaced by* $\eta_t = \frac{\alpha}{\mu(t+E)} \leq \eta_0 = \frac{1}{2L\kappa}$ *with* $\kappa = \frac{L}{\mu}$ *and* $\alpha = 2$*, and all other occurrences of* L *in* E *and* T *replaced by* $L\kappa$.

We compare our result in Theorem 3.2.4 with that in [13] in the following remark.

Remark 3.2.1. *By strong convexity of* F*, Lemma 3.2.4 implies* $\mathbb{E}[\|\nabla f(w;\xi)\|^2] \leq 2\kappa^2\|\nabla F(w)\|^2 + N$*, for* $\forall w \in \mathbb{R}^d$*, where* $\kappa = \frac{L}{\mu}$ *and* $N = 2\mathbb{E}[\|\nabla f(w_*;\xi)\|^2]$*. We can now substitute the value* $M = 2\kappa^2$ *into Theorem 4.7 in [13]. We observe that the resulting initial learning rate in [13] has to satisfy* $\eta_0 \leq \frac{1}{2L\kappa^2}$ *while our results allows* $\eta_0 = \frac{1}{2L\kappa}$*. We are able to achieve this improvement by introducing Assumption 3.2.1, which holds for many ML problems.*

Recall that under Assumption 3.2.2, our initial learning rate is $\eta_0 = \frac{1}{2L}$ *(in Theorem 3.2.2). Thus Assumption 3.2.2 provides further improvement of the conditions on the learning rate.*

3.3 Asynchronous Stochastic Optimization aka Hogwild!

Hogwild! [64] is an asynchronous stochastic optimization method where writes to and reads from vector positions in shared memory can be inconsistent (this corresponds to (3.22) as we shall see). However, as mentioned in [44], for the purpose of analysis the method in [64] performs single vector entry updates that are randomly selected from the non-zero entries of the computed gradient as in (3.21) (explained later) and requires the assumption of consistent vector reads together with the bounded gradient assumption to prove convergence. Both [44] and [15] prove the same result for fixed step size based on the assumption of bounded stochastic gradients in the strongly convex case but now without assuming consistent vector reads and writes. In these works the fixed step size η must depend on σ from the bounded gradient assumption, however, one does not usually know σ and thus, we cannot compute a suitable η a-priori.

As claimed by the authors in [44], they can eliminate the bounded gradient assumption in their analysis of Hogwild!, which however was only mentioned as a remark without proof. On the other hand, the authors of recent unpublished work [35] formulate and prove, without the bounded gradient assumption, a precise theorem about the convergence rate of Hogwild! of the form

$$\mathbb{E}[\|w_t - w_*\|^2] \leq (1-\rho)^t (2\|w_0 - w_*\|^2) + b,$$

where ρ is a function of several parameters but independent of the fixed chosen step size η and where b is a function of several parameters and has a linear dependency with respect to the fixed step size, i.e., $b = O(\eta)$.

In this section, we discuss the convergence of Hogwild! with **diminishing** stepsize where writes to and reads from vector positions in shared memory can be **inconsistent**. This is a slight modification of the original Hogwild! where the stepsize is fixed. In our analysis we also **do not use the bounded gradient assumption** as in [35]. Moreover, (a) we focus on solving the **more general problem** in (3.1), while [35] considers the specific case of the "finite-sum" problem in (3.2), and (b) we show that our analysis generalizes to the **non-convex case**, i.e., we do not need to assume functions $f(w;\xi)$ are convex (we only require $F(w) = \mathbb{E}[f(w;\xi)]$ to be strongly convex) as opposed to the assumption in [35].

3.3.1 Recursion

We first formulate a general recursion for w_t to which our analysis applies, next we will explain how the different variables in the recursion interact and describe two special cases, and finally we present pseudo code of the algorithm using the recursion.

The recursion explains which positions in w_t should be updated in order to compute w_{t+1}. Since w_t is stored in shared memory and is being updated in a possibly non-consistent way by multiple cores who each perform recursions, the shared memory will contain a vector w whose entries represent a mix of updates. That is, before performing the computation of a recursion, a core will first read w from shared memory, however, while reading w from shared memory, the entries in w are being updated out of order. The final vector $\hat{w}_t$ read by the core represents an aggregate of a mix of updates in previous iterations.

The general recursion is defined as follows: For $t \geq 0$,

$$w_{t+1} = w_t - \eta_t d_{\xi_t} S_{u_t}^{\xi_t} \nabla f(\hat{w}_t; \xi_t), \tag{3.19}$$

where

- $\hat{w}_t$ represents the vector used in computing the gradient $\nabla f(\hat{w}_t; \xi_t)$ and whose entries have been read (one by one) from an aggregate of a mix of previous updates that led to w_j, $j \leq t$, and
- the $S_{u_t}^{\xi_t}$ are diagonal 0/1-matrices with the property that there exist real numbers d_ξ satisfying

$$d_\xi \mathbb{E}[S_u^\xi | \xi] = D_\xi, \tag{3.20}$$

  where the expectation is taken over u and D_ξ is the diagonal 0/1 matrix whose 1-entries correspond to the non-zero positions in $\nabla f(w; \xi)$, i.e., the i-th entry of D_ξ's diagonal is equal to 1 if and only if there exists a w such that the i-th position of $\nabla f(w; \xi)$ is non-zero.

The role of matrix $S_{u_t}^{\xi_t}$ is that it filters which positions of gradient $\nabla f(\hat{w}_t; \xi_t)$ play a role in (3.19) and need to be computed. Notice that D_ξ represents the support of $\nabla f(w; \xi)$; by

$|D_\xi|$ we denote the number of 1s in D_ξ, i.e., $|D_\xi|$ equals the size of the support of $\nabla f(w;\xi)$.

We will restrict ourselves to choosing (i.e., fixing a-priori) *non-empty* matrices S_u^ξ that "partition" D_ξ in D approximately "equally sized" S_u^ξ:

$$\sum_u S_u^\xi = D_\xi,$$

where each matrix S_u^ξ has either $\lfloor |D_\xi|/D \rfloor$ or $\lceil |D_\xi|/D \rceil$ ones on its diagonal. We uniformly choose one of the matrices $S_{u_t}^{\xi_t}$ in (3.19), hence, d_ξ equals the number of matrices S_u^ξ, see (3.20).

In other to explain recursion (3.19) we first consider two special cases. For $D = \bar{\Delta}$, where

$$\bar{\Delta} = \max_\xi \{|D_\xi|\}$$

represents the maximum number of non-zero positions in any gradient computation $f(w;\xi)$, we have that for all ξ, there are exactly $|D_\xi|$ diagonal matrices S_u^ξ with a single 1 representing each of the elements in D_ξ. Since $p_\xi(u) = 1/|D_\xi|$ is the uniform distribution, we have $\mathbb{E}[S_u^\xi|\xi] = D_\xi/|D_\xi|$, hence, $d_\xi = |D_\xi|$. This gives the recursion

$$w_{t+1} = w_t - \eta_t |D_\xi| [\nabla f(\hat{w}_t;\xi_t)]_{u_t}, \tag{3.21}$$

where $[\nabla f(\hat{w}_t;\xi_t)]_{u_t}$ denotes the u_t-th position of $\nabla f(\hat{w}_t;\xi_t)$ and where u_t is a uniformly selected position that corresponds to a non-zero entry in $\nabla f(\hat{w}_t;\xi_t)$.

At the other extreme, for $D = 1$, we have exactly one matrix $S_1^\xi = D_\xi$ for each ξ, and we have $d_\xi = 1$. This gives the recursion

$$w_{t+1} = w_t - \eta_t \nabla f(\hat{w}_t;\xi_t). \tag{3.22}$$

Recursion (3.22) represents Hogwild!. In a single-core setting where updates are done in a consistent way and $\hat{w}_t = w_t$ yields SGD.

Algorithm 3 gives the pseudo code corresponding to recursion (3.19) with our choice of sets S_u^ξ (for parameter D).

Algorithm 3 Hogwild! general recursion

Input: $w_0 \in \mathbb{R}^d$
for $t = 0, 1, 2, \ldots$ **in parallel do**
  read each position of shared memory w denoted by $\hat{w}_t$ **(each position read is atomic)**
  draw a random sample ξ_t and a random "filter" $S_{u_t}^{\xi_t}$
  for positions h where $S_{u_t}^{\xi_t}$ has a 1 on its diagonal **do**
    compute g_h as the gradient $\nabla f(\hat{w}_t; \xi_t)$ at position h
    add $\eta_t d_{\xi_t} g_h$ to the entry at position h of w in shared memory **(each position update is atomic)**
  end for
end for

3.3.2 Analysis

Besides Assumptions 3.1.1, 3.2.1, and for now 3.2.2, we assume the following assumption regarding a parameter τ, called the delay, which indicates which updates in previous iterations have certainly made their way into shared memory w.

Assumption 3.3.1 (Consistent with delay τ)**.** *We say that shared memory is consistent with delay τ with respect to recursion (3.19) if, for all t, vector $\hat{w}_t$ includes the aggregate of the updates up to and including those made during the $(t-\tau)$-th iteration (where (3.19) defines the $(t+1)$-st iteration). Each position read from shared memory is atomic and each position update to shared memory is atomic (in that these cannot be interrupted by another update to the same position).*

In other words in the $(t+1)$-th iteration, $\hat{w}_t$ equals $w_{t-\tau}$ plus some subset of position updates made during iterations $t-\tau, t-\tau+1, \ldots, t-1$. We assume that there exists a constant delay τ satisfying Assumption 3.3.1.

Section 3.4 proves the following theorem where

$$\bar{\Delta}_D \stackrel{\text{def}}{=} D \cdot \mathbb{E}[\lceil |D_\xi|/D \rceil].$$

Theorem 3.3.1. *Suppose Assumptions 3.1.1, 3.2.1, 3.2.2 and 3.3.1 and consider Algorithm 3 for sets S_u^ξ with parameter D. Let $\eta_t = \frac{\alpha_t}{\mu(t+E)}$ with $4 \leq \alpha_t \leq \alpha$ and $E = \max\{2\tau, \frac{4L\alpha D}{\mu}\}$. Then, the expected number of single vector entry updates after t iterations*

is equal to

$$t' = t\bar{\Delta}_D/D$$

and expectations $\mathbb{E}[\|\hat{w}_t - w_*\|^2]$ *and* $\mathbb{E}[\|w_t - w_*\|^2]$ *are at most*

$$\frac{4\alpha^2 DN}{\mu^2}\frac{t}{(t+E-1)^2} + O\left(\frac{\ln t}{(t+E-1)^2}\right).$$

In terms of t', the expected number single vector entry updates after t iterations, $\mathbb{E}[\|\hat{w}_t - w_*\|^2]$ and $\mathbb{E}[\|w_t - w_*\|^2]$ are at most

$$\frac{4\alpha^2 \bar{\Delta}_D N}{\mu^2}\frac{1}{t'} + O\left(\frac{\ln t'}{t'^2}\right).$$

Remark 3.3.1. *In (3.21)* $D = \bar{\Delta}$*, hence,* $\lceil |D_\xi|/D \rceil = 1$ *and* $\bar{\Delta}_D = \bar{\Delta} = \max_\xi\{|D_\xi|\}$*. In (3.22)* $D = 1$*, hence,* $\bar{\Delta}_D = \mathbb{E}[|D_\xi|]$*. This shows that the upper bound in Theorem 3.3.1 is better for (3.22) with* $D = 1$*. If we assume no delay, i.e.* $\tau = 0$*, in addition to* $D = 1$*, then we obtain SGD. Theorem 3.2.2 shows that, measured in* t'*, we obtain the upper bound*

$$\frac{4\alpha_{SGD}^2 \bar{\Delta}_D N}{\mu^2}\frac{1}{t'}$$

with $\alpha_{SGD} = 2$ *as opposed to* $\alpha \geq 4$*.*

With respect to parallelism, SGD assumes a single core, while (3.22) and (3.21) allow multiple cores. Notice that recursion (3.21) allows us to partition the position of the shared memory among the different processor cores in such a way that each partition can only be updated by its assigned core and where partitions can be read by all cores. This allows optimal resource sharing and could make up for the difference between $\bar{\Delta}_D$ *for (3.21) and (3.22). We hypothesize that, for a parallel implementation,* D *equal to a fraction of* $\bar{\Delta}$ *will lead to best performance.*

Remark 3.3.2. *Surprisingly, the leading term of the upper bound on the convergence rate is independent of delay* τ*. On one hand, one would expect that a more recent read which contains more of the updates done during the last* τ *iterations will lead to better convergence. When inspecting the second order term in the proof in Section 3.4, we do see that a smaller*

τ (and/or smaller sparsity) makes the convergence rate smaller. That is, asymptotically t should be large enough as a function of τ (and other parameters) in order for the leading term to dominate.

Nevertheless, in asymptotic terms (for larger t) the dependence on τ is not noticeable. In fact, Section 3.4 shows that we may allow τ to be a monotonic increasing function of t with

$$\frac{2L\alpha D}{\mu} \leq \tau(t) \leq \sqrt{t \cdot L(t)},$$

where $L(t) = \frac{1}{\ln t} - \frac{1}{(\ln t)^2}$ (this will make $E = \max\{2\tau(t), \frac{4L\alpha D}{\mu}\}$ also a function of t). The leading term of the convergence rate does not change while the second order terms increase to $O(\frac{1}{t \ln t})$. We show that, for

$$t \geq T_0 = \exp[2\sqrt{\Delta}(1 + \frac{(L+\mu)\alpha}{\mu})],$$

where $\Delta = \max_i \mathbb{P}(i \in D_\xi)$ measures sparsity, the higher order terms that contain $\tau(t)$ (as defined above) are at most the leading term.

Our intuition behind this phenomenon is that for large τ, all the last τ iterations before the t-th iteration use vectors $\hat{w}_j$ with entries that are dominated by the aggregate of updates that happened till iteration $t - \tau$. Since the average sum of the updates during the last τ iterations is equal to

$$-\frac{1}{\tau} \sum_{j=t-\tau}^{t-1} \eta_j d_{\xi_j} S_{u_j}^{\xi_j} \nabla f(\hat{w}_j; \xi_t) \tag{3.23}$$

and all $\hat{w}_j$ look alike in that they mainly represent learned information before the $(t-\tau)$-th iteration, (3.23) becomes an estimate of the expectation of (3.23), i.e.,

$$\sum_{j=t-\tau}^{t-1} \frac{-\eta_j}{\tau} \mathbb{E}[d_{\xi_j} S_{u_j}^{\xi_j} \nabla f(\hat{w}_j; \xi_t)] = \sum_{j=t-\tau}^{t-1} \frac{-\eta_j}{\tau} \nabla F(\hat{w}_j). \tag{3.24}$$

This looks like GD which in the strong convex case has convergence rate $\leq c^{-t}$ for some constant $c > 1$. This already shows that larger τ could help convergence as well. However, estimate (3.23) has estimation noise with respect to (3.24) which explains why in this thought experiment we cannot attain c^{-t} but can only reach a much smaller convergence rate of e.g.

$O(1/t)$ *as in Theorem 3.3.1.*

Experiments in Section 3.5 confirm our analysis.

Remark 3.3.3. *The higher order terms in the proof in Section 3.4 show that, as in Theorem 3.2.2, the expected convergence rate in Theorem 3.3.1 depends on* $\|w_0 - w_*\|^2$. *The proof shows that, for*

$$t \geq T_1 = \frac{\mu^2}{\alpha^2 ND}\|w_0 - w_*\|^2,$$

the higher order term that contains $\|w_0 - w_*\|^2$ *is at most the leading term. This is comparable to* T *in Theorem 3.2.2 for SGD.*

Remark 3.3.4. *Step size* $\eta_t = \frac{\alpha_t}{\mu(t+E)}$ *with* $4 \leq \alpha_t \leq \alpha$ *can be chosen to be fixed during periods whose ranges exponentially increase. For* $t + E \in [2^h, 2^{h+1})$ *we define* $\alpha_t = \frac{4(t+E)}{2^h}$. *Notice that* $4 \leq \alpha_t < 8$ *which satisfies the conditions of Theorem 3.3.1 for* $\alpha = 8$. *This means that we can choose*

$$\eta_t = \frac{\alpha_t}{\mu(t+E)} = \frac{4}{\mu 2^h}$$

as step size for $t + E \in [2^h, 2^{h+1})$. *This choice for* η_t *allows changes in* η_t *to be easily synchronized between cores since these changes only happen when* $t + E = 2^h$ *for some integer* h. *That is, if each core is processing iterations at the same speed, then each core on its own may reliably assume that after having processed* $(2^h - E)/P$ *iterations the aggregate of all* P *cores has approximately processed* $2^h - E$ *iterations. So, after* $(2^h - E)/P$ *iterations a core will increment its version of* h *to* $h+1$. *This will introduce some noise as the different cores will not increment their* h *versions at exactly the same time, but this only happens during a small interval around every* $t + E = 2^h$. *This will occur rarely for larger* h.

3.3.3 Convergence Analysis without Convexity

In Section 3.4, we also show that the proof of Theorem 3.3.1 can easily be modified such that Theorem 3.3.1 with $E \geq \frac{4L\kappa\alpha D}{\mu}$ also holds in the non-convex case of the component functions, i.e., we do not need Assumption 3.2.2. Note that this case is not analyzed in [35].

Theorem 3.3.2. *Let Assumptions 3.1.1 and 3.2.1 hold. Then, we can conclude the statement of Theorem 3.3.1 with* $E \geq \frac{4L\kappa\alpha D}{\mu}$ *for* $\kappa = \frac{L}{\mu}$.

We then provide the analysis for Algorithm 3 in detail in the following section.

3.4 Analysis for Algorithm 3

3.4.1 Recurrence and Notation

We introduce the following notation: For each ξ, we define $D_\xi \subseteq \{1, \ldots, d\}$ as the set of possible non-zero positions in a vector of the form $\nabla f(w; \xi)$ for some w. We consider a fixed mapping from $u \in U$ to subsets $S_u^\xi \subseteq D_\xi$ for each possible ξ. In our notation we also let D_ξ represent the diagonal $d \times d$ matrix with ones exactly at the positions corresponding to D_ξ and with zeroes elsewhere. Similarly, S_u^ξ also denotes a diagonal matrix with ones at the positions corresponding to D_ξ.

We will use a probability distribution $p_\xi(u)$ to indicate how to randomly select a matrix S_u^ξ. We choose the matrices S_u^ξ and distribution $p_\xi(u)$ so that there exist d_ξ such that

$$d_\xi \mathbb{E}[S_u^\xi | \xi] = D_\xi, \tag{3.25}$$

where the expectation is over $p_\xi(u)$.

We will restrict ourselves to choosing *non-empty* sets S_u^ξ that partition D_ξ in D approximately equally sized sets together with uniform distributions $p_\xi(u)$ for some fixed D. So, if $D \leq |D_\xi|$, then sets have sizes $\lfloor |D_\xi|/D \rfloor$ and $\lceil |D_\xi|/D \rceil$. For the special case $D > |D_\xi|$ we have exactly $|D_\xi|$ singleton sets of size 1 (in our definition we only use non-empty sets).

For example, for $D = \bar{\Delta}$, where

$$\bar{\Delta} = \max_\xi \{|D_\xi|\}$$

represents the maximum number of non-zero positions in any gradient computation $f(w; \xi)$, we have that for all ξ, there are exactly $|D_\xi|$ singleton sets S_u^ξ representing each of the elements in D_ξ. Since $p_\xi(u) = 1/|D_\xi|$ is the uniform distribution, we have $\mathbb{E}[S_u^\xi|\xi] = D_\xi/|D_\xi|$, hence, $d_\xi = |D_\xi|$. As another example at the other extreme, for $D = 1$, we have exactly one set $S_1^\xi = D_\xi$ for each ξ. Now $p_\xi(1) = 1$ and we have $d_\xi = 1$.

We define the parameter

$$\bar{\Delta}_D \stackrel{\text{def}}{=} D \cdot \mathbb{E}[\lceil |D_\xi|/D \rceil],$$

where the expectation is over ξ. We use $\bar{\Delta}_D$ in the leading asymptotic term for the convergence rate in our main theorem. We observe that

$$\bar{\Delta}_D \leq \mathbb{E}[|D_\xi|] + D - 1$$

and $\bar{\Delta}_D \leq \bar{\Delta}$ with equality for $D = \bar{\Delta}$.

For completeness we define

$$\Delta \stackrel{\text{def}}{=} \max_i \mathbb{P}\left(i \in D_\xi\right).$$

Let us remark, that $\Delta \in (0, 1]$ measures the probability of collision. Small Δ means that there is a small chance that the support of two random realizations of $\nabla f(w; \xi)$ will have an intersection. On the other hand, $\Delta = 1$ means that almost surely, the support of two stochastic gradients will have non-empty intersection.

With this definition of Δ it is an easy exercise to show that for iid ξ_1 and ξ_2 in a finite-sum setting (i.e., ξ_i and ξ_2 can only take on a finite set of possible values) we have

$$\begin{aligned} &\mathbb{E}[|\langle \nabla f(w_1; \xi_1), \nabla f(w_2; \xi_2) \rangle|] \\ &\leq \frac{\sqrt{\Delta}}{2} \left(\mathbb{E}[\|\nabla f(w_1; \xi_1)\|^2] + \mathbb{E}[\|\nabla f(w_2; \xi_2)\|^2] \right) \end{aligned} \tag{3.26}$$

(see Proposition 10 in [35]). We notice that in the non-finite sum setting we can use the property that for any two vectors a and b, $\langle a, b \rangle \leq (\|a\|^2 + \|b\|^2)/2$ and this proves (3.26) with Δ set to $\Delta = 1$. In our asymptotic analysis of the convergence rate, we will show how Δ plays a role in non-leading terms – this, with respect to the leading term, it will not matter whether we use $\Delta = 1$ or Δ equal the probability of collision (in the finite sum case).

We have

$$w_{t+1} = w_t - \eta_t d_{\xi_t} S_{u_t}^{\xi_t} \nabla f(\hat{w}_t; \xi_t), \tag{3.27}$$

where $\hat{w}_t$ represents the vector used in computing the gradient $\nabla f(\hat{w}_t; \xi_t)$ and whose entries

have been read (one by one) from an aggregate of a mix of previous updates that led to w_j, $j \leq t$. Here, we assume that

- updating/writing to vector positions is atomic, reading vector positions is atomic, and
- there exists a "delay" τ such that, for all t, vector $\hat{w}_t$ includes all the updates up to and including those made during the $(t-\tau)$-th iteration (where (3.27) defines the $(t+1)$-st iteration).

Notice that we do **not assume consistent reads and writes of vector positions**. We only assume that up to a "delay" τ all writes/updates are included in the values of positions that are being read.

According to our definition of τ, in (3.27) vector $\hat{w}_t$ represents an inconsistent read with entries that contain all of the updates made during the 1st to $(t-\tau)$-th iteration. Furthermore each entry in $\hat{w}_t$ includes some of the updates made during the $(t-\tau+1)$-th iteration up to t-th iteration. Each entry includes its own subset of updates because writes are inconsistent. We model this by "masks" $\Sigma_{t,j}$ for $t-\tau \leq j \leq t-1$. A mask $\Sigma_{t,j}$ is a diagonal 0/1-matrix with the 1s expressing which of the entry updates made in the $(j+1)$-th iteration are included in $\hat{w}_t$. That is,

$$\hat{w}_t = w_{t-\tau} - \sum_{j=t-\tau}^{t-1} \eta_j d_{\xi_j} \Sigma_{t,j} S_{u_j}^{\xi_j} \nabla f(\hat{w}_j; \xi_j). \tag{3.28}$$

Notice that the recursion (3.27) implies

$$w_t = w_{t-\tau} - \sum_{j=t-\tau}^{t-1} \eta_j d_{\xi_j} S_{u_j}^{\xi_j} \nabla f(\hat{w}_j; \xi_j). \tag{3.29}$$

By combining (3.29) and (3.28) we obtain

$$w_t - \hat{w}_t = - \sum_{j=t-\tau}^{t-1} \eta_j d_{\xi_j} (I - \Sigma_{t,j}) S_{u_j}^{\xi_j} \nabla f(\hat{w}_j; \xi_j), \tag{3.30}$$

where I represents the identity matrix.

3.4.2 Main Analysis

We first derive a couple lemmas which will help us deriving our main bounds. In what follows let Assumptions 3.1.1, 3.2.1, 3.2.2 and 3.3.1 hold for all lemmas. We define

$$\mathcal{F}_t = \sigma(w_0, \xi_1, u_1, \sigma_1, \ldots, \xi_{t-1}, u_{t-1}, \sigma_{t-1}),$$

where

$$\sigma_{t-1} = (\Sigma_{t,t-\tau}, \ldots, \Sigma_{t,t-1}).$$

When we subtract τ from, for example, t and write $t-\tau$, we will actually mean $\max\{t-\tau, 0\}$.

Lemma 3.4.1. *We have*

$$\mathbb{E}[\|d_{\xi_t} S_{u_t}^{\xi_t} \nabla f(\hat{w}_t; \xi_t)\|^2 | \mathcal{F}_t, \xi_t] \leq D \|\nabla f(\hat{w}_t; \xi_t)\|^2$$

and

$$\mathbb{E}[d_{\xi_t} S_{u_t}^{\xi_t} \nabla f(\hat{w}_t; \xi_t) | \mathcal{F}_t] = \nabla F(\hat{w}_t).$$

Proof. For the first bound, if we take the expectation of $\|d_{\xi_t} S_{u_t}^{\xi_t} \nabla f(\hat{w}_t; \xi_t)\|^2$ with respect to u_t, then we have (for vectors x we denote the value if its i-th position by $[x]_i$)

$$\begin{aligned}
&\mathbb{E}[\|d_{\xi_t} S_{u_t}^{\xi_t} \nabla f(\hat{w}_t; \xi_t)\|^2 | \mathcal{F}_t, \xi_t] = d_{\xi_t}^2 \sum_u p_{\xi_t}(u) \|S_u^{\xi_t} \nabla f(\hat{w}_t; \xi_t)\|^2 \\
&= d_{\xi_t}^2 \sum_u p_{\xi_t}(u) \sum_{i \in S_u^{\xi_t}} [\nabla f(\hat{w}_t; \xi_t)]_i^2 \\
&= d_{\xi_t} \sum_{i \in D_{\xi_t}} [\nabla f(\hat{w}_t; \xi_t)]_i^2 = d_{\xi_t} \|f(\hat{w}_t; \xi_t)\|^2 \leq D \|\nabla f(\hat{w}_t; \xi_t)\|^2,
\end{aligned}$$

where the transition to the second line follows from (3.25).

For the second bound, if we take the expectation of $d_{\xi_t} S_{u_t}^{\xi_t} \nabla f(\hat{w}_t; \xi_t)$ wrt u_t, then we have:

$$\mathbb{E}[d_{\xi_t} S_{u_t}^{\xi_t} \nabla f(\hat{w}_t; \xi_t) | \mathcal{F}_t, \xi_t] = d_{\xi_t} \sum_u p_{\xi_t}(u) S_u^{\xi_t} \nabla f(\hat{w}_t; \xi_t) = D_{\xi_t} \nabla f(\hat{w}_t; \xi_t) = \nabla f(\hat{w}_t; \xi_t),$$

and this can be used to derive

$$\mathbb{E}[d_{\xi_t} S_{u_t}^{\xi_t} f(\hat{w}_t;\xi_t)|\mathcal{F}_t] = \mathbb{E}[\mathbb{E}[d_{\xi_t} S_{u_t}^{\xi_t} f(\hat{w}_t;\xi_t)|\mathcal{F}_t,\xi_t]|\mathcal{F}_t] = \nabla F(\hat{w}_t).$$

□

As a consequence of this lemma we derive a bound on the expectation of $\|w_t - \hat{w}_t\|^2$.

Lemma 3.4.2. *The expectation of* $\|w_t - \hat{w}_t\|^2$ *is at most*

$$\mathbb{E}[\|w_t - \hat{w}_t\|^2] \leq (1+\sqrt{\Delta}\tau) D \sum_{j=t-\tau}^{t-1} \eta_j^2 (2L^2 \mathbb{E}[\|\hat{w}_j - w_*\|^2] + N).$$

Proof. As shown in (3.30),

$$w_t - \hat{w}_t = -\sum_{j=t-\tau}^{t-1} \eta_j d_{\xi_j} (I - \Sigma_{t,j}) S_{u_j}^{\xi_j} \nabla f(\hat{w}_j;\xi_j).$$

This can be used to derive an expression for the square of its norm:

$$\begin{aligned}
\|w_t - \hat{w}_t\|^2 &= \|\sum_{j=t-\tau}^{t-1} \eta_j d_{\xi_j} (I - \Sigma_{t,j}) S_{u_j}^{\xi_j} \nabla f(\hat{w}_j;\xi_j)\|^2 \\
&= \sum_{j=t-\tau}^{t-1} \|\eta_j d_{\xi_j} (I - \Sigma_{t,j}) S_{u_j}^{\xi_j} \nabla f(\hat{w}_j;\xi_j)\|^2 \\
&+ \sum_{i\neq j \in \{t-\tau,\ldots,t-1\}} \langle \eta_j d_{\xi_j} (I - \Sigma_{t,j}) S_{u_j}^{\xi_j} \nabla f(\hat{w}_j;\xi_j), \eta_i d_{\xi_i} (I - \Sigma_{t,j}) S_{u_i}^{\xi_i} \nabla f(\hat{w}_i;\xi_i)\rangle.
\end{aligned}$$

Applying (3.26) to the inner products implies

$$\begin{aligned}
\|w_t - \hat{w}_t\|^2 &\leq \sum_{j=t-\tau}^{t-1} \|\eta_j d_{\xi_j} (I - \Sigma_{t,j}) S_{u_j}^{\xi_j} \nabla f(\hat{w}_j;\xi_j)\|^2 \\
&\quad + \sum_{i\neq j \in \{t-\tau,\ldots,t-1\}} [\|\eta_j d_{\xi_j} (I - \Sigma_{t,j}) S_{u_j}^{\xi_j} \nabla f(\hat{w}_j;\xi_j)\|^2 \\
&\quad + \|\eta_i d_{\xi_i} (I - \Sigma_{t,j}) S_{u_i}^{\xi_i} \nabla f(\hat{w}_i;\xi_i)\|^2] \sqrt{\Delta}/2 \\
&= (1+\sqrt{\Delta}\tau) \sum_{j=t-\tau}^{t-1} \|\eta_j d_{\xi_j} (I - \Sigma_{t,j}) S_{u_j}^{\xi_j} \nabla f(\hat{w}_j;\xi_j)\|^2
\end{aligned}$$

$$\leq \quad (1+\sqrt{\Delta}\tau)\sum_{j=t-\tau}^{t-1}\eta_j^2\|d_{\xi_j}S_{u_j}^{\xi_j}\nabla f(\hat{w}_j;\xi_j)\|^2.$$

Taking expectations shows

$$\mathbb{E}[\|w_t-\hat{w}_t\|^2]\leq(1+\sqrt{\Delta}\tau)\sum_{j=t-\tau}^{t-1}\eta_j^2\mathbb{E}[\|d_{\xi_j}S_{u_j}^{\xi_j}\nabla f(\hat{w}_j;\xi_j)\|^2].$$

Now, we can apply Lemma 3.4.1: We first take the expectation over u_j and this shows

$$\mathbb{E}[\|w_t-\hat{w}_t\|^2]\leq(1+\sqrt{\Delta}\tau)\sum_{j=t-\tau}^{t-1}\eta_j^2 D\mathbb{E}[\|\nabla f(\hat{w}_j;\xi_j)\|^2].$$

From Lemma 3.2.3 we infer

$$\mathbb{E}[\|\nabla f(\hat{w}_j;\xi_j)\|^2]\leq 4L\mathbb{E}[F(\hat{w}_j)-F(w_*)]+N \tag{3.31}$$

and by L-smoothness, see Equation 5.6 with $\nabla F(w_*)=0$,

$$F(\hat{w}_j)-F(w_*)\leq\frac{L}{2}\|\hat{w}_j-w_*\|^2.$$

Combining the above inequalities proves the lemma. □

Together with the next lemma we will be able to start deriving a recursive inequality from which we will be able to derive a bound on the convergence rate.

Lemma 3.4.3. *Let $0<\eta_t\leq\frac{1}{4LD}$ for all $t\geq 0$. Then,*

$$\mathbb{E}[\|w_{t+1}-w_*\|^2|\mathcal{F}_t]\leq\left(1-\frac{\mu\eta_t}{2}\right)\|w_t-w_*\|^2+[(L+\mu)\eta_t+2L^2\eta_t^2D]\|\hat{w}_t-w_t\|^2+2\eta_t^2DN.$$

Proof. Since $w_{t+1}=w_t-\eta_t d_{\xi_t}S_{u_t}^{\xi_t}\nabla f(\hat{w}_t;\xi_t)$, we have

$$\|w_{t+1}-w_*\|^2=\|w_t-w_*\|^2-2\eta_t\langle d_{\xi_t}S_{u_t}^{\xi_t}\nabla f(\hat{w}_t;\xi_t),(w_t-w_*)\rangle+\eta_t^2\|d_{\xi_t}S_{u_t}^{\xi_t}\nabla f(\hat{w}_t;\xi_t)\|^2.$$

We now take expectations over u_t and ξ_t and use Lemma 3.4.1:

$$\begin{aligned}
& \mathbb{E}[\|w_{t+1} - w_*\|^2 | \mathcal{F}_t] \\
\leq \; & \|w_t - w_*\|^2 - 2\eta_t \langle \nabla F(\hat{w}_t), (w_t - w_*) \rangle + \eta_t^2 D \mathbb{E}[\|\nabla f(\hat{w}_t; \xi_t)\|^2 | \mathcal{F}_t] \\
= \; & \|w_t - w_*\|^2 - 2\eta_t \langle \nabla F(\hat{w}_t), (w_t - \hat{w}_t) \rangle - 2\eta_t \langle \nabla F(\hat{w}_t), (\hat{w}_t - w_*) \rangle \\
& + \eta_t^2 D \mathbb{E}[\|\nabla f(\hat{w}_t; \xi_t)\|^2 | \mathcal{F}_t].
\end{aligned}$$

By (3.3) and (5.6), we have

$$-\langle \nabla F(\hat{w}_t), (\hat{w}_t - w_*) \rangle \leq -[F(\hat{w}_t) - F(w_*)] - \frac{\mu}{2} \|\hat{w}_t - w_*\|^2, \text{ and} \tag{3.32}$$

$$-\langle \nabla F(\hat{w}_t), (w_t - \hat{w}_t) \rangle \leq F(\hat{w}_t) - F(w_t) + \frac{L}{2} \|\hat{w}_t - w_t\|^2 \tag{3.33}$$

Thus, $\mathbb{E}[\|w_{t+1} - w_*\|^2 | \mathcal{F}_t]$ is at most

$$\begin{aligned}
& \overset{(3.32)(3.33)}{\leq} \|w_t - w_*\|^2 + 2\eta_t [F(\hat{w}_t) - F(w_t)] + L\eta_t \|\hat{w}_t - w_t\|^2 - 2\eta_t [F(\hat{w}_t) - F(w_*)] \\
& - \mu\eta_t \|\hat{w}_t - w_*\|^2 + \eta_t^2 D \mathbb{E}[\|\nabla f(\hat{w}_t; \xi_t)\|^2 | \mathcal{F}_t] \\
= \; & \|w_t - w_*\|^2 - 2\eta_t [F(w_t) - F(w_*)] + L\eta_t \|\hat{w}_t - w_t\|^2 - \mu\eta_t \|\hat{w}_t - w_*\|^2 \\
& + \eta_t^2 D \mathbb{E}[\|\nabla f(\hat{w}_t; \xi_t)\|^2 | \mathcal{F}_t].
\end{aligned}$$

Since

$$-\|\hat{w}_t - w_*\|^2 = -\|(w_t - w_*) - (w_t - \hat{w}_t)\|^2 \overset{(3.11)}{\leq} -\frac{1}{2} \|w_t - w_*\|^2 + \|w_t - \hat{w}_t\|^2,$$

$\mathbb{E}[\|w_{t+1} - w_*\|^2 | \mathcal{F}_t, \sigma_t]$ is at most

$$(1 - \frac{\mu\eta_t}{2}) \|w_t - w_*\|^2 - 2\eta_t [F(w_t) - F(w_*)] + (L + \mu)\eta_t \|\hat{w}_t - w_t\|^2 + \eta_t^2 D \mathbb{E}[\|\nabla f(\hat{w}_t; \xi_t)\|^2 | \mathcal{F}_t].$$

We now use $\|a\|^2 = \|a - b + b\|^2 \leq 2\|a - b\|^2 + 2\|b\|^2$ for $\mathbb{E}[\|\nabla f(\hat{w}_t; \xi_t)\|^2 | \mathcal{F}_t]$ to obtain

$$\mathbb{E}[\|\nabla f(\hat{w}_t; \xi_t)\|^2 | \mathcal{F}_t] \leq 2\mathbb{E}[\|\nabla f(\hat{w}_t; \xi_t) - \nabla f(w_t; \xi_t)\|^2 | \mathcal{F}_t] + 2\mathbb{E}[\|\nabla f(w_t; \xi_t)\|^2 | \mathcal{F}_t]. \tag{3.34}$$

By Lemma 3.2.3, we have

$$\mathbb{E}[\|\nabla f(w_t;\xi_t)\|^2|\mathcal{F}_t] \leq 4L[F(w_t) - F(w_*)] + N. \tag{3.35}$$

Applying (5.5) twice gives

$$\mathbb{E}[\|\nabla f(\hat{w}_t;\xi_t) - \nabla f(w_t;\xi_t)\|^2|\mathcal{F}_t,\sigma_t] \leq L^2\|\hat{w}_t - w_t\|^2$$

and together with (3.34) and (3.35) we obtain

$$\mathbb{E}[\|\nabla f(\hat{w}_t;\xi_t)\|^2|\mathcal{F}_t] \leq 2L^2\|\hat{w}_t - w_t\|^2 + 4L[F(w_t) - F(w_*)] + N.$$

Plugging this into the previous derivation yields

$$\begin{aligned}
\mathbb{E}[\|w_{t+1} - w_*\|^2|\mathcal{F}_t] &\leq (1 - \frac{\mu\eta_t}{2})\|w_t - w_*\|^2 - 2\eta_t[F(w_t) - F(w_*)] + (L+\mu)\eta_t\|\hat{w}_t - w_t\|^2 \\
&\quad + 2L^2\eta_t^2 D\|\hat{w}_t - w_t\|^2 + 8L\eta_t^2 D[F(w_t) - F(w_*)] + 2\eta_t^2 DN \\
&= (1 - \frac{\mu\eta_t}{2})\|w_t - w_*\|^2 + [(L+\mu)\eta_t + 2L^2\eta_t^2 D]\|\hat{w}_t - w_t\|^2 \\
&\quad - 2\eta_t(1 - 4L\eta_t D)[F(w_t) - F(w_*)] + 2\eta_t^2 DN.
\end{aligned}$$

Since $\eta_t \leq \frac{1}{4LD}$, $-2\eta_t(1 - 4L\eta_t D)[F(w_t) - F(w_*)] \leq 0$ (we can get a negative upper bound by applying strong convexity but this will not improve the asymptotic behavior of the convergence rate in our main result although it would improve the constant of the leading term making the final bound applied to SGD closer to the bound of Theorem 3.2.2 for SGD),

$$\mathbb{E}[\|w_{t+1} - w_*\|^2|\mathcal{F}_t] \leq \left(1 - \frac{\mu\eta_t}{2}\right)\|w_t - w_*\|^2 + [(L+\mu)\eta_t + 2L^2\eta_t^2 D]\|\hat{w}_t - w_t\|^2 + 2\eta_t^2 DN$$

and this concludes the proof. □

Assume $0 < \eta_t \leq \frac{1}{4LD}$ for all $t \geq 0$. Then, after taking the full expectation of the

inequality in Lemma 3.4.3, we can plug Lemma 3.4.2 into it which yields the recurrence

$$\begin{aligned}\mathbb{E}[\|w_{t+1}-w_*\|^2] \;\leq\; & \left(1-\frac{\mu\eta_t}{2}\right)\mathbb{E}[\|w_t-w_*\|^2] \\ & +[(L+\mu)\eta_t+2L^2\eta_t^2D](1+\sqrt{\Delta}\tau)D\sum_{j=t-\tau}^{t-1}\eta_j^2(2L^2\mathbb{E}[\|\hat{w}_j-w_*\|^2]+N) \\ & +2\eta_t^2DN. \end{aligned} \tag{3.36}$$

This can be solved by using the next lemma. For completeness, we follow the convention that an empty product is equal to 1 and an empty sum is equal to 0, i.e.,

$$\prod_{i=h}^{k} g_i = 1 \text{ and } \sum_{i=h}^{k} g_i = 0 \text{ if } k < h. \tag{3.37}$$

Lemma 3.4.4. *Let Y_t, β_t and γ_t be sequences such that $Y_{t+1} \leq \beta_t Y_t + \gamma_t$, for all $t \geq 0$. Then,*

$$Y_{t+1} \leq (\sum_{i=0}^{t}[\prod_{j=i+1}^{t}\beta_j]\gamma_i) + (\prod_{j=0}^{t}\beta_j)Y_0. \tag{3.38}$$

Proof. We prove the lemma by using induction. It is obvious that (3.38) is true for $t = 0$ because $Y_1 \leq \beta_1 Y_0 + \gamma_1$. Assume as induction hypothesis that (3.38) is true for $t-1$. Since $Y_{t+1} \leq \beta_t Y_t + \gamma_t$,

$$\begin{aligned} Y_{t+1} &\leq \beta_t Y_t + \gamma_t \\ &\leq \beta_t[(\sum_{i=0}^{t-1}[\prod_{j=i+1}^{t-1}\beta_j]\gamma_i) + (\prod_{j=0}^{t-1}\beta_j)Y_0] + \gamma_t \\ &\overset{(3.37)}{=} (\sum_{i=0}^{t-1}\beta_t[\prod_{j=i+1}^{t-1}\beta_j]\gamma_i) + \beta_t(\prod_{j=0}^{t-1}\beta_j)Y_0 + (\prod_{j=t+1}^{t}\beta_j)\gamma_t \\ &= [(\sum_{i=0}^{t-1}[\prod_{j=i+1}^{t}\beta_j]\gamma_i) + (\prod_{j=t+1}^{t}\beta_j)\gamma_t] + (\prod_{j=0}^{t}\beta_j)Y_0 \\ &= (\sum_{i=0}^{t}[\prod_{j=i+1}^{t}\beta_j]\gamma_i) + (\prod_{j=0}^{t}\beta_j)Y_0. \end{aligned}$$

□

Applying the above lemma to (3.36) will yield the following bound.

Lemma 3.4.5. *Let* $\eta_t = \frac{\alpha_t}{\mu(t+E)}$ *with* $4 \leq \alpha_t \leq \alpha$ *and* $E = \max\{2\tau, \frac{4L\alpha D}{\mu}\}$. *Then, expectation* $\mathbb{E}[\|w_{t+1} - w_*\|^2]$ *is at most*

$$\frac{\alpha^2 D}{\mu^2}\frac{1}{(t+E-1)^2}\left(\sum_{i=1}^{t}\left[4a_i(1+\sqrt{\Delta}\tau)[N\tau + 2L^2\sum_{j=i-\tau}^{i-1}\mathbb{E}[\|\hat{w}_j - w_*\|^2] + 2N\right]\right) + \frac{(E+1)^2}{(t+E-1)^2}\mathbb{E}[\|w_0 - w_*\|^2],$$

where $a_i = (L+\mu)\eta_i + 2L^2\eta_i^2 D$.

Proof. Notice that we may use (3.36) because $\eta_t \leq \frac{1}{4LD}$ follows from $\eta_t = \frac{\alpha_t}{\mu(t+E)} \leq \frac{\alpha}{\mu(t+E)}$ combined with $E \geq \frac{4L\alpha D}{\mu}$. From (3.36) with $a_t = (L+\mu)\eta_t + 2L^2\eta_t^2 D$ and η_t being decreasing in t we infer

$$\begin{aligned}
& \mathbb{E}[\|w_{t+1} - w_*\|^2] \\
\leq\ & \left(1 - \frac{\mu\eta_t}{2}\right)\mathbb{E}[\|w_t - w_*\|^2] + a_t(1+\sqrt{\Delta}\tau)D\eta_{t-\tau}^2\sum_{j=t-\tau}^{t-1}(2L^2\mathbb{E}[\|\hat{w}_j - w_*\|^2] + N) \\
& + 2\eta_t^2 DN \\
=\ & \left(1 - \frac{\mu\eta_t}{2}\right)\mathbb{E}[\|w_t - w_*\|^2] + a_t(1+\sqrt{\Delta}\tau)D\eta_{t-\tau}^2[N\tau + 2L^2\sum_{j=t-\tau}^{t-1}\mathbb{E}[\|\hat{w}_j - w_*\|^2] \\
& + 2\eta_t^2 DN.
\end{aligned}$$

Since $E \geq 2\tau$, $\frac{1}{t-\tau+E} \leq \frac{2}{t+E}$. Hence, together with $\eta_{t-\tau} = \frac{\alpha_{t-\tau}}{\mu(t-\tau+E)} \leq \frac{\alpha}{\mu(t-\tau+E)}$ we have

$$\eta_{t-\tau}^2 \leq \frac{4\alpha^2}{\mu^2}\frac{1}{(t+E)^2}. \tag{3.39}$$

This translates the above bound into

$$\mathbb{E}[\|w_{t+1} - w_*\|^2] \leq \beta_t\mathbb{E}[\|w_t - w_*\|^2] + \gamma_t,$$

for

$$\beta_t = 1 - \frac{\mu\eta_t}{2},$$

$$\gamma_t = 4a_t(1+\sqrt{\Delta}\tau)D\frac{\alpha^2}{\mu^2}\frac{1}{(t+E)^2}[N\tau + 2L^2\sum_{j=t-\tau}^{t-1}\mathbb{E}[\|\hat{w}_j - w_*\|^2] + 2\eta_t^2 DN, where$$

$$a_t = (L+\mu)\eta_t + 2L^2\eta_t^2 D.$$

Application of Lemma 3.4.4 for $Y_{t+1} = \mathbb{E}[\|w_{t+1} - w_*\|^2]$ and $Y_t = \mathbb{E}[\|w_t - w_*\|^2]$ gives

$$\mathbb{E}[\|w_{t+1} - w_*\|^2] \leq \left(\sum_{i=0}^{t}\left[\prod_{j=i+1}^{t}\left(1 - \frac{\mu\eta_j}{2}\right)\right]\gamma_i\right) + \left(\prod_{j=0}^{t}\left(1 - \frac{\mu\eta_j}{2}\right)\right)\mathbb{E}[\|w_0 - w_*\|^2].$$

In order to analyze this formula, since $\eta_j = \frac{\alpha_j}{\mu(j+E)}$ with $\alpha_j \geq 4$, we have

$$1 - \frac{\mu\eta_j}{2} = 1 - \frac{\alpha_j}{2(j+E)} \leq 1 - \frac{2}{j+E},$$

Hence (we can also use $1 - x \leq e^{-x}$ which leads to similar results and can be used to show that our choice for η_t leads to the tightest convergence rates in our framework),

$$\begin{aligned}\prod_{j=i}^{t}\left(1 - \frac{\mu\eta_j}{2}\right) &\leq \prod_{j=i}^{t}\left(1 - \frac{2}{j+E}\right) = \prod_{j=i}^{t}\frac{j+E-2}{j+E} \\ &= \frac{i+E-2}{i+E}\frac{i+E-1}{i+E+1}\frac{i+E}{i+E+2}\frac{i+E+1}{i+E+3}\cdots\frac{t+E-3}{t+E-1}\frac{t+E-2}{t+E} \\ &= \frac{(i+E-2)(i+E-1)}{(t+E-1)(t+E)} \leq \frac{(i+E-1)^2}{(t+E-1)(t+E)} \leq \frac{(i+E)^2}{(t+E-1)^2}.\end{aligned}$$

From this calculation we infer that

$$\mathbb{E}[\|w_{t+1} - w_*\|^2] \leq \left(\sum_{i=0}^{t}\left[\frac{(i+E)^2}{(t+E-1)^2}\right]\gamma_i\right) + \frac{(E+1)^2}{(t+E-1)^2}\mathbb{E}[\|w_0 - w_*\|^2]. \tag{3.40}$$

Now, we substitute $\eta_i \leq \frac{\alpha}{\mu(i+E)}$ in γ_i and compute

$$\begin{aligned}&\frac{(i+E)^2}{(t+E-1)^2}\gamma_i \\ &= \frac{(i+E)^2}{(t+E-1)^2}4a_i(1+\sqrt{\Delta}\tau)D\frac{\alpha^2}{\mu^2}\frac{1}{(i+E)^2}[N\tau + 2L^2\sum_{j=i-\tau}^{i-1}\mathbb{E}[\|\hat{w}_j - w_*\|^2] \\ &\quad + \frac{(i+E)^2}{(t+E-1)^2}2ND\frac{\alpha^2}{\mu^2(i+E)^2}\end{aligned}$$

$$= \frac{\alpha^2 D}{\mu^2} \frac{1}{(t+E-1)^2} \left[4a_i(1+\sqrt{\Delta}\tau)[N\tau + 2L^2 \sum_{j=i-\tau}^{i-1} \mathbb{E}[\|\hat{w}_j - w_*\|^2] + 2N \right].$$

Substituting this in (3.40) proves the lemma. □

As an immediate corollary we can apply the inequality $\|a+b\|^2 \leq 2\|a\|^2 + 2\|b\|^2$ to $\mathbb{E}[\|\hat{w}_{t+1} - w_*\|^2]$ to obtain

$$\mathbb{E}[\|\hat{w}_{t+1} - w_*\|^2] \leq 2\mathbb{E}[\|\hat{w}_{t+1} - w_{t+1}\|^2] + 2\mathbb{E}[\|w_{t+1} - w_*\|^2], \tag{3.41}$$

which in turn can be bounded by the previous lemma together with Lemma 3.4.2:

$$\begin{aligned}
&\mathbb{E}[\|\hat{w}_{t+1} - w_*\|^2] \\
&\quad \leq 2(1+\sqrt{\Delta}\tau)D \sum_{j=t+1-\tau}^{t} \eta_j^2 (2L^2 \mathbb{E}[\|\hat{w}_j - w_*\|^2] + N) \\
&\quad + 2\frac{\alpha^2 D}{\mu^2} \frac{1}{(t+E-1)^2} \left(\sum_{i=1}^{t} \left[4a_i(1+\sqrt{\Delta}\tau)[N\tau + 2L^2 \sum_{j=i-\tau}^{i-1} \mathbb{E}[\|\hat{w}_j - w_*\|^2] + 2N \right] \right) \\
&\quad + \frac{(E+1)^2}{(t+E-1)^2} \mathbb{E}[\|w_0 - w_*\|^2].
\end{aligned}$$

Now assume a decreasing sequence Z_t for which we want to prove that $\mathbb{E}[\|\hat{w}_t - w_*\|^2] \leq Z_t$ by induction in t. Then, the above bound can be used together with the property that Z_t and η_t are decreasing in t to show

$$\begin{aligned}
\sum_{j=t+1-\tau}^{t} \eta_j^2 (2L^2 \mathbb{E}[\|\hat{w}_j - w_*\|^2] + N) &\leq \tau \eta_{t-\tau}^2 (2L^2 Z_{t+1-\tau} + N) \\
&\leq 4\tau \frac{\alpha^2}{\mu^2} \frac{1}{(t+E-1)^2} (2L^2 Z_{t+1-\tau} + N),
\end{aligned}$$

where the last inequality follows from (3.39), and

$$\sum_{j=i-\tau}^{i-1} \mathbb{E}[\|\hat{w}_j - w_*\|^2] \leq \tau Z_{i-\tau}.$$

From these inequalities we infer

$$\begin{aligned}\mathbb{E}[\|\hat{w}_{t+1}-w_*\|^2] &\leq 8(1+\sqrt{\Delta}\tau)\tau D\frac{\alpha^2}{\mu^2}\frac{1}{(t+E-1)^2}(2L^2Z_{t+1-\tau}+N)+\\ &\quad 2\frac{\alpha^2 D}{\mu^2}\frac{1}{(t+E-1)^2}\left(\sum_{i=1}^{t}\Big[4a_i(1+\sqrt{\Delta}\tau)[N\tau+2L^2\tau Z_{i-\tau}]+2N\Big]\right)+\\ &\quad \frac{(E+1)^2}{(t+E-1)^2}\mathbb{E}[\|w_0-w_*\|^2]. \end{aligned} \tag{3.42}$$

Even if we assume a constant $Z \geq Z_0 \geq Z_1 \geq Z_2 \geq \ldots$, we can get a first bound on the convergence rate of vectors $\hat{w}^t$: Substituting Z gives

$$\begin{aligned}\mathbb{E}[\|\hat{w}_{t+1}-w_*\|^2] &\leq 8(1+\sqrt{\Delta}\tau)\tau D\frac{\alpha^2}{\mu^2}\frac{1}{(t+E-1)^2}(2L^2Z+N)+\\ &\quad 2\frac{\alpha^2 D}{\mu^2}\frac{1}{(t+E-1)^2}\left(\sum_{i=1}^{t}\Big[4a_i(1+\sqrt{\Delta}\tau)[N\tau+2L^2\tau Z]+2N\Big]\right)+\\ &\quad \frac{(E+1)^2}{(t+E-1)^2}\mathbb{E}[\|w_0-w_*\|^2]. \end{aligned} \tag{3.43}$$

Since $a_i = (L+\mu)\eta_i + 2L^2\eta_i^2 D$ and $\eta_i \leq \frac{\alpha}{\mu(i+E)}$, we have

$$\begin{aligned}\sum_{i=1}^{t} a_i &= (L+\mu)\sum_{i=1}^{t}\eta_i + 2L^2 D\sum_{i=1}^{t}\eta_i^2\\ &\leq (L+\mu)\sum_{i=1}^{t}\frac{\alpha}{\mu(i+E)} + 2L^2 D\sum_{i=1}^{t}\frac{\alpha^2}{\mu^2(i+E)^2}\\ &\leq \frac{(L+\mu)\alpha}{\mu}\sum_{i=1}^{t}\frac{1}{i} + \frac{2L^2\alpha^2 D}{\mu^2}\sum_{i=1}^{t}\frac{1}{i^2}\\ &\leq \frac{(L+\mu)\alpha}{\mu}(1+\ln t) + \frac{L^2\alpha^2 D\pi^2}{3\mu^2}, \end{aligned} \tag{3.44}$$

where the last inequality is a property of the harmonic sequence $\sum_{i=1}^{t}\frac{1}{i} \leq 1+\ln t$ and $\sum_{i=1}^{t}\frac{1}{i^2} \leq \sum_{i=1}^{\infty}\frac{1}{i^2} = \frac{\pi^2}{6}$.

Substituting (3.44) in (3.43) and collecting terms yields

$$\begin{aligned}&\mathbb{E}[\|\hat{w}_{t+1}-w_*\|^2]\\ \leq\ & \frac{2\alpha^2 D}{\mu^2(t+E-1)^2}\left(2Nt + 4(1+\sqrt{\Delta}\tau)\tau[N+2L^2Z]\left\{\frac{(L+\mu)\alpha}{\mu}(1+\ln t) + \frac{L^2\alpha^2 D\pi^2}{3\mu^2+1}\right\}\right)\end{aligned}$$

$$+\frac{(E+1)^2}{(t+E-1)^2}\mathbb{E}[\|w_0 - w_*\|^2]. \tag{3.45}$$

Notice that the asymptotic behavior in t is dominated by the term

$$\frac{4\alpha^2 DN}{\mu^2}\frac{t}{(t+E-1)^2}.$$

If we define Z_{t+1} to be the right hand side of (3.45) and observe that this Z_{t+1} is decreasing and a constant Z exists (since the terms with Z decrease much faster in t compared to the dominating term), then this Z_{t+1} satisfies the derivations done above and a proof by induction can be completed.

Our derivations prove our main result: The expected convergence rate of read vectors is

$$\mathbb{E}[\|\hat{w}_{t+1} - w_*\|^2] \quad \leq \quad \frac{4\alpha^2 DN}{\mu^2}\frac{t}{(t+E-1)^2} + O\left(\frac{\ln t}{(t+E-1)^2}\right).$$

We can use this result in Lemma 3.4.5 in order to show that the expected convergence rate $\mathbb{E}[\|w_{t+1} - w_*\|^2]$ satisfies the same bound.

We remind the reader, that in the $(t+1)$-th iteration at most $\leq \lceil |D_{\xi_t}|/D \rceil$ vector positions are updated. Therefore the expected number of single vector entry updates is at most $\bar{\Delta}_D/D$.

Theorem 3.3.1. *Suppose Assumptions 3.1.1, 3.2.1, 3.2.2 and 3.3.1 and consider Algorithm 3. Let* $\eta_t = \frac{\alpha_t}{\mu(t+E)}$ *with* $4 \leq \alpha_t \leq \alpha$ *and* $E = \max\{2\tau, \frac{4L\alpha D}{\mu}\}$. *Then,* $t' = t\bar{\Delta}_D/D$ *is the expected number of single vector entry updates after* t *iterations and expectations* $\mathbb{E}[\|\hat{w}_t - w_*\|^2]$ *and* $\mathbb{E}[\|w_t - w_*\|^2]$ *are at most*

$$\frac{4\alpha^2 DN}{\mu^2}\frac{t}{(t+E-1)^2} + O\left(\frac{\ln t}{(t+E-1)^2}\right).$$

3.4.3 Convergence without Convexity of Component Functions

For the non-convex case, L in (3.31) must be replaced by $L\kappa$ and as a result L^2 in Lemma 3.4.2 must be replaced by $L^2\kappa$. Also L in (3.35) must be replaced by $L\kappa$. We now require

that $\eta_t \leq \frac{1}{4L\kappa D}$ so that $-2\eta_t(1 - 4L\kappa\eta_t D)[F(w_t) - F(w_*)] \leq 0$. This leads to Lemma 3.4.3 where no changes are needed except requiring $\eta_t \leq \frac{1}{4L\kappa D}$. The changes in Lemmas 3.4.2 and 3.4.3 lead to a Lemma 3.4.5 where we require $E \geq \frac{4L\kappa\alpha D}{\mu}$ and where in the bound of the expectation L^2 must be replaced by $L^2\kappa$. This perculates through to inequality (3.45) with a similar change finally leading to Theorem 3.3.2, i.e., Theorem 3.3.1 where we only need to strengthen the condition on E to $E \geq \frac{4L\kappa\alpha D}{\mu}$ in order to remove Assumption 3.2.2.

3.4.4 Sensitivity to τ

What about the upper bound's sensitivity with respect to τ? Suppose τ is not a constant but an increasing function of t, which also makes E a function of t:

$$\frac{2L\alpha D}{\mu} \leq \tau(t) \leq t \text{ and } E(t) = 2\tau(t).$$

In order to obtain a similar theorem we increase the lower bound on α_t to

$$12 \leq \alpha_t \leq \alpha.$$

This allows us to modify the proof of Lemma 3.4.5 where we analyse the product

$$\prod_{j=i}^{t} \left(1 - \frac{\mu\eta_j}{2}\right).$$

Since $\alpha_j \geq 12$ and $E(j) = 2\tau(j) \leq 2j$,

$$1 - \frac{\mu\eta_j}{2} = 1 - \frac{\alpha_j}{2(j + E(j))} \leq 1 - \frac{12}{2(j + 2j)} = 1 - \frac{2}{j} \leq 1 - \frac{2}{j+1}.$$

The remaining part of the proof of Lemma 3.4.5 continues as before where constant E in the proof is replaced by 1. This yields instead of (3.40)

$$\mathbb{E}[\|w_{t+1} - w_*\|^2] \leq \left(\sum_{i=1}^{t} \left[\frac{(i+1)^2}{t^2}\right] \gamma_i\right) + \frac{4}{t^2}\mathbb{E}[\|w_0 - w_*\|^2].$$

We again substitute $\eta_i \leq \frac{\alpha}{\mu(i+E(i))}$ in γ_i, realize that $\frac{(i+1)}{(i+E(i))} \leq 1$, and compute

$$\begin{aligned}\frac{(i+1)^2}{t^2}\gamma_i &= \frac{(i+1)^2}{t^2}4a_i(1+\sqrt{\Delta}\tau(i))D\frac{\alpha^2}{\mu^2}\frac{1}{(i+E(i))^2}[N\tau(i)+2L^2\sum_{j=i-\tau(i)}^{i-1}\mathbb{E}[\|\hat{w}_j-w_*\|^2] \\ &\quad + \frac{(i+1)^2}{t^2}2ND\frac{\alpha^2}{\mu^2(i+E(i))^2} \\ &\leq \frac{\alpha^2 D}{\mu^2}\frac{1}{t^2}\left[4a_i(1+\sqrt{\Delta}\tau(i))[N\tau(i)+2L^2\sum_{j=i-\tau(i)}^{i-1}\mathbb{E}[\|\hat{w}_j-w_*\|^2]+2N\right].\end{aligned}$$

This gives a new Lemma 3.4.5:

Lemma 3.4.6. *Assume $\frac{2L\alpha D}{\mu} \leq \tau(t) \leq t$ with $\tau(t)$ monotonic increasing. Let $\eta_t = \frac{\alpha_t}{\mu(t+E(t))}$ with $12 \leq \alpha_t \leq \alpha$ and $E(t) = 2\tau(t)$. Then, expectation $\mathbb{E}[\|w_{t+1}-w_*\|^2]$ is at most*

$$\begin{aligned}&\frac{\alpha^2 D}{\mu^2}\frac{1}{t^2}\left(\sum_{i=1}^{t}\left[4a_i(1+\sqrt{\Delta}\tau(i))[N\tau(i)+2L^2\sum_{j=i-\tau(i)}^{i-1}\mathbb{E}[\|\hat{w}_j-w_*\|^2]+2N\right]\right) \\ &\quad + \frac{4}{t^2}\mathbb{E}[\|w_0-w_*\|^2],\end{aligned}$$

where $a_i = (L+\mu)\eta_i + 2L^2\eta_i^2 D$.

Now we can continue the same analysis that led to Theorem 3.3.1 and conclude that there exists a constant Z such that, see (3.43),

$$\begin{aligned}\mathbb{E}[\|\hat{w}_{t+1}-w_*\|^2] &\leq 8(1+\sqrt{\Delta}\tau(t))\tau(t)D\frac{\alpha^2}{\mu^2}\frac{1}{t^2}(2L^2Z+N)+ \\ &\quad 2\frac{\alpha^2 D}{\mu^2}\frac{1}{t^2}\left(\sum_{i=1}^{t}\Big[4a_i(1+\sqrt{\Delta}\tau(i))[N\tau(i)+2L^2\tau(i)Z]+2N\Big]\right)+ \\ &\quad \frac{4}{t^2}\mathbb{E}[\|w_0-w_*\|^2]. \qquad (3.46)\end{aligned}$$

Let us assume

$$\tau(t) \leq \sqrt{t \cdot L(t)}, \qquad (3.47)$$

where

$$L(t) = \frac{1}{\ln t} - \frac{1}{(\ln t)^2}$$

which has the property that the derivative of $t/(\ln t)$ is equal to $L(t)$. Now we observe

$$\begin{aligned}\sum_{i=1}^{t} a_i \tau(i)^2 &= \sum_{i=1}^{t}[(L+\mu)\eta_i + 2L^2\eta_i^2 D]\tau(i)^2 \leq \sum_{i=1}^{t}[(L+\mu)\frac{\alpha}{\mu i} + 2L^2\frac{\alpha^2}{\mu^2 i^2}D]\cdot iL(i) \\ &= \frac{(L+\mu)\alpha}{\mu}\sum_{i=1}^{t} L(i) + O(\ln t) = \frac{(L+\mu)\alpha}{\mu}\frac{t}{\ln t} + O(\ln t)\end{aligned}$$

and

$$\begin{aligned}\sum_{i=1}^{t} a_i \tau(i) &= \sum_{i=1}^{t}[(L+\mu)\eta_i + 2L^2\eta_i^2 D]\tau(i) \leq \sum_{i=1}^{t}[(L+\mu)\frac{\alpha}{\mu i} + 2L^2\frac{\alpha^2}{\mu^2 i^2}D]\cdot\sqrt{i}) \\ &= O(\sum_{i=1}^{t}\frac{1}{\sqrt{i}}) = O(\sqrt{t}).\end{aligned}$$

Substituting both inequalities in (3.46) gives

$$\begin{aligned}&\mathbb{E}[\|\hat{w}_{t+1} - w_*\|^2] \\ \leq\ & 8(1+\sqrt{\Delta}\tau(t))\tau(t)D\frac{\alpha^2}{\mu^2}\frac{1}{t^2}(2L^2 Z + N) + \\ & 2\frac{\alpha^2 D}{\mu^2}\frac{1}{t^2}\left(2Nt + 4\sqrt{\Delta}[\frac{(L+\mu)\alpha}{\mu}\frac{t}{\ln t} + O(\ln t)][N + 2L^2 Z] + O(\sqrt{t})\right) + \\ & \frac{4}{t^2}\mathbb{E}[\|w_0 - w_*\|^2] \\ \leq\ & 2\frac{\alpha^2 D}{\mu^2}\frac{1}{t^2}\left(2Nt + 4\sqrt{\Delta}[(1 + \frac{(L+\mu)\alpha}{\mu})\frac{t}{\ln t} + O(\ln t)][N + 2L^2 Z] + O(\sqrt{t})\right) + \\ & \frac{4}{t^2}\mathbb{E}[\|w_0 - w_*\|^2] \qquad (3.48)\end{aligned}$$

Again we define Z_{t+1} as the right hand side of this inequality. Notice that $Z_t = O(1/t)$, since the above derivation proves

$$\mathbb{E}[\|\hat{w}_{t+1} - w_*\|^2] \leq \frac{4\alpha^2 DN}{\mu^2}\frac{1}{t} + O(\frac{1}{t\ln t}).$$

Summarizing we have the following main lemma:

Lemma 3.4.7. *Let Assumptions 3.1.1, 3.2.1, 3.2.2 and 3.3.1 hold and consider Algorithm 3. Assume* $\frac{2L\alpha D}{\mu} \leq \tau(t) \leq \sqrt{t \cdot L(t)}$ *with* $\tau(t)$ *monotonic increasing. Let* $\eta_t =$

$\frac{\alpha_t}{\mu(t+2\tau(t))}$ *with* $12 \leq \alpha_t \leq \alpha$. *Then, the expected convergence rate of read vectors is*

$$\mathbb{E}[\|\hat{w}_{t+1} - w_*\|^2] \quad \leq \quad \frac{4\alpha^2 DN}{\mu^2}\frac{1}{t} + O(\frac{1}{t \ln t}),$$

where $L(t) = \frac{1}{\ln t} - \frac{1}{(\ln t)^2}$. *The expected convergence rate* $\mathbb{E}[\|w_{t+1} - w_*\|^2]$ *satisfies the same bound.*

Notice that we can plug $Z_t = O(1/t)$ back into an equivalent of (3.42) where we may bound $Z_{i-\tau(i)} = O(1/(i - \tau(i))$ which replaces Z in the second line of (3.43). On careful examination this leads to a new upper bound (3.48) where the $2L^2 Z$ terms gets absorped in a higher order term. This can be used to show that, for

$$t \geq T_0 = \exp[2\sqrt{\Delta}(1 + \frac{(L+\mu)\alpha}{\mu})],$$

the higher order terms that contain $\tau(t)$ (as defined above) are at most the leading term as given in Lemma 3.4.7.

Upper bound (3.48) also shows that, for

$$t \geq T_1 = \frac{\mu^2}{\alpha^2 ND}\|w_0 - w_*\|^2,$$

the higher order term that contains $\|w_0 - w_*\|^2$ is at most the leading term.

3.5 Numerical Experiments

For our numerical experiments, we consider the finite sum minimization problem in (3.2). We consider ℓ_2-regularized logistic regression problems with

$$f_i(w) = \log(1 + \exp(-y_i\langle x_i, w\rangle)) + \frac{\lambda}{2}\|w\|^2,$$

where the penalty parameter λ is set to $1/n$, a widely-used value in literature [34].

We conducted experiments on a single core for Algorithm 3 on two popular datasets `ijcnn1` ($n = 91,701$ training data) and `covtype` ($n = 406,709$ training data) from the

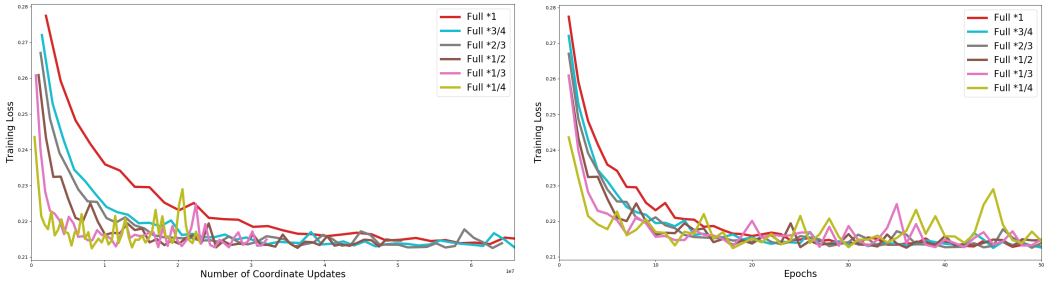


Figure 3.1: *ijcnn1* for different fraction of non-zero set

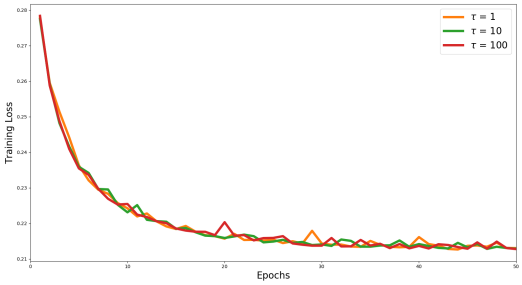


Figure 3.2: *ijcnn1* for different τ with the whole non-zero set

LIBSVM[2] website. Since we are interested in the expected convergence rate with respect to the number of iterations, respectively number of single position vector updates, we do not need a parallelized multi-core simulation to confirm our analysis. The impact of efficient resource scheduling over multiple cores leads to a performance improvement complementary to our analysis of (3.19) (which, as discussed, lends itself for an efficient parallelized implementation). We experimented with 10 runs and reported the average results. We choose the step size based on Theorem 3.3.1, i.e, $\eta_t = \frac{4}{\mu(t+E)}$ and $E = \max\{2\tau, \frac{16LD}{\mu}\}$. For each fraction $v \in \{1, 3/4, 2/3, 1/2, 1/3, 1/4\}$ we performed the following experiment: In Algorithm 3 we choose each "filter" matrix $S_{u_t}^{\xi_t}$ to correspond with a random subset of size $v|D_{\xi_t}|$ of the non-zero positions of D_{ξ_t} (i.e., the support of the gradient corresponding to ξ_t). In addition we use $\tau = 10$. For the two datasets, Figures 3.1 and 3.3 plot the training loss for each fraction with $\tau = 10$. The top plots have t', the number of coordinate updates, for the horizontal axis. The bottom plots have the number of epochs, each epoch counting n iterations, for the horizontal axis. The results show that each fraction shows a sublinear expected convergence rate of $O(1/t')$; the smaller fractions exhibit larger deviations but do seem to converge faster to the minimum solution.

In Figures 3.2 and 3.4, we show experiments with different values of $\tau \in \{1, 10, 100\}$

[2]http://www.csie.ntu.edu.tw/∼cjlin/libsvmtools/datasets/

where we use the whole non-zero set of gradient positions (i.e., $v = 1$) for the update. Our analysis states that, for $t = 50$ epochs times n iterations per epoch, τ can be as large as $\sqrt{t \cdot L(t)} = 524$ for `ijcnn1` and 1058 for `covtype`. The experiments indeed show that $\tau \leq 100$ has little effect on the expected convergence rate.

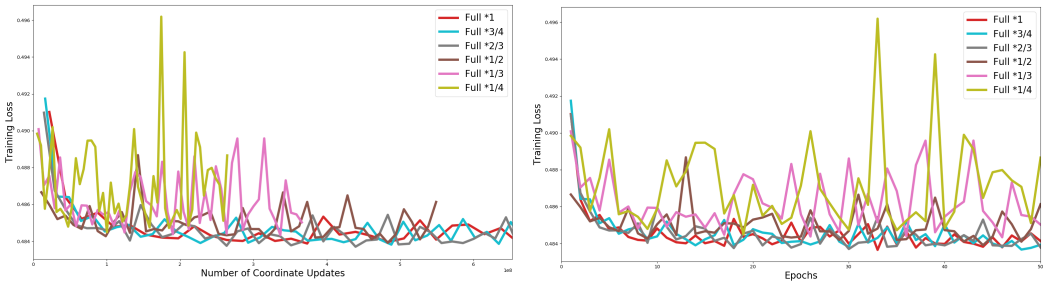


Figure 3.3: *covtype* for different fraction of non-zero set

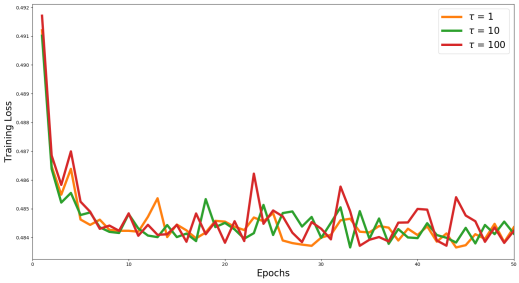


Figure 3.4: *covtype* for different τ with the whole non-zero set

3.6 Conclusion

We have provided the analysis of stochastic gradient algorithms with a diminishing step size in the strongly convex case under the condition of Lipschitz continuity of the individual function realizations, but without requiring any bounds on the stochastic gradients. We showed almost sure convergence of SGD and provided sublinear upper bounds for the expected convergence rate of a general recursion which includes Hogwild! for inconsistent reads and writes as a special case. We also provided new intuition which will help understanding convergence as observed in practice.

Part III

SARAH Algorithm

Chapter 4

SARAH for Convex Optimization

In this chapter, we propose the SARAH algorithm, as well as its practical variant SARAH+, as a novel approach to the finite-sum minimization problems. Different from the vanilla SGD [1] and other modern stochastic methods such as SVRG, S2GD, SAG and SAGA, SARAH admits a simple recursive framework for updating stochastic gradient estimates; when comparing to SAG/SAGA, SARAH does not require a storage of past gradients. The linear convergence rate of SARAH is proven under a strong convexity assumption. We also prove a linear convergence rate (in the strongly convex case) for an inner loop of SARAH, a property that SVRG does not possess. Numerical experiments demonstrate the efficiency of our algorithm.

4.1 Introduction

We are interested in solving a problem of the form

$$\min_{w\in\mathbb{R}^d} \left\{ F(w) \stackrel{\text{def}}{=} \frac{1}{n}\sum_{i\in[n]} f_i(w) \right\}, \tag{4.1}$$

where each f_i, $i \in [n] \stackrel{\text{def}}{=} \{1,\ldots,n\}$ is a convex function with a Lipschitz continuous gradient. Throughout the chapter, we assume that there exists an optimal solution w_* of (5.1).

[1] We mark here that even though stochastic gradient is referred to as SG in literature, the term stochastic gradient descent (SGD) has been widely used in many important works of large-scale learning, including SAG/SAGA, SDCA, SVRG and MISO.

Problems of this type arise frequently in supervised learning applications [25]. Given a training set $\{(x_i, y_i)\}_{i=1}^n$ with $x_i \in \mathbb{R}^d, y_i \in \mathbb{R}$, the least squares regression model, for example, is written as (5.1) with $f_i(w) \stackrel{\text{def}}{=} (x_i^T w - y_i)^2 + \frac{\lambda}{2}\|w\|^2$, where $\|\cdot\|$ denotes the ℓ_2-norm. The ℓ_2-regularized logistic regression for binary classification is written with $f_i(w) \stackrel{\text{def}}{=} \log(1 + \exp(-y_i x_i^T w)) + \frac{\lambda}{2}\|w\|^2$ $(y_i \in \{-1, 1\})$.

In recent years, many advanced optimization methods have been developed for problem (5.1). While the objective function is smooth and convex, the traditional optimization methods, such as gradient descent (GD) or Newton method are often impractical for this problem, when n – the number of training samples and hence the number of f_i's – is very large. In particular, GD updates iterates as follows

$$w_{t+1} = w_t - \eta_t \nabla F(w_t), \quad t = 0, 1, 2, \ldots.$$

Under strong convexity assumption on F and with appropriate choice of η_t, GD converges at a linear rate in terms of objective function values $F(w_t)$. However, when n is large, computing $\nabla F(w_t)$ at each iteration can be prohibitive.

As an alternative, stochastic gradient descent (SGD), originating from the seminal work of Robbins and Monro in 1951 [66], has become the method of choice for solving (5.1). At each step, SGD picks an index $i \in [n]$ uniformly at random, and updates the iterate as $w_{t+1} = w_t - \eta_t \nabla f_i(w_t)$, which is up-to n times cheaper than an iteration of a full gradient method. The convergence rate of SGD is slower than that of GD, in particular, it is sublinear in the strongly convex case. The tradeoff, however, is advantageous due to the tremendous per-iteration savings and the fact that low accuracy solutions are sufficient. This trade-off has been thoroughly analyzed in [11]. Unfortunately, in practice SGD method is often too slow and its performance is too sensitive to the variance in the sample gradients $\nabla f_i(w_t)$. Use of mini-batches (averaging multiple sample gradients $\nabla f_i(w_t)$) was used in [70, 14, 77] to reduce the variance and improve convergence rate by constant factors. Using diminishing sequence $\{\eta_t\}$ is used to control the variance [71, 13], but the practical convergence of SGD is known to be very sensitive to the choice of this sequence, which needs to be hand-picked.

Recently, a class of more sophisticated algorithms have emerged, which use the specific

Table 4.1: Comparisons between different algorithms for strongly convex functions. $\kappa = L/\mu$ is the condition number.

Method	Complexity	Fixed Learning Rate	Low Storage Cost
GD	$\mathcal{O}\left(n\kappa \log\left(1/\epsilon\right)\right)$	✔	✔
SGD	$\mathcal{O}\left(1/\epsilon\right)$	✗	✔
SVRG	$\mathcal{O}\left(\left(n+\kappa\right)\log\left(1/\epsilon\right)\right)$	✔	✔
SAG/SAGA	$\mathcal{O}\left(\left(n+\kappa\right)\log\left(1/\epsilon\right)\right)$	✔	✗
SARAH	$\mathcal{O}\left(\left(n+\kappa\right)\log\left(1/\epsilon\right)\right)$	✔	✔

Table 4.2: Comparisons between different algorithms for convex functions.

Method	Complexity
GD	$\mathcal{O}\left(n/\epsilon\right)$
SGD	$\mathcal{O}\left(1/\epsilon^2\right)$
SVRG	$\mathcal{O}\left(n+\left(\sqrt{n}/\epsilon\right)\right)$
SAGA	$\mathcal{O}\left(n+\left(n/\epsilon\right)\right)$
SARAH	$\mathcal{O}\left(\left(n+\left(1/\epsilon\right)\right)\log(1/\epsilon)\right)$
SARAH (one outer loop)	$\mathcal{O}\left(n+\left(1/\epsilon^2\right)\right)$

finite-sum form of (5.1) and combine some deterministic and stochastic aspects to reduce variance of the steps. The examples of these methods are SAG/SAGA [34, 16], SDCA [72], SVRG [28, 79], DIAG [46], MISO [42] and S2GD [32], all of which enjoy faster convergence rate than that of SGD and use a fixed learning rate parameter η. In this chapter we introduce a new method in this category, SARAH, which further improves several aspects of the existing methods. In Table 5.1 we summarize complexity and some other properties of the existing methods and SARAH when applied to strongly convex problems. Even though SVRG seems competitive as SARAH, SARAH does have a practical variant introduced in Section 4.4.

In addition, theoretical results for complexity of the methods or their variants when applied to general convex functions have been derived [68, 16, 65, 6, 3]. In Table 4.2 we summarize the key complexity results, noting that convergence rate is now sublinear.

Our Contributions. In this chapter, we propose a novel algorithm which combines some of the good properties of existing algorithms, such as SAGA and SVRG, while aiming to improve on both of these methods. In particular, our algorithm does not take steps along a stochastic gradient direction, but rather along an accumulated direction using past stochastic gradient information (as in SAGA) and occasional exact gradient information (as

in SVRG). We summarize the key properties of the proposed algorithm below.

- Similarly to SVRG, SARAH's iterations are divided into the outer loop where a full gradient is computed and the inner loop where only stochastic gradient is computed. Unlike the case of SVRG, the steps of the inner loop of SARAH are based on accumulated stochastic information.
- Like SAG/SAGA and SVRG, SARAH has a sublinear rate of convergence for general convex functions, and a linear rate of convergence for strongly convex functions.
- SARAH uses a constant learning rate, whose size is larger than that of SVRG. However, unlike SAG/SAGA but similar to SVRG, SARAH does not require a storage of n past stochastic gradients.
- We also prove a linear convergence rate (in the strongly convex case) for the inner loop of SARAH, the property that SVRG does not possess. We show that the variance of the steps inside the inner loop goes to zero, thus SARAH is theoretically more stable and reliable than SVRG.
- We provide a practical variant of SARAH based on the convergence properties of the inner loop, where the simple stable stopping criterion for the inner loop is used (see Section 4.4 for more details). This variant shows how SARAH can be made more stable than SVRG in practice.

4.2 SARAH Algorithm

Now we are ready to present our SARAH algorithm (Algorithm 4).

The key step of the algorithm is a recursive update of the stochastic gradient estimate *(SARAH update)*

$$v_t = \nabla f_{i_t}(w_t) - \nabla f_{i_t}(w_{t-1}) + v_{t-1}, \tag{4.2}$$

followed by the iterate update:

$$w_{t+1} = w_t - \eta v_t. \tag{4.3}$$

Algorithm 4 SARAH

Parameters: the learning rate $\eta > 0$ and the inner loop size m.
Initialize: $\tilde{w}_0$
Iterate:
for $s = 1, 2, \ldots$ **do**
  $w_0 = \tilde{w}_{s-1}$
  $v_0 = \frac{1}{n}\sum_{i=1}^{n} \nabla f_i(w_0)$
  $w_1 = w_0 - \eta v_0$
  Iterate:
  for $t = 1, \ldots, m-1$ **do**
    Sample i_t uniformly at random from $[n]$
    $v_t = \nabla f_{i_t}(w_t) - \nabla f_{i_t}(w_{t-1}) + v_{t-1}$
    $w_{t+1} = w_t - \eta v_t$
  end for
  Set $\tilde{w}_s = w_t$ with t chosen uniformly at random from $\{0, 1, \ldots, m\}$
end for

For comparison, SVRG update can be written in a similar way as

$$v_t = \nabla f_{i_t}(w_t) - \nabla f_{i_t}(w_0) + v_0. \tag{4.4}$$

Observe that in SVRG, v_t is an unbiased estimator of the gradient, while it is not true for SARAH. Specifically, [2]

$$\mathbb{E}[v_t|\mathcal{F}_t] = \nabla F(w_t) - \nabla F(w_{t-1}) + v_{t-1} \neq \nabla F(w_t), \tag{4.5}$$

where [3] $\mathcal{F}_t = \sigma(w_0, i_1, i_2, \ldots, i_{t-1})$ is the σ-algebra generated by $w_0, i_1, i_2, \ldots, i_{t-1}$; $\mathcal{F}_0 = \mathcal{F}_1 = \sigma(w_0)$. Hence, SARAH is different from SGD and SVRG type of methods, however, the following total expectation holds,

$$\mathbb{E}[v_t] = \mathbb{E}[\nabla F(w_t)],$$

differentiating SARAH from SAG/SAGA.

SARAH is similar to SVRG [28] since they both contain outer loops which require one full gradient evaluation per outer iteration followed by one full gradient descent step with

[2] $\mathbb{E}[\cdot|\mathcal{F}_t] = \mathbb{E}_{i_t}[\cdot]$, which is expectation with respect to the random choice of index i_t (conditioned on $w_0, i_1, i_2, \ldots, i_{t-1}$).

[3] $\mathcal{F}_t$ also contains all the information of $w_0, \ldots, w_t$ as well as $v_0, \ldots, v_{t-1}$.

a given learning rate. The difference lies in the inner loop, where SARAH updates the stochastic step direction v^t recursively by adding and subtracting component gradients to and from the previous v_{t-1} ($t \geq 1$) in (6.4). Each inner iteration evaluates 2 stochastic gradients and hence the total work per outer iteration is $\mathcal{O}(n+m)$ in terms of the number of gradient evaluations. Note that due to its nature, without running the inner loop, i.e., $m = 1$, SARAH reduces to the GD algorithm.

4.3 Theoretical Analysis

To proceed with the analysis of the proposed algorithm, we will make the following common assumptions.

Assumption 4.3.1 (L-smooth)**.** *Each $f_i : \mathbb{R}^d \to \mathbb{R}$, $i \in [n]$, is L-smooth, i.e., there exists a constant $L > 0$ such that*

$$\|\nabla f_i(w) - \nabla f_i(w')\| \leq L\|w - w'\|, \ \forall w, w' \in \mathbb{R}^d.$$

Note that this assumption implies that $F(w) = \frac{1}{n}\sum_{i=1}^n f_i(w)$ is also *L-smooth*. The following strong convexity assumption will be made for the appropriate parts of the analysis, otherwise, it would be dropped.

Assumption 4.3.2a (μ-strongly convex)**.** *The function $P : \mathbb{R}^d \to \mathbb{R}$, is μ-strongly convex, i.e., there exists a constant $\mu > 0$ such that $\forall w, w' \in \mathbb{R}^d$,*

$$F(w) \geq F(w') + \nabla F(w')^T(w - w') + \frac{\mu}{2}\|w - w'\|^2.$$

Another, stronger, assumption of μ-strong convexity for (5.1) will also be imposed when required in our analysis. Note that Assumption 4.3.2b implies Assumption 4.3.2a but not vice versa.

Assumption 4.3.2b. *Each function $f_i : \mathbb{R}^d \to \mathbb{R}$, $i \in [n]$, is strongly convex with $\mu > 0$.*

Under Assumption 4.3.2a, let us define the (unique) optimal solution of (5.1) as w_*,

Then strong convexity of F implies that

$$2\mu[F(w) - F(w_*)] \leq \|\nabla F(w)\|^2, \ \forall w \in \mathbb{R}^d. \tag{4.6}$$

We note here, for future use, that for strongly convex functions of the form (5.1), arising in machine learning applications, the condition number is defined as $\kappa \overset{\text{def}}{=} L/\mu$. Furthermore, we should also notice that Assumption 4.3.2a or 4.3.2b covers a wide range of problems, e.g. l_2-regularized empirical risk minimization problems with convex losses.

Finally, as a special case of the strong convexity of all f_i's with $\mu = 0$, we state the general convexity assumption, which we will use for convergence analysis.

Assumption 4.3.3. *Each function $f_i : \mathbb{R}^d \to \mathbb{R}$, $i \in [n]$, is convex, i.e.,*

$$f_i(w) \geq f_i(w') + \nabla f_i(w')^T(w - w'), \quad \forall i \in [n].$$

Again, we note that Assumption 4.3.2b implies Assumption 4.3.3, but Assumption 4.3.2a does not. Hence in our analysis, depending on the result we aim at, we will require Assumption 4.3.3 to hold by itself, or Assumption 4.3.2a and Assumption 4.3.3 to hold together, or Assumption 4.3.2b to hold by itself. We will always use Assumption 4.3.1.

Our iteration complexity analysis aims to bound the number of outer iterations $\mathcal{T}$ (or total number of stochastic gradient evaluations) which is needed to guarantee that $\|\nabla F(w_{\mathcal{T}})\|^2 \leq \epsilon$. In this case we will say that $w_{\mathcal{T}}$ is an ϵ-accurate solution. However, as is common practice for stochastic gradient algorithms, we aim to obtain the bound on the number of iterations, which is required to guarantee the bound on the expected squared norm of a gradient, i.e.,

$$\mathbb{E}[\|\nabla F(w_{\mathcal{T}})\|^2] \leq \epsilon. \tag{4.7}$$

We recall some useful existing results that are using in the proofs of this chaper.

Lemma 4.3.1 (Theorem 2.1.5 in [49])**.** *Suppose that f is convex and L-smooth. Then, for any w, $w' \in \mathbb{R}^d$,*

$$f(w) \leq f(w') + \nabla f(w')^T(w - w') + \frac{L}{2}\|w - w'\|^2, \tag{4.8}$$

$$f(w) \geq f(w') + \nabla f(w')^T(w - w') + \frac{1}{2L}\|\nabla f(w) - \nabla f(w')\|^2, \tag{4.9}$$

$$(\nabla f(w) - \nabla f(w'))^T(w - w') \geq \frac{1}{L}\|\nabla f(w) - \nabla f(w')\|^2. \tag{4.10}$$

Note that (4.8) does not require the convexity of f.

Lemma 4.3.2 (Theorem 2.1.11 in [49])**.** *Suppose that f is μ-strongly convex and L-smooth. Then, for any w, $w' \in \mathbb{R}^d$,*

$$(\nabla f(w) - \nabla f(w'))^T(w - w') \geq \frac{\mu L}{\mu + L}\|w - w'\|^2 + \frac{1}{\mu + L}\|\nabla f(w) - \nabla f(w')\|^2. \tag{4.11}$$

4.3.1 Linearly Diminishing Step-Size in a Single Inner Loop

The most important property of the SVRG algorithm is the variance reduction of the steps. This property holds as the number of outer iteration grows, but it does not hold, if only the number of inner iterations increases. In other words, if we simply run the inner loop for many iterations (without executing additional outer loops), the variance of the steps does not reduce in the case of SVRG, while it goes to zero in the case of SARAH. To illustrate this effect, let us take a look at Figures 4.1 and 4.2.

In Figure 4.1, we applied one outer loop of SVRG and SARAH to a sum of 5 quadratic functions in a two-dimensional space, where the optimal solution is at the origin, the black lines and black dots indicate the trajectory of each algorithm and the red point indicates the final iterate. Initially, both SVRG and SARAH take steps along stochastic gradient directions towards the optimal solution. However, later iterations of SVRG wander randomly around the origin with large deviation from it, while SARAH follows a much more stable convergent trajectory, with a final iterate falling in a small neighborhood of the optimal solution.

In Figure 4.2, the x-axis denotes the *number of effective passes* which is equivalent to the number of passes through all of the data in the dataset, the cost of each pass being equal to the cost of one full gradient evaluation; and y-axis represents $\|v_t\|^2$. Figure 4.2 shows the evolution of $\|v_t\|^2$ for SARAH, SVRG, SGD+ (SGD with decreasing learning rate) and FISTA (an accelerated version of GD [8]) with $m = 4n$, where the left plot shows the trend

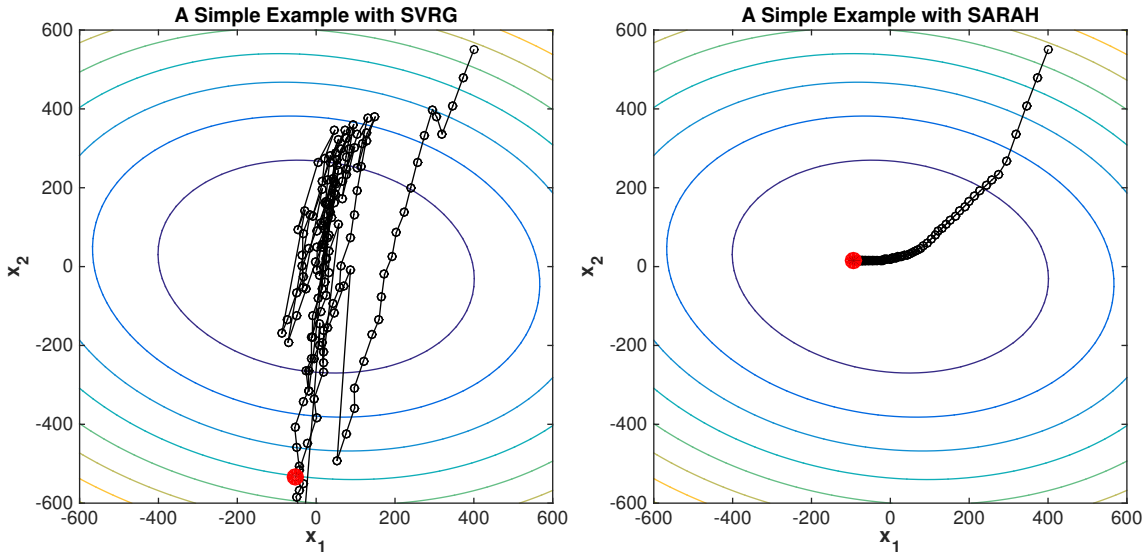


Figure 4.1: A two-dimensional example of $\min_w F(w)$ with $n = 5$ for SVRG (left) and SARAH (right).

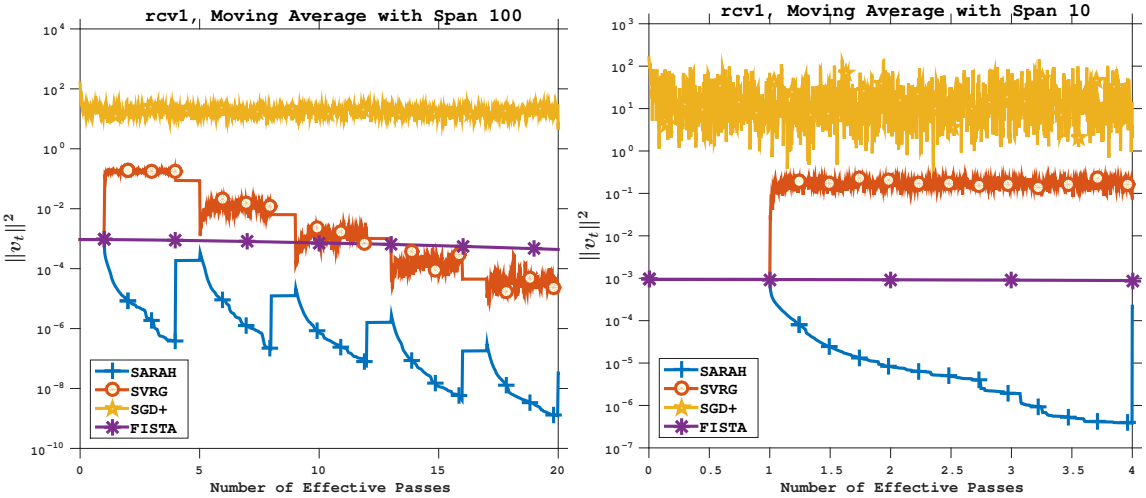


Figure 4.2: An example of ℓ_2-regularized logistic regression on *rcv1* training dataset for SARAH, SVRG, SGD+ and FISTA with multiple outer iterations (left) and a single outer iteration (right).

over multiple outer iterations and the right plot shows a single outer iteration[4]. We can see that for SVRG, $\|v_t\|^2$ decreases over the outer iterations, while it has an increasing trend or oscillating trend for each inner loop. In contrast, SARAH enjoys decreasing trends both in the outer and the inner loop iterations.

We will now show that the stochastic steps computed by SARAH converge linearly in the inner loop. We present two linear convergence results based on our two different assumptions of *μ-strong convexity*. These results substantiate our conclusion that SARAH uses more stable stochastic gradient estimates than SVRG. The following theorem is our first result to demonstrate the linear convergence of our stochastic recursive step v_t.

Theorem 4.3.1a. *Suppose that Assumptions 4.3.1, 4.3.2a and 4.3.3 hold. Consider v_t defined by* (6.4) *in SARAH (Algorithm 4) with $\eta < 2/L$. Then, for any $t \geq 1$,*

$$\begin{aligned}\mathbb{E}[\|v_t\|^2] &\leq \left[1 - \left(\frac{2}{\eta L} - 1\right)\mu^2\eta^2\right]\mathbb{E}[\|v_{t-1}\|^2] \\ &\leq \left[1 - \left(\frac{2}{\eta L} - 1\right)\mu^2\eta^2\right]^t \mathbb{E}[\|\nabla F(w_0)\|^2].\end{aligned}$$

Proof. For $t \geq 1$, we have

$$\begin{aligned}\|\nabla F(w_t) - \nabla F(w_{t-1})\|^2 &= \left\|\frac{1}{n}\sum_{i=1}^{n}[\nabla f_i(w_t) - \nabla f_i(w_{t-1})]\right\|^2 \\ &\leq \frac{1}{n}\sum_{i=1}^{n}\|\nabla f_i(w_t) - \nabla f_i(w_{t-1})\|^2 \\ &= \mathbb{E}[\|\nabla f_{i_t}(w_t) - \nabla f_{i_t}(w_{t-1})\|^2|\mathcal{F}_t]. \qquad (4.12)\end{aligned}$$

Using the proof of Lemma 4.3.5, for $t \geq 1$, we have

$$\begin{aligned}\mathbb{E}[\|v_t\|^2|\mathcal{F}_t] &\leq \|v_{t-1}\|^2 + \left(1 - \frac{2}{\eta L}\right)\mathbb{E}[\|\nabla f_{i_t}(w_{t-1}) - \nabla f_{i_t}(w_t)\|^2|\mathcal{F}_t] \\ &\overset{(6.19)}{\leq} \|v_{t-1}\|^2 + \left(1 - \frac{2}{\eta L}\right)\|\nabla F(w_t) - \nabla F(w_{t-1})\|^2 \\ &\leq \|v_{t-1}\|^2 + \left(1 - \frac{2}{\eta L}\right)\mu^2\eta^2\|v_{t-1}\|^2.\end{aligned}$$

[4]In the plots of Figure 4.2, since the data for SVRG is noisy, we smooth it by using moving average filters with spans 100 for the left plot and 10 for the right one.

Note that $1 - \frac{2}{\eta L} < 0$ since $\eta < 2/L$. The last inequality follows by the strong convexity of F, that is, $\mu\|w_t - w_{t-1}\| \leq \|\nabla F(w_t) - \nabla F(w_{t-1})\|$ and the fact that $w_t = w_{t-1} - \eta v_{t-1}$. By taking the expectation and applying recursively, we have

$$\begin{aligned}
\mathbb{E}[\|v_t\|^2] &\leq \left[1 - \left(\frac{2}{\eta L} - 1\right)\mu^2\eta^2\right]\mathbb{E}[\|v_{t-1}\|^2] \\
&\leq \left[1 - \left(\frac{2}{\eta L} - 1\right)\mu^2\eta^2\right]^t\mathbb{E}[\|v_0\|^2] \\
&= \left[1 - \left(\frac{2}{\eta L} - 1\right)\mu^2\eta^2\right]^t\mathbb{E}[\|\nabla F(w_0)\|^2].
\end{aligned}$$

□

This result implies that by choosing $\eta = \mathcal{O}(1/L)$, we obtain the linear convergence of $\|v_t\|^2$ in expectation with the rate $(1 - 1/\kappa^2)$. Below we show that a better convergence rate can be obtained under a stronger convexity assumption.

Theorem 4.3.1b. *Suppose that Assumptions 4.3.1 and 4.3.2b hold. Consider v_t defined by* (6.4) *in SARAH (Algorithm 4) with $\eta \leq 2/(\mu + L)$. Then the following bound holds,*

$$\mathbb{E}[\|v_t\|^2] \leq \left(1 - \frac{2\mu L\eta}{\mu + L}\right)\mathbb{E}[\|v_{t-1}\|^2], \quad \forall\ t \geq 1,$$

and hence,

$$\mathbb{E}[\|v_t\|^2] \leq \left(1 - \frac{2\mu L\eta}{\mu + L}\right)^t\mathbb{E}[\|\nabla F(w_0)\|^2], \quad \forall\ t \geq 1.$$

Proof. We obviously have $\mathbb{E}[\|v_0\|^2|\mathcal{F}_0] = \|\nabla F(w_0)\|^2$. For $t \geq 1$, we have

$$\begin{aligned}
\mathbb{E}[\|v_t\|^2|\mathcal{F}_t] &\overset{(6.4)}{=} \mathbb{E}[\|v_{t-1} - (\nabla f_{i_t}(w_{t-1}) - \nabla f_{i_t}(w_t))\|^2|\mathcal{F}_t] \\
&\overset{(6.5)}{=} \|v_{t-1}\|^2 + \mathbb{E}[\|\nabla f_{i_t}(w_{t-1}) - \nabla f_{i_t}(w_t)\|^2 \\
&\quad - \frac{2}{\eta}(\nabla f_{i_t}(w_{t-1}) - \nabla f_{i_t}(w_t))^T(w_{t-1} - w_t)|\mathcal{F}_t] \\
&\overset{(6.12)}{\leq} \|v_{t-1}\|^2 - \frac{2\mu L\eta}{\mu + L}\|v_{t-1}\|^2 + \mathbb{E}[\|\nabla f_{i_t}(w_{t-1}) \\
&\quad - \nabla f_{i_t}(w_t)\|^2|\mathcal{F}_t] - \frac{2}{\eta}\cdot\frac{1}{\mu + L}\mathbb{E}[\|\nabla f_{i_t}(w_{t-1}) - \nabla f_{i_t}(w_t)\|^2|\mathcal{F}_t]
\end{aligned}$$

$$= (1 - \frac{2\mu L\eta}{\mu + L})\|v_{t-1}\|^2 + (1 - \frac{2}{\eta} \cdot \frac{1}{\mu + L})\mathbb{E}[\|\nabla f_{i_t}(w_{t-1}) - \nabla f_{i_t}(w_t)\|^2 | \mathcal{F}_t]$$
$$\leq \left(1 - \frac{2\mu L\eta}{\mu + L}\right) \|v_{t-1}\|^2, \tag{4.13}$$

where in last inequality we have used that $\eta \leq 2/(\mu + L)$. By taking the expectation and applying recursively, the desired result is achieved. □

Again, by setting $\eta = \mathcal{O}(1/L)$, we derive the linear convergence with the rate of $(1-1/\kappa)$, which is a significant improvement over the result of Theorem 4.3.1a, when the problem is severely ill-conditioned.

4.3.2 Convergence Analysis

In this section, we derive the general convergence rate results for Algorithm 4. First, we present two important Lemmas as the foundation of our theory. Then, we proceed to provide the sublinear convergence in a single outer iteration for the general convex functions. In the end, we show a competitive linear convergence for the strongly convex case with multiple outer iterations.

We begin with proving two useful lemmas that do not require any convexity assumption. The first Lemma 4.3.3 bounds the sum of expected values of $\|\nabla F(w_t)\|^2$. The second, Lemma 4.3.4, bounds $\mathbb{E}[\|\nabla F(w_t) - v_t\|^2]$.

Lemma 4.3.3. *Suppose that Assumption 4.3.1 holds. Consider SARAH (Algorithm 4). Then, we have*

$$\sum_{t=0}^{m} \mathbb{E}[\|\nabla F(w_t)\|^2] \leq \frac{2}{\eta}\mathbb{E}[F(w_0) - F(w_*)] + \sum_{t=0}^{m} \mathbb{E}[\|\nabla F(w_t) - v_t\|^2] - (1 - L\eta) \sum_{t=0}^{m} \mathbb{E}[\|v_t\|^2]. \tag{4.14}$$

Proof. By Assumption 4.3.1 and $w_{t+1} = w_t - \eta v_t$, we have

$$\mathbb{E}[F(w_{t+1})] \overset{(4.8)}{\leq} \mathbb{E}[F(w_t)] - \eta\mathbb{E}[\nabla F(w_t)^T v_t] + \frac{L\eta^2}{2}\mathbb{E}[\|v_t\|^2]$$
$$= \mathbb{E}[F(w_t)] - \frac{\eta}{2}\mathbb{E}[\|\nabla F(w_t)\|^2] + \frac{\eta}{2}\mathbb{E}[\|\nabla F(w_t) - v_t\|^2] - \left(\frac{\eta}{2} - \frac{L\eta^2}{2}\right)\mathbb{E}[\|v_t\|^2],$$

where the last equality follows from the fact $a^T b = \frac{1}{2}\left[\|a\|^2 + \|b\|^2 - \|a-b\|^2\right]$.

By summing over $t = 0, \ldots, m$, we have

$$\mathbb{E}[F(w_{m+1})] \leq \mathbb{E}[F(w_0)] - \frac{\eta}{2}\sum_{t=0}^{m}\mathbb{E}[\|\nabla F(w_t)\|^2] + \frac{\eta}{2}\sum_{t=0}^{m}\mathbb{E}[\|\nabla F(w_t) - v_t\|^2] - \left(\frac{\eta}{2} - \frac{L\eta^2}{2}\right)\sum_{t=0}^{m}\mathbb{E}[\|v_t\|^2],$$

which is equivalent to ($\eta > 0$):

$$\begin{aligned}\sum_{t=0}^{m}\mathbb{E}[\|\nabla F(w_t)\|^2] &\leq \frac{2}{\eta}\mathbb{E}[F(w_0) - F(w_{m+1})] + \sum_{t=0}^{m}\mathbb{E}[\|\nabla F(w_t) - v_t\|^2] \\ &\quad - (1 - L\eta)\sum_{t=0}^{m}\mathbb{E}[\|v_t\|^2] \\ &\leq \frac{2}{\eta}\mathbb{E}[F(w_0) - F(w_*)] + \sum_{t=0}^{m}\mathbb{E}[\|\nabla F(w_t) - v_t\|^2] - (1 - L\eta)\sum_{t=0}^{m}\mathbb{E}[\|v_t\|^2],\end{aligned}$$

where the last inequality follows since w_* is a global minimizer of F. □

Lemma 4.3.4. *Suppose that Assumption 4.3.1 holds. Consider v_t defined by* (6.4) *in SARAH (Algorithm 4). Then for any $t \geq 1$,*

$$\mathbb{E}[\|\nabla F(w_t) - v_t\|^2] = \sum_{j=1}^{t}\mathbb{E}[\|v_j - v_{j-1}\|^2] - \sum_{j=1}^{t}\mathbb{E}[\|\nabla F(w_j) - \nabla F(w_{j-1})\|^2].$$

Proof. Note that $\mathcal{F}_j$ contains all the information of $w_0, \ldots, w_j$ as well as $v_0, \ldots, v_{j-1}$. For $j \geq 1$, we have

$$\begin{aligned}&\mathbb{E}[\|\nabla F(w_j) - v_j\|^2|\mathcal{F}_j] \\ &= \mathbb{E}[\|[\nabla F(w_{j-1}) - v_{j-1}] + [\nabla F(w_j) - \nabla F(w_{j-1})] - [v_j - v_{j-1}]\|^2|\mathcal{F}_j] \\ &= \|\nabla F(w_{j-1}) - v_{j-1}\|^2 + \|\nabla F(w_j) - \nabla F(w_{j-1})\|^2 + \mathbb{E}[\|v_j - v_{j-1}\|^2|\mathcal{F}_j] \\ &\quad + 2(\nabla F(w_{j-1}) - v_{j-1})^T(\nabla F(w_j) - \nabla F(w_{j-1})) \\ &\quad - 2(\nabla F(w_{j-1}) - v_{j-1})^T\mathbb{E}[v_j - v_{j-1}|\mathcal{F}_j] \\ &\quad - 2(\nabla F(w_j) - \nabla F(w_{j-1}))^T\mathbb{E}[v_j - v_{j-1}|\mathcal{F}_j]\end{aligned}$$

$$= \|\nabla F(w_{j-1}) - v_{j-1}\|^2 - \|\nabla F(w_j) - \nabla F(w_{j-1})\|^2 + \mathbb{E}[\|v_j - v_{j-1}\|^2 | \mathcal{F}_j],$$

where the last equality follows from

$$\mathbb{E}[v_j - v_{j-1} | \mathcal{F}_j] \overset{(6.4)}{=} \mathbb{E}[\nabla f_{i_j}(w_j) - \nabla f_{i_j}(w_{j-1}) | \mathcal{F}_j] = \nabla F(w_j) - \nabla F(w_{j-1}).$$

By taking expectation for the above equation, we have

$$\begin{aligned}
&\mathbb{E}[\|\nabla F(w_j) - v_j\|^2] \\
&= \mathbb{E}[\|\nabla F(w_{j-1}) - v_{j-1}\|^2] - \mathbb{E}[\|\nabla F(w_j) - \nabla F(w_{j-1})\|^2] + \mathbb{E}[\|v_j - v_{j-1}\|^2].
\end{aligned}$$

Note that $\|\nabla F(w_0) - v_0\|^2 = 0$. By summing over $j = 1, \ldots, t$ $(t \geq 1)$, we have

$$\mathbb{E}[\|\nabla F(w_t) - v_t\|^2] = \sum_{j=1}^{t} \mathbb{E}[\|v_j - v_{j-1}\|^2] - \sum_{j=1}^{t} \mathbb{E}[\|\nabla F(w_j) - \nabla F(w_{j-1})\|^2].$$

□

Now we are ready to provide the main theoretical results for SARAH.

General Convex Case

Following from Lemma 4.3.4, we can obtain the following upper bound for $\mathbb{E}[\|\nabla F(w_t) - v_t\|^2]$ for convex functions $f_i, i \in [n]$.

Lemma 4.3.5. *Suppose that Assumptions 4.3.1 and 4.3.3 hold. Consider v_t defined as* (6.4) *in SARAH (Algorithm 4) with $\eta < 2/L$. Then we have that for any $t \geq 1$,*

$$\begin{aligned}
\mathbb{E}[\|\nabla F(w_t) - v_t\|^2] &\leq \frac{\eta L}{2 - \eta L} \Big[\mathbb{E}[\|v_0\|^2] - \mathbb{E}[\|v_t\|^2] \Big] \\
&\leq \frac{\eta L}{2 - \eta L} \mathbb{E}[\|v_0\|^2]. \qquad (4.15)
\end{aligned}$$

Proof. For $j \geq 1$, we have

$$\mathbb{E}[\|v_j\|^2 | \mathcal{F}_j]$$

$$
\begin{aligned}
&= \mathbb{E}[\|v_{j-1} - (\nabla f_{i_j}(w_{j-1}) - \nabla f_{i_j}(w_j))\|^2 | \mathcal{F}_j] \\
&= \|v_{j-1}\|^2 + \mathbb{E}\Big[\|\nabla f_{i_j}(w_{j-1}) - \nabla f_{i_j}(w_j)\|^2 - \frac{2}{\eta}(\nabla f_{i_j}(w_{j-1}) - \nabla f_{i_j}(w_j))^T (w_{j-1} - w_j) | \mathcal{F}_j\Big] \\
&\overset{(4.10)}{\leq} \|v_{j-1}\|^2 + \mathbb{E}\Big[\|\nabla f_{i_j}(w_{j-1}) - \nabla f_{i_j}(w_j)\|^2 - \frac{2}{L\eta}\|\nabla f_{i_j}(w_{j-1}) - \nabla f_{i_j}(w_j)\|^2 | \mathcal{F}_j\Big] \\
&= \|v_{j-1}\|^2 + \left(1 - \frac{2}{\eta L}\right) \mathbb{E}[\|\nabla f_{i_j}(w_{j-1}) - \nabla f_{i_j}(w_j)\|^2 | \mathcal{F}_j] \\
&\overset{(6.4)}{=} \|v_{j-1}\|^2 + \left(1 - \frac{2}{\eta L}\right) \mathbb{E}[\|v_j - v_{j-1}\|^2 | \mathcal{F}_j],
\end{aligned}
$$

which, if we take expectation, implies that

$$
\mathbb{E}[\|v_j - v_{j-1}\|^2] \leq \frac{\eta L}{2 - \eta L}\Big[\mathbb{E}[\|v_{j-1}\|^2] - \mathbb{E}[\|v_j\|^2]\Big],
$$

when $\eta < 2/L$.

By summing the above inequality over $j = 1, \ldots, t$ $(t \geq 1)$, we have

$$
\sum_{j=1}^{t} \mathbb{E}[\|v_j - v_{j-1}\|^2] \leq \frac{\eta L}{2 - \eta L}\Big[\mathbb{E}[\|v_0\|^2] - \mathbb{E}[\|v_t\|^2]\Big]. \tag{4.16}
$$

By Lemma 4.3.4, we have

$$
\mathbb{E}[\|\nabla F(w_t) - v_t\|^2] \leq \sum_{j=1}^{t} \mathbb{E}[\|v_j - v_{j-1}\|^2] \overset{(6.23)}{\leq} \frac{\eta L}{2 - \eta L}\Big[\mathbb{E}[\|v_0\|^2] - \mathbb{E}[\|v_t\|^2]\Big].
$$

□

Using the above lemmas, we can state and prove one of our core theorems as follows.

Theorem 4.3.2. *Suppose that Assumptions 4.3.1 and 4.3.3 hold. Consider SARAH (Algorithm 4) with $\eta \leq 1/L$. Then for any $s \geq 1$, we have*

$$
\mathbb{E}[\|\nabla F(\tilde{w}_s)\|^2] \leq \frac{2}{\eta(m+1)} \mathbb{E}[P(\tilde{w}_{s-1}) - F(w_*)] + \frac{\eta L}{2 - \eta L} \mathbb{E}[\|\nabla P(\tilde{w}_{s-1})\|^2]. \tag{4.17}
$$

Proof. Since $v_0 = \nabla F(w_0)$ implies $\|\nabla F(w_0) - v_0\|^2 = 0$ then by Lemma 4.3.5, we can write

$$
\sum_{t=0}^{m} \mathbb{E}[\|\nabla F(w_t) - v_t\|^2] \leq \frac{m\eta L}{2 - \eta L} \mathbb{E}[\|v_0\|^2]. \tag{4.18}
$$

Hence, by Lemma 4.3.3 with $\eta \leq 1/L$, we have

$$\sum_{t=0}^{m} \mathbb{E}[\|\nabla F(w_t)\|^2] \leq \frac{2}{\eta}\mathbb{E}[F(w_0) - F(w_*)] + \sum_{t=0}^{m} \mathbb{E}[\|\nabla F(w_t) - v_t\|^2]$$
$$\overset{(6.26)}{\leq} \frac{2}{\eta}\mathbb{E}[F(w_0) - F(w_*)] + \frac{m\eta L}{2 - \eta L}\mathbb{E}[\|v_0\|^2]. \tag{4.19}$$

Since we are considering one outer iteration, with $s \geq 1$, then we have $v_0 = \nabla F(w_0) = \nabla P(\tilde{w}_{s-1})$ (since $w_0 = \tilde{w}_{s-1}$), and $\tilde{w}_s = w_t$, where t is picked uniformly at random from $\{0, 1, \ldots, m\}$. Therefore, the following holds,

$$\mathbb{E}[\|\nabla F(\tilde{w}_s)\|^2] = \frac{1}{m+1}\sum_{t=0}^{m} \mathbb{E}[\|\nabla F(w_t)\|^2]$$
$$\overset{(6.27)}{\leq} \frac{2}{\eta(m+1)}\mathbb{E}[P(\tilde{w}_{s-1}) - F(w_*)] + \frac{\eta L}{2 - \eta L}\mathbb{E}[\|\nabla P(\tilde{w}_{s-1})\|^2]. \qquad \square$$

Theorem 4.3.2, in the case when $\eta \leq 1/L$ implies that

$$\mathbb{E}[\|\nabla F(\tilde{w}_s)\|^2] \leq \frac{2}{\eta(m+1)}\mathbb{E}[P(\tilde{w}_{s-1}) - F(w_*)] + \eta L\mathbb{E}[\|\nabla P(\tilde{w}_{s-1})\|^2].$$

By choosing the learning rate $\eta = \sqrt{\frac{2}{L(m+1)}}$ (with m such that $\sqrt{\frac{2}{L(m+1)}} \leq 1/L$) we can derive the following convergence result,

$$\mathbb{E}[\|\nabla F(\tilde{w}_s)\|^2] \leq \sqrt{\frac{2L}{m+1}}\mathbb{E}[P(\tilde{w}_{s-1}) - F(w_*) + \|\nabla P(\tilde{w}_{s-1})\|^2].$$

Clearly, this result shows a sublinear convergence rate for SARAH under general convexity assumption within a single inner loop, with increasing m, and consequently, we have the following result for complexity bound.

Corollary 4.3.1. *Suppose that Assumptions 4.3.1 and 4.3.3 hold. Consider SARAH (Algorithm 4) within a single outer iteration with the learning rate $\eta = \sqrt{\frac{2}{L(m+1)}}$ where $m \geq 2L-1$ is the total number of iterations, then $\|\nabla F(w_t)\|^2$ converges sublinearly in expectation with a rate of $\sqrt{\frac{2L}{m+1}}$, and therefore, the total complexity to achieve an ϵ-accurate solution defined*

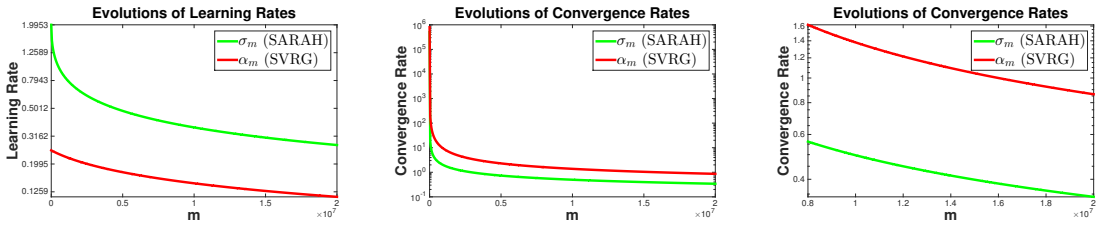


Figure 4.3: Theoretical comparisons of learning rates (left) and convergence rates (middle and right) with $n = 1,000,000$ for SVRG and SARAH in one inner loop.

in (6.6) *is* $\mathcal{O}(n + 1/\epsilon^2)$.

We now turn to estimating convergence of SARAH with multiple outer steps. Simply using Theorem 4.3.2 for each of the outer steps we have the following lemma.

Theorem 4.3.3. *Suppose that Assumptions 4.3.1 and 4.3.3 hold. Consider SARAH (Algorithm 4) and define*

$$\delta_k = \frac{2}{\eta(m+1)}\mathbb{E}[F(\tilde{w}_k) - F(w_*)], \ k = 0, 1, \ldots, s-1,$$

and $\delta = \max_{0\le k\le s-1} \delta_k$. *Then we have*

$$\mathbb{E}[\|\nabla F(\tilde{w}_s)\|^2] - \Delta \le \alpha^s(\|\nabla F(\tilde{w}_0)\|^2 - \Delta), \tag{4.20}$$

where $\Delta = \delta\left(1 + \frac{\eta L}{2(1-\eta L)}\right)$, *and* $\alpha = \frac{\eta L}{2-\eta L}$.

Proof. By Theorem 4.3.2, we have

$$\begin{aligned}
\mathbb{E}[\|\nabla F(\tilde{w}_s)\|^2] &\le \frac{2}{\eta(m+1)}\mathbb{E}[F(\tilde{w}_{s-1}) - F(w_*)] + \frac{\eta L}{2-\eta L}\mathbb{E}[\|\nabla F(\tilde{w}_{s-1})\|^2] \\
&= \delta_{s-1} + \alpha\mathbb{E}[\|\nabla F(\tilde{w}_{s-1})\|^2] \\
&\le \delta_{s-1} + \alpha\delta_{s-2} + \cdots + \alpha^{s-1}\delta_0 + \alpha^s\|\nabla F(\tilde{w}_0)\|^2 \\
&\le \delta + \alpha\delta + \cdots + \alpha^{s-1}\delta + \alpha^s\|\nabla F(\tilde{w}_0)\|^2 \\
&\le \delta\frac{1-\alpha^s}{1-\alpha} + \alpha^s\|\nabla F(\tilde{w}_0)\|^2 \\
&= \Delta(1-\alpha^s) + \alpha^s\|\nabla F(\tilde{w}_0)\|^2 \\
&= \Delta + \alpha^s(\|\nabla F(\tilde{w}_0)\|^2 - \Delta),
\end{aligned}$$

where the second last equality follows since

$$\frac{\delta}{1-\alpha} = \delta\left(\frac{2-\eta L}{2-2\eta L}\right) = \delta\left(1 + \frac{\eta L}{2(1-\eta L)}\right) = \Delta.$$

Hence, the desired result is achieved. □

Based on Theorem 4.3.3, we have the following total complexity for SARAH in the general convex case.

Corollary 4.3.2. *Let us choose $\Delta = \epsilon/4$, $\alpha = 1/2$ (with $\eta = 2/(3L)$), and $m = \mathcal{O}(1/\epsilon)$ in Theorem 4.3.3. Then, the total complexity to achieve an ϵ-accuracy solution defined in* (6.6) *is $\mathcal{O}((n + (1/\epsilon))\log(1/\epsilon))$.*

Proof. Based on Theorem 4.3.3, if we would aim for an ϵ-accuracy solution, we can choose $\Delta = \epsilon/4$ and $\alpha = 1/2$ (with $\eta = 2/(3L)$). To obtain the convergence to an ϵ-accuracy solution, we need to have $\delta = \mathcal{O}(\epsilon)$, or equivalently, $m = \mathcal{O}(1/\epsilon)$. Then we have

$$\begin{aligned}
\mathbb{E}[\|\nabla F(\tilde{w}_s)\|^2] &\overset{(4.20)}{\leq} \frac{\Delta}{2} + \frac{1}{2}\mathbb{E}[\|\nabla F(\tilde{w}_{s-1})\|^2] \\
&\leq \frac{\Delta}{2} + \frac{\Delta}{2^2} + \frac{1}{2^2}\mathbb{E}[\|\nabla F(\tilde{w}_{s-2})\|^2] \\
&\leq \Delta\left(\frac{1}{2} + \frac{1}{2^2} + \cdots + \frac{1}{2^s}\right) + \frac{1}{2^s}\|\nabla F(\tilde{w}_0)\|^2 \\
&\leq \Delta + \frac{1}{2^s}\|\nabla F(\tilde{w}_0)\|^2.
\end{aligned}$$

To guarantee that $\mathbb{E}[\|\nabla F(\tilde{w}_s)\|^2] \leq \epsilon$, it is sufficient to make $\frac{1}{2^s}\|\nabla F(\tilde{w}_0)\|^2 \leq \frac{3}{4}\epsilon$, or $s = \mathcal{O}(\log(1/\epsilon))$. This implies the total complexity to achieve an ϵ-accuracy solution is $(n + 2m)s = \mathcal{O}((n + (1/\epsilon))\log(1/\epsilon))$. □

Strongly Convex Case

We now turn to the discussion of the linear convergence rate of SARAH under the strong convexity assumption on F. From Theorem 4.3.2, for any $s \geq 1$, using property (6.8) of the *μ-strongly convex F*, we have

$$\mathbb{E}[\|\nabla F(\tilde{w}_s)\|^2] \leq \frac{2}{\eta(m+1)}\mathbb{E}[P(\tilde{w}_{s-1}) - F(w_*)] + \frac{\eta L}{2-\eta L}\mathbb{E}[\|\nabla P(\tilde{w}_{s-1})\|^2]$$

$$\overset{(6.8)}{\leq} \left(\frac{1}{\mu\eta(m+1)} + \frac{\eta L}{2-\eta L}\right) \mathbb{E}[\|\nabla P(\tilde{w}_{s-1})\|^2],$$

and equivalently,

$$\mathbb{E}[\|\nabla F(\tilde{w}_s)\|^2] \leq \sigma_m \, \mathbb{E}[\|\nabla P(\tilde{w}_{s-1})\|^2]. \tag{4.21}$$

If we define

$$\sigma_m \overset{\text{def}}{=} \frac{1}{\mu\eta(m+1)} + \frac{\eta L}{2-\eta L}. \tag{4.22}$$

Then by choosing η and m such that $\sigma_m < 1$, and applying (4.21) recursively, we are able to reach the following convergence result.

Theorem 4.3.4. *Suppose that Assumptions 4.3.1, 4.3.2a and 4.3.3 hold. Consider SARAH (Algorithm 4) with the choice of η and m such that*

$$\sigma_m \overset{def}{=} \frac{1}{\mu\eta(m+1)} + \frac{\eta L}{2-\eta L} < 1. \tag{4.23}$$

Then, we have

$$\mathbb{E}[\|\nabla F(\tilde{w}_s)\|^2] \leq (\sigma_m)^s \|\nabla F(\tilde{w}_0)\|^2.$$

Remark 4.3.1. *Theorem 4.3.4 implies that any $\eta < 1/L$ will work for SARAH. Let us compare our convergence rate to that of SVRG. The linear rate of SVRG, as presented in [28], is given by*

$$\alpha_m = \frac{1}{\mu\eta(1-2L\eta)m} + \frac{2\eta L}{1-2\eta L} < 1.$$

We observe that it implies that the learning rate has to satisfy $\eta < 1/(4L)$, which is a tighter restriction than $\eta < 1/L$ required by SARAH. In addition, with the same values of m and η, the rate or convergence of (the outer iterations) of SARAH is always smaller than that of SVRG.

$$\sigma_m = \frac{1}{\mu\eta(m+1)} + \frac{\eta L}{2-\eta L} = \frac{1}{\mu\eta(m+1)} + \frac{1}{2/(\eta L) - 1}$$

$$< \frac{1}{\mu\eta(1-2L\eta)m} + \frac{1}{0.5/(\eta L)-1} = \alpha_m.$$

Remark 4.3.2. *To further demonstrate the better convergence properties of SARAH, let us consider following optimization problem*

$$\min_{0<\eta<1/L} \sigma_m, \qquad \min_{0<\eta<1/4L} \alpha_m,$$

which can be interpreted as the best convergence rates for different values of m, for both SARAH and SVRG. After simple calculations, we plot both learning rates and the corresponding theoretical rates of convergence, as shown in Figure 4.3, where the right plot is a zoom-in on a part of the middle plot. The left plot shows that the optimal learning rate for SARAH is significantly larger than that of SVRG, while the other two plots show significant improvement upon outer iteration convergence rates for SARAH over SVRG.

Based on Theorem 4.3.4, we are able to derive the following total complexity for SARAH in the strongly convex case.

Corollary 4.3.3. *Fix $\epsilon \in (0,1)$, and let us run SARAH with $\eta = 1/(2L)$ and $m = 4.5\kappa$ for $\mathcal{T}$ iterations where $\mathcal{T} = \lceil \log(\|\nabla F(\tilde{w}_0)\|^2/\epsilon)/\log(9/7) \rceil$, then we can derive an ϵ-accuracy solution defined in* (6.6)*. Furthermore, we can obtain the total complexity of SARAH, to achieve the ϵ-accuracy solution, as $\mathcal{O}\left((n+\kappa)\log(1/\epsilon)\right)$.*

Proof. Based on Theorem 4.3.4, let us run SARAH with $\eta = 1/(2L)$ and $m = 4.5\kappa$, then we can calculate σ_m in (4.23) as

$$\sigma_m = \frac{1}{\mu\eta(m+1)} + \frac{\eta L}{2-\eta L} = \frac{1}{[\mu/(2L)](4.5\kappa+1)} + \frac{1/2}{2-1/2} < \frac{4}{9} + \frac{1}{3} = \frac{7}{9}.$$

According to Theorem 4.3.4, if we run SARAH for $\mathcal{T}$ iterations where

$$\mathcal{T} = \lceil \log(\|\nabla F(\tilde{w}_0)\|^2/\epsilon)/\log(9/7) \rceil = \lceil \log_{7/9}(\epsilon/\|\nabla F(\tilde{w}_0)\|^2) \rceil \geq \log_{7/9}(\epsilon/\|\nabla F(\tilde{w}_0)\|^2),$$

then we have

$$\mathbb{E}[\|\nabla F(\tilde{w}_{\mathcal{T}})\|^2] \leq (\sigma_m)^{\mathcal{T}} \|\nabla F(\tilde{w}_0)\|^2 < (7/9)^{\mathcal{T}} \|\nabla F(\tilde{w}_0)\|^2$$
$$\leq (7/9)^{\log_{7/9}(\epsilon/\|\nabla F(\tilde{w}_0)\|^2)} \|\nabla F(\tilde{w}_0)\|^2 = \epsilon,$$

thus we can derive (6.6). If we consider the number of gradient evaluations as the main computational complexity, then the total complexity can be obtained as

$$(n + 2m)\mathcal{T} = \mathcal{O}\left((n + \kappa)\log(1/\epsilon)\right).$$

□

4.4 A Practical Variant

While SVRG is an efficient variance-reducing stochastic gradient method, one of its main drawbacks is the sensitivity of the practical performance with respect to the choice of m. It is know that m should be around $\mathcal{O}(\kappa)$,[5] while it still remains unknown that what the exact best choice is. In this section, we propose a practical variant of SARAH as SARAH+ (Algorithm 5), which provides an automatic and adaptive choice of the inner loop size m. Guided by the linear convergence of the steps in the inner loop, demonstrated in Figure 4.2, we introduce a stopping criterion based on the values of $\|v_t\|^2$ while upper-bounding the total number of steps by a large enough m for robustness. The other modification compared to Algorithm 4 is the more practical choice $\tilde{w}_s = w_t$, where t is the last index of the particular inner loop, instead of randomly selected intermediate index.

Different from SARAH, SARAH+ provides a possibility of earlier termination and unnecessary careful choices of m, and it also covers the classical gradient descent when we set $\gamma = 1$ (since the while loop does not proceed). In Figure 4.4 we present the numerical performance of SARAH+ with different γs on *rcv1* and *news20* datasets. The size of the inner loop provides a trade-off between the fast sub-linear convergence in the inner loop

[5] In practice, when n is large, $F(w)$ is often considered as a regularized Empirical Loss Minimization problem with regularization parameter $\mu = \frac{1}{n}$, then $\kappa \sim \mathcal{O}(n)$.

Algorithm 5 SARAH+

Parameters: the learning rate $\eta > 0$, $0 < \gamma \leq 1$ and the maximum inner loop size m.
Initialize: $\tilde{w}_0$
Iterate:
for $s = 1, 2, \ldots$ **do**
  $w_0 = \tilde{w}_{s-1}$
  $v_0 = \frac{1}{n}\sum_{i=1}^{n} \nabla f_i(w_0)$
  $w_1 = w_0 - \eta v_0$
  $t = 1$
  while $\|v_{t-1}\|^2 > \gamma \|v_0\|^2$ **and** $t < m$ **do**
    Sample i_t uniformly at random from $[n]$
    $v_t = \nabla f_{i_t}(w_t) - \nabla f_{i_t}(w_{t-1}) + v_{t-1}$
    $w_{t+1} = w_t - \eta v_t$
    $t = t + 1$
  end while
  Set $\tilde{w}_s = w_t$
end for

and linear convergence in the outer loop. From the results, it appears that $\gamma = 1/8$ is the optimal choice. With a larger γ, i.e. $\gamma > 1/8$, the iterates in the inner loop do not provide sufficient reduction, before another full gradient computation is required, while with $\gamma < 1/8$ an unnecessary number of inner steps is performed without gaining substantial progress. Clearly γ is another parameter that requires tuning, however, in our experiments, the performance of SARAH+ has been very robust with respect to the choices of γ and did not vary much from one data set to another.

Similar to SVRG, SARAH+ has $\|v_t\|^2$ decreasing in outer iterations. However, SARAH+ also inherits from SARAH the consistent decreasing of $\|v_t\|^2$ in expectation in inner loops. It is rather impossible to apply the similar trick to SVRG, as $\|v_t\|^2$ of SVRG is increasing in each inner loop and the trend is irregular with high fluctuations in later iterations as shown in Figure 4.2.

4.5 Numerical Experiments

To support the theoretical analyses and insights, we present our empirical experiments, comparing SARAH and SARAH+ with the state-of-the-art first-order methods for ℓ_2-

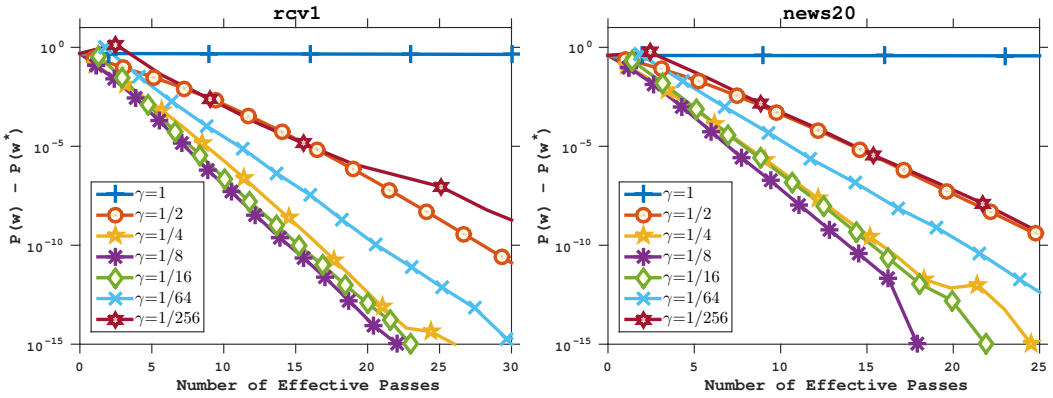


Figure 4.4: An example of ℓ_2-regularized logistic regression on *rcv1* (left) and *news20* (right) training datasets for SARAH+ with different γs on loss residuals $F(w) - F(w_*)$.

Table 4.3: Summary of datasets used for experiments.

Dataset	d	n (train)	Sparsity	n (test)	L
covtype	54	406,709	22.12%	174,303	1.90396
ijcnn1	22	91, 701	59.09%	49, 990	1.77662
news20	1,355,191	13, 997	0.03375%	5, 999	0.2500
rcv1	47,236	677,399	0.1549%	20,242	0.2500

regularized logistic regression problems with

$$f_i(w) = \log(1 + \exp(-y_i x_i^T w)) + \frac{\lambda}{2}\|w\|^2,$$

on datasets *covtype, ijcnn1, news20* and *rcv1* [6]. For *ijcnn1* and *rcv1* we use the predefined testing and training sets, while *covtype* and *news20* do not have test data, hence we randomly split the datasets with 70% for training and 30% for testing. Some statistics of the datasets are summarized in Table 4.3.

The penalty parameter λ is set to $1/n$ as is common practice [34]. Note that like SVRG/S2GD and SAG/SAGA, SARAH also allows an efficient sparse implementation named "lazy updates" [31]. We conduct and compare numerical results of SARAH with SVRG, SAG, SGD+ and FISTA. SVRG [28] and SAG [34] are classic modern stochastic methods. SGD+ is SGD with decreasing learning rate $\eta = \eta_0/(k+1)$ where k is the number of effective passes and η_0 is some initial constant learning rate. FISTA [8] is the Fast Iterative Shrinkage-Thresholding Algorithm, well-known as an efficient accelerated version of the gradient descent. Even though for each method, there is a theoretical safe learning rate, we compare the results for the best learning rates in hindsight.

[6] All datasets are available at http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/.

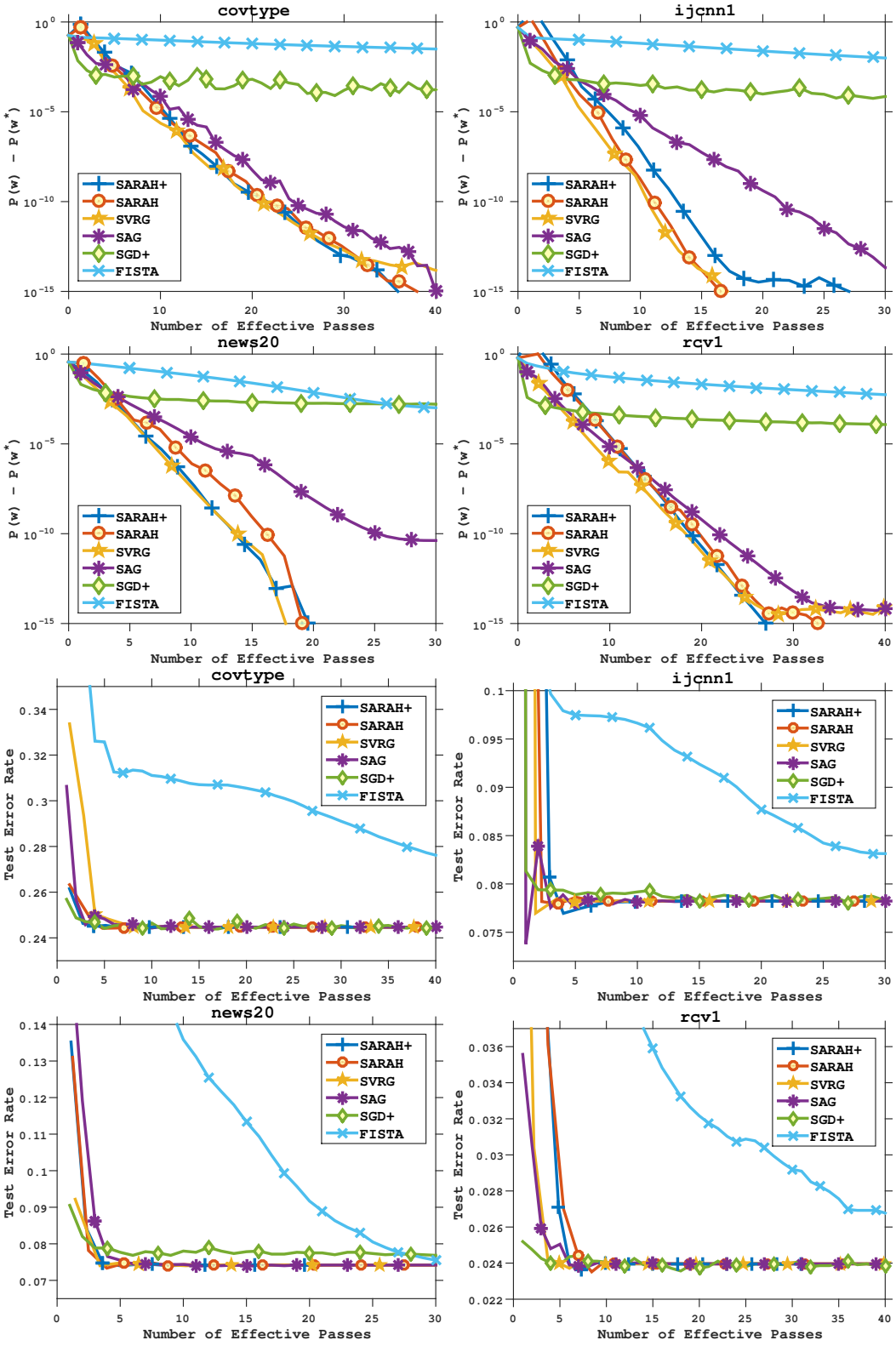


Figure 4.5: Comparisons of loss residuals $F(w)-F(w_*)$ (top) and test errors (bottom) from different modern stochastic methods on *covtype, ijcnn1, news20* and *rcv1*.

Table 4.4: Summary of best parameters for all the algorithms on different datasets.

Dataset	SARAH (m^*, η^*)	SVRG (m^*, η^*)	SAG (η^*)	SGD+ (η^*)	FISTA (η^*)
covtype	(2n, 0.9/L)	(n, 0.8/L)	0.3/L	0.06/L	50/L
ijcnn1	(0.5n, 0.8/L)	(n, 0.5/L)	0.7/L	0.1/L	90/L
news20	(0.5n, 0.9/L)	(n, 0.5/L)	0.1/L	0.2/L	30/L
rcv1	(0.7n, 0.7/L)	(0.5n, 0.9/L)	0.1/L	0.1/L	120/L

Figure 4.5 shows numerical results in terms of loss residuals (top) and test errors (bottom) on the four datasets, SARAH is sometimes comparable or a little worse than other methods at the beginning. However, it quickly catches up to or surpasses all other methods, demonstrating a faster rate of decrease across all experiments. We observe that on *covtype* and *rcv1*, SARAH, SVRG and SAG are comparable with some advantage of SARAH on *covtype*. On *ijcnn1* and *news20*, SARAH and SVRG consistently surpass the other methods.

In particular, to validate the efficiency of our practical variant SARAH+, we provide an insight into how important the choices of m and η are for SVRG and SARAH in Table 5.2 and Figure 4.6. Table 5.2 presents the optimal choices of m and η for each of the algorithm, while Figure 4.6 shows the behaviors of SVRG and SARAH with different choices of m for *covtype* and *ijcnn1*, where m^*s denote the best choices. In Table 5.2, the optimal learning rates of SARAH vary less among different datasets compared to all the other methods and they fall within the learning rate limit of SARAH $(1/L)$; on the contrary, other methods can vary beyond the limits (SVRG with $1/(4L)$, SAG with $1/(16L)$, FISTA with $1/L$) for different datasets ; the empirical studies suggest that it is much easier to tune and find the ideal learning rate for SARAH. As observed in Figure 4.6, the behaviors of both SARAH and SVRG are quite sensitive to the choices of m. With improper choices of m, the loss residuals can be increased considerably from 10^{-15} to 10^{-3} on both *covtype* in 40 effective passes and *ijcnn1* in 17 effective passes for SARAH/SVRG.

4.6 Conclusion

We propose a new variance reducing stochastic recursive gradient algorithm SARAH, which combines some of the properties of well known existing algorithms, such as SAGA and

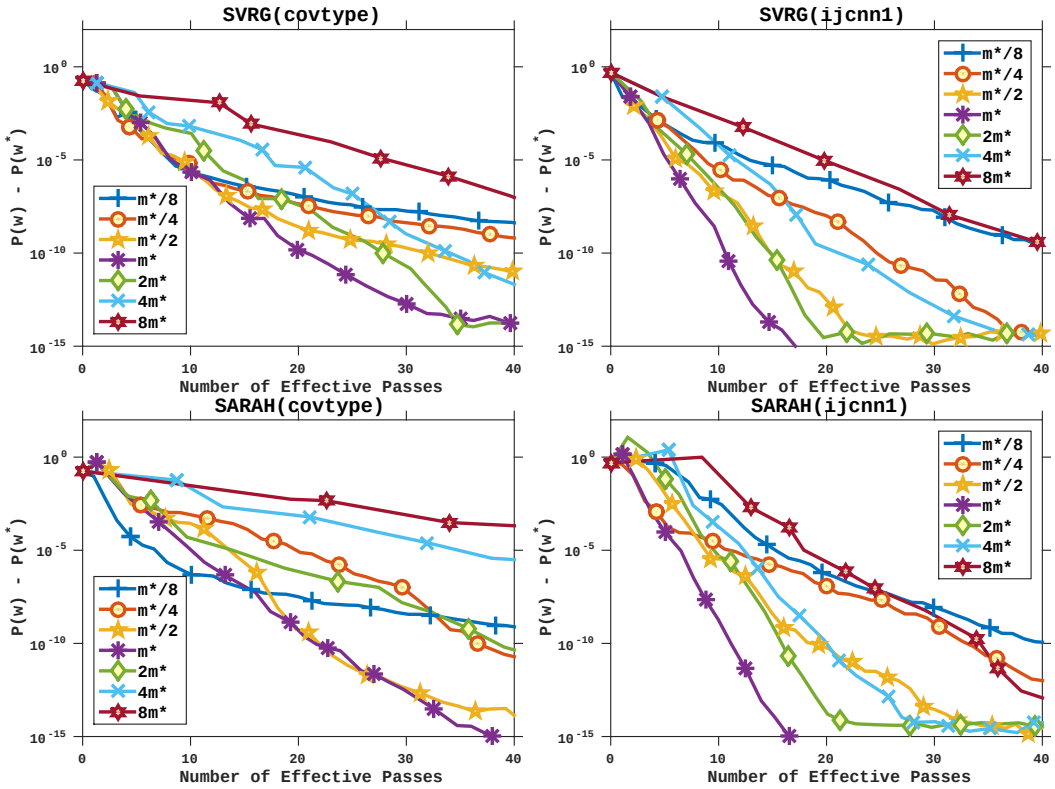


Figure 4.6: Comparisons of loss residuals $F(w) - F(w_*)$ for different inner loop sizes with SVRG (top) and SARAH (bottom) on *covtype* and *ijcnn1*.

SVRG. For smooth convex functions, we show a sublinear convergence rate, while for strongly convex cases, we prove the linear convergence rate and the computational complexity as those of SVRG and SAG. However, compared to SVRG, SARAH's convergence rate constant is smaller and the algorithms is more stable both theoretically and numerically. Additionally, we prove the linear convergence for inner loops of SARAH which support the claim of stability. Based on this convergence we derive a practical version of SARAH, with a simple stopping criterion for the inner loops.

Chapter 5

SARAH for Nonconvex Optimization

In this chapter, we study and analyze a mini-batch version of the SARAH algorithm, a method employing stochastic recursive gradients, for solving empirical loss minimization for the case of nonconvex losses. We provide a sublinear convergence rate (to stationary points) for general nonconvex functions and a linear convergence rate for gradient dominated functions, both of which have some advantages compared to other modern stochastic gradient algorithms for nonconvex losses.

5.1 Introduction

We are interested in the following finite-sum minimization problem

$$\min_{w\in\mathbb{R}^d}\left\{F(w)\stackrel{\text{def}}{=}\frac{1}{n}\sum_{i\in[n]}f_i(w)\right\}, \tag{5.1}$$

where each f_i, $i\in[n]\stackrel{\text{def}}{=}\{1,\ldots,n\}$, is smooth but can be nonconvex, and F is also not necessarily convex. Throughout the chapter, we assume that there exists a global optimal solution w_* of (5.1); in other words, there exists a lower bound $F(w_*)$ of (5.1), however we do not assume the knowledge of this bound and we do not seek convergence to w_*, in general.

Problems of form (5.1) cover a wide range of convex and nonconvex problems including but not limited to logistic regression, multi-kernel learning, conditional random fields, neural networks, etc. In many of these applications, the number n of individual components is very large, which makes the exact computation of $F(w)$ and its derivatives and thus the use of gradient descent (GD) [57] to solve (5.1) expensive.

A traditional approach is to employ stochastic gradient descent (SGD) [66, 71]. Recently, a large number of improved variants of stochastic gradient algorithms have emerged, including SAG/SAGA [68, 16], MISO/FINITO [43, 17], SDCA [72], SVRG/S2GD [28, 31], SARAH [51] [1]. While, nonconvex problems of the form (5.1) are now widely used due to the recent interest in deep neural networks, the majority of methods are designed and analyzed for the convex/strongly convex cases. Limited results have been developed for the nonconvex problems [65, 5, 4], in particular, [65, 5] introduce nonconvex SVRG, and Natasha [4] is a new algorithm but a variant of SVRG for nonconvex optimization.

In this chapter we develop convergence rate analysis of a mini-batch variant SARAH for nonconvex problems of the form (5.1). SARAH has been introduced in [51] and shown to have a sublinear rate of convergence for general convex functions, and a linear rate of convergence for strongly convex functions. As the SVRG method, SARAH has an inner and an outer loop. It has been shown in [51] that, unlike the inner loop of SVRG, the inner loop of SARAH converges. Here we explore the properties of the inner loop of SARAH for general nonconvex functions and show that it converges at the same rate as SGD, but under weaker assumptions and with better constants in the convergence rate. We then analyze the full SARAH algorithm in the case of gradient dominated functions as a special class of nonconvex functions [60, 50, 65] for which we show linear convergence to a global minimum. We will provide the definition of a gradient dominated function in Section 5.3. We also note that this type of function includes the case where the objective function F is strongly convex, but the component functions f_i, $i \in [n]$, are not necessarily convex.

We now summarize the complexity results of SARAH and other existing methods for nonconvex functions in Table 5.1. All complexity estimates are in terms of the number of

[1] Note that numerous modifications of stochastic gradient algorithms have been proposed, including non-uniform sampling, acceleration, repeated scheme and asynchronous parallelization. In this chapter, we refrain from checking and analyzing those variants, and compare only the primary methods.

calls to the *incremental first order oracle* (IFO) defined in [1], in other words computations of $(f_i(w), \nabla f_i(w))$ for some $i \in [n]$. The iteration complexity analysis aims to bound the number of iterations $\mathcal{T}$, which is needed to guarantee that $\|\nabla F(w_{\mathcal{T}})\|^2 \leq \epsilon$. In this case we will say that $w_{\mathcal{T}}$ is an ϵ-accurate solution. However, it is common practice for stochastic gradient algorithms to obtain the bound on the number of IFOs after which the algorithm can be terminated with the guaranteed the bound on the expectation, as follows,

$$\mathbb{E}[\|\nabla F(w_{\mathcal{T}})\|^2] \leq \epsilon. \tag{5.2}$$

It is important to note that for the stochastic algorithms discussed here, the output $w_{\mathcal{T}}$ is not the last iterate computed by the algorithm, but a randomly selected iterate from the computed sequence.

Let us discuss the results in Table 5.1. The analysis of SGD in [22] in performed under the assumption that $\|\nabla f_i(\cdot)\| \leq \sigma$, for all $i \in [n]$, for some fixed constant σ. This limits the applicability of the convergence results for SGD and adds dependence on σ which can be large. In contrast, convergence rate of SVRG only requires L-Lipschitz continuity of the gradient as does the analysis of SARAH. Convergence of SVRG for general nonconvex functions is better than that of the inner loop of SARAH in terms of its dependence on ϵ, but it is worse in term of its dependence on n. In addition the bound for SVRG includes an unknown universal constant ν, whose magnitude is not clear and can be quite small. Convergence rate of the full SARAH algorithm for general nonconvex functions remains an open question. In the case of τ-gradient dominated functions, full SARAH convergence rate dominates that of the other algorithms.

Table 5.1: Comparisons between different algorithms for nonconvex functions.

Method	GD ([49, 65])	SGD ([22, 65])	SVRG ([65])	**SARAH**
Nonconvex	$\mathcal{O}\left(\frac{nL}{\epsilon}\right)$	$\mathcal{O}\left(\frac{L\sigma^2}{\epsilon^2}\right)$	$\mathcal{O}\left(n + \frac{n^{2/3}L}{\nu\epsilon}\right)$	$\mathcal{O}\left(n + \frac{L^2}{\epsilon^2}\right)$
τ-Gradient Dominated	$\mathcal{O}\left(nL\tau \log(\frac{1}{\epsilon})\right)$	$\mathcal{O}\left(\frac{L\tau\sigma^2}{\epsilon^2}\right)$	$\mathcal{O}\left((n + \frac{n^{2/3}L\tau}{\nu}) \log(\frac{1}{\epsilon})\right)$	$\mathcal{O}\left((n + L^2\tau^2) \log(\frac{1}{\epsilon})\right)$

Our contributions. In summary, in this chapter we analyze SARAH with mini-batches for nonconvex optimization. SARAH originates from the idea of momentum SGD,

SAG/SAGA, SVRG and L-BFGS and is initially proposed for convex optimization, and is now proven to be effective for minimizing finite-sum problems of general nonconvex functions. We summarize the key contributions of the chapter as follows.

- We study and extend SARAH framework [51] with *mini-batches* to solving *nonconvex* loss functions, which cover the popular deep neural network problems. We are able to provide a sublinear convergence rate of the inner loop of SARAH for general nonconvex functions, under milder assumptions than that of SGD.
- Like SVRG [65], SARAH algorithm is shown to enjoy linear convergence rate for τ-gradient dominated functions–a special class of possibly nonconvex functions [60, 50].
- Similarly to SVRG, SARAH maintains a constant learning rate for nonconvex optimization, and a larger mini-batch size allows the use of a more aggressive learning rate and a smaller inner loop size.
- Finally, we present numerical results, where a practical version of SARAH, introduced in [51] is shown to be competitive on standard neural network training tasks.

5.2 SARAH Algorithm

The pivotal idea of SARAH, like many existing algorithms, such as SAG, SAGA and BFGS [57], is to utilize past stochastic gradient estimates to improve convergence. In contrast with SAG, SAGA and BFGS [57], SARAH does not store past information thus significantly reducing storage cost. We present SARAH as a two-loop algorithm in Figure 5.1, with SARAH-IN in Figure 5.2 describing the inner loop.

Input: $\tilde{w}_0$, the learning rate $\eta > 0$, the batch size b and the inner loop size m.

Iterate:

for $s = 1, 2, \ldots$ **do**

$\tilde{w}_s = \text{SARAH-IN}(\tilde{w}_{s-1}, \eta, b, m)$

end for

Output: $\tilde{w}_s$

Figure 5.1: Algorithm SARAH

Input: $w_0(=\tilde{w}_{s-1})$, the learning rate $\eta > 0$, the batch size b and the inner loop size m.

Evaluate the full gradient: $v_0 = \frac{1}{n}\sum_{i=1}^{n} \nabla f_i(w_0)$

Take a gradient descent step: $w_1 = w_0 - \eta v_0$

Iterate:

for $t = 1, \ldots, m-1$ **do**

Choose a mini-batch $I_t \subseteq [n]$ of size b uniformly at random (without replacement)

Update the stochastic recursive gradient:

$$v_t = \frac{1}{b}\sum_{i \in I_t} [\nabla f_i(w_t) - \nabla f_i(w_{t-1})] + v_{t-1} \tag{5.3}$$

Update the iterate: $w_{t+1} = w_t - \eta v_t$

end for

$\tilde{w} = w_t$ with t chosen uniformly randomly from $\{0, 1, \ldots, m\}$

Output: $\tilde{w}$

Figure 5.2: Algorithm SARAH within a single outer loop: SARAH-IN(w_0, η, b, m)

Similarly to SVRG, in each outer iteration, SARAH proceeds with the evaluation of a full gradient followed by an inner loop of m stochastic steps. SARAH requires one computation of the full gradient at the start of its inner loop and then proceeds by updating this gradient information using stochastic gradient estimates over m inner steps. Hence, each outer iteration corresponds to a cost of $\mathcal{O}(n+bm)$ component gradient evaluations (or IFOs). For simplicity let us consider the inner loop update for $b = 1$, as presented in [51]:

$$v_t = \nabla f_{i_t}(w_t) - \nabla f_{i_t}(w_{t-1}) + v_{t-1}, \tag{5.4}$$

Note that unlike SVRG, which uses the gradient updates $v_t = \nabla f_{i_t}(w_t) - \nabla f_{i_t}(w_0) + v_0$, SARAH's gradient estimate v_t iteratively includes all past stochastic gradients, however, SARAH consumes a memory of $\mathcal{O}(d)$ instead of $\mathcal{O}(nd)$ in the cases of SAG/SAGA and

BFGS, because this past information is simply averaged, instead of being stored.

With either $m = 1$ or $s = 1$ and $b = n$, the algorithm SARAH recovers gradient descent (GD). We remark here that we also recover the convergence rate theoretically for GD with $s = 1$ and $b = n$ In the following section, we analyze theoretical convergence properties of SARAH when applied to nonconvex functions.

5.3 Convergence Analysis

First, we will introduce the sublinear convergence of SARAH-IN for general nonconvex functions. Then we present the linear convergence of SARAH over a special class of gradient dominated functions [60, 50, 65]. Before proceeding to the analysis, let us start by stating some assumptions.

Assumption 5.3.1 (L-smooth)**.** *Each $f_i : \mathbb{R}^d \to \mathbb{R}$, $i \in [n]$, is L-smooth, i.e., there exists a constant $L > 0$ such that*

$$\|\nabla f_i(w) - \nabla f_i(w')\| \leq L\|w - w'\|, \ \forall w, w' \in \mathbb{R}^d. \tag{5.5}$$

Assumption 5.3.1 implies that F is also L-smooth. Then, by the property of L-smooth function (in [49]), we have

$$F(w) \leq F(w') + \nabla F(w')^T(w - w') + \frac{L}{2}\|w - w'\|^2, \ \forall w, w' \in \mathbb{R}^d. \tag{5.6}$$

The following assumption will be made only when appropriate, otherwise, it will be dropped.

Assumption 5.3.2 (τ-gradient dominated)**.** *F is τ-gradient dominated, i.e., there exists a constant $\tau > 0$ such that $\forall w \in \mathbb{R}^d$,*

$$F(w) - F(w_*) \leq \tau\|\nabla F(w)\|^2, \tag{5.7}$$

where w_* is a global minimizer of F.

We can observe that every stationary point of the τ-gradient dominated function F is a global minimizer. However, such a function F needs not necessarily be convex. If F is *μ-strongly convex* (but each f_i, $i \in [n]$, is possibly nonconvex), then $2\mu[F(w) - F(w_*)] \leq \|\nabla F(w)\|^2, \forall w \in \mathbb{R}^d$. Thus, a μ-strongly convex function is also $1/(2\mu)$-gradient dominated.

Lemma 5.3.1. *Suppose that Assumption 5.3.1 holds. Consider SARAH-IN (SARAH within a single outer loop in Figure 5.2), then we have*

$$\sum_{t=0}^{m} \mathbb{E}[\|\nabla F(w_t)\|^2] \leq \frac{2}{\eta}[F(w_0) - F(w_*)] + \sum_{t=0}^{m} \mathbb{E}[\|\nabla F(w_t) - v_t\|^2] - (1 - L\eta) \sum_{t=0}^{m} \mathbb{E}[\|v_t\|^2], \tag{5.8}$$

where w_* is a global minimizer of F.

Proof. By Assumption 5.3.1 and $w_{t+1} = w_t - \eta v_t$, we have

$$\begin{aligned}
\mathbb{E}[F(w_{t+1})] &\overset{(5.6)}{\leq} \mathbb{E}[F(w_t)] - \eta \mathbb{E}[\nabla F(w_t)^T v_t] + \frac{L\eta^2}{2} \mathbb{E}[\|v_t\|^2] \\
&= \mathbb{E}[F(w_t)] - \frac{\eta}{2} \mathbb{E}[\|\nabla F(w_t)\|^2] + \frac{\eta}{2} \mathbb{E}[\|\nabla F(w_t) - v_t\|^2] - \left(\frac{\eta}{2} - \frac{L\eta^2}{2}\right) \mathbb{E}[\|v_t\|^2],
\end{aligned}$$

where the last equality follows from the fact $r^T q = \frac{1}{2}\left[\|r\|^2 + \|q\|^2 - \|r - q\|^2\right]$, for any $r, q \in \mathbb{R}^d$.

By summing over $t = 0, \ldots, m$, we have

$$\begin{aligned}
\mathbb{E}[F(w_{m+1})] \leq \mathbb{E}[F(w_0)] &- \frac{\eta}{2} \sum_{t=0}^{m} \mathbb{E}[\|\nabla F(w_t)\|^2] + \frac{\eta}{2} \sum_{t=0}^{m} \mathbb{E}[\|\nabla F(w_t) - v_t\|^2] \\
&- \left(\frac{\eta}{2} - \frac{L\eta^2}{2}\right) \sum_{t=0}^{m} \mathbb{E}[\|v_t\|^2],
\end{aligned}$$

which is equivalent to ($\eta > 0$):

$$\begin{aligned}
\sum_{t=0}^{m} \mathbb{E}[\|\nabla F(w_t)\|^2] &\leq \frac{2}{\eta} \mathbb{E}[F(w_0) - F(w_{m+1})] + \sum_{t=0}^{m} \mathbb{E}[\|\nabla F(w_t) - v_t\|^2] \\
&\quad - (1 - L\eta) \sum_{t=0}^{m} \mathbb{E}[\|v_t\|^2] \\
&\leq \frac{2}{\eta}[F(w_0) - F(w_*)] + \sum_{t=0}^{m} \mathbb{E}[\|\nabla F(w_t) - v_t\|^2] - (1 - L\eta) \sum_{t=0}^{m} \mathbb{E}[\|v_t\|^2],
\end{aligned}$$

where the last inequality follows since w_* is a global minimizer of F. (Note that w_0 is given.) $\square$

Lemma 5.3.2. *Suppose that Assumption 5.3.1 holds. Consider v_t defined by* (5.3) *in SARAH-IN, then for any $t \geq 1$,*

$$\mathbb{E}[\|\nabla F(w_t) - v_t\|^2] = \sum_{j=1}^{t} \mathbb{E}[\|v_j - v_{j-1}\|^2] - \sum_{j=1}^{t} \mathbb{E}[\|\nabla F(w_j) - \nabla F(w_{j-1})\|^2].$$

Proof. Let $\mathcal{F}_j = \sigma(w_0, i_1, i_2, \ldots, i_{j-1})$ be the σ-algebra generated by $w_0, i_1, i_2, \ldots, i_{j-1}$; $\mathcal{F}_0 = \mathcal{F}_1 = \sigma(w_0)$. Note that $\mathcal{F}_j$ also contains all the information of $w_0, \ldots, w_j$ as well as $v_0, \ldots, v_{j-1}$. For $j \geq 1$, we have

$$\begin{aligned}
&\mathbb{E}[\|\nabla F(w_j) - v_j\|^2 | \mathcal{F}_j] \\
&= \mathbb{E}[\|[\nabla F(w_{j-1}) - v_{j-1}] + [\nabla F(w_j) - \nabla F(w_{j-1})] - [v_j - v_{j-1}]\|^2 | \mathcal{F}_j] \\
&= \|\nabla F(w_{j-1}) - v_{j-1}\|^2 + \|\nabla F(w_j) - \nabla F(w_{j-1})\|^2 + \mathbb{E}[\|v_j - v_{j-1}\|^2 | \mathcal{F}_j] \\
&\quad + 2(\nabla F(w_{j-1}) - v_{j-1})^T (\nabla F(w_j) - \nabla F(w_{j-1})) \\
&\quad - 2(\nabla F(w_{j-1}) - v_{j-1})^T \mathbb{E}[v_j - v_{j-1} | \mathcal{F}_j] \\
&\quad - 2(\nabla F(w_j) - \nabla F(w_{j-1}))^T \mathbb{E}[v_j - v_{j-1} | \mathcal{F}_j] \\
&= \|\nabla F(w_{j-1}) - v_{j-1}\|^2 - \|\nabla F(w_j) - \nabla F(w_{j-1})\|^2 + \mathbb{E}[\|v_j - v_{j-1}\|^2 | \mathcal{F}_j],
\end{aligned}$$

where the last equality follows from

$$\begin{aligned}
\mathbb{E}[v_j - v_{j-1} | \mathcal{F}_j] &\overset{(5.3)}{=} \mathbb{E}\Big[\frac{1}{b} \sum_{i \in I_j} [\nabla f_i(w_j) - \nabla f_i(w_{j-1})] \Big| \mathcal{F}_j\Big] \\
&= \frac{1}{b} \cdot \frac{b}{n} \sum_{i=1}^{n} [\nabla f_i(w_j) - \nabla f_i(w_{j-1})] = \nabla F(w_j) - \nabla F(w_{j-1}).
\end{aligned}$$

By taking expectation for the above equation, we have

$$\begin{aligned}
&\mathbb{E}[\|\nabla F(w_j) - v_j\|^2] \\
&\quad = \mathbb{E}[\|\nabla F(w_{j-1}) - v_{j-1}\|^2] - \mathbb{E}[\|\nabla F(w_j) - \nabla F(w_{j-1})\|^2] + \mathbb{E}[\|v_j - v_{j-1}\|^2].
\end{aligned}$$

Note that $\|\nabla F(w_0) - v_0\|^2 = 0$. By summing over $j = 1, \ldots, t$ $(t \geq 1)$, we have

$$\mathbb{E}[\|\nabla F(w_t) - v_t\|^2] = \sum_{j=1}^{t} \mathbb{E}[\|v_j - v_{j-1}\|^2] - \sum_{j=1}^{t} \mathbb{E}[\|\nabla F(w_j) - \nabla F(w_{j-1})\|^2].$$

□

With the above Lemmas, we can derive the following upper bound for $\mathbb{E}[\|\nabla F(w_t) - v_t\|^2]$.

Lemma 5.3.3. *Suppose that Assumption 5.3.1 holds. Consider v_t defined by* (5.3) *in SARAH-IN. Then for any $t \geq 1$,*

$$\mathbb{E}[\|\nabla F(w_t) - v_t\|^2] \leq \frac{1}{b}\left(\frac{n-b}{n-1}\right) L^2 \eta^2 \sum_{j=1}^{t} \mathbb{E}[\|v_{j-1}\|^2].$$

Proof. Let

$$\xi_t = \nabla f_t(w_j) - \nabla f_t(w_{j-1}). \tag{5.9}$$

We have

$$\begin{aligned}
&\mathbb{E}[\|v_j - v_{j-1}\|^2 | \mathcal{F}_j] - \|\nabla F(w_j) - \nabla F(w_{j-1})\|^2 \\
&\quad \overset{(5.3)}{=} \mathbb{E}\Big[\Big\|\frac{1}{b}\sum_{i\in I_j}[\nabla f_i(w_j) - \nabla f_i(w_{j-1})]\Big\|^2 \Big| \mathcal{F}_j\Big] - \Big\|\frac{1}{n}\sum_{i=1}^{n}[\nabla f_i(w_j) - \nabla f_i(w_{j-1})]\Big\|^2 \\
&\quad \overset{(5.9)}{=} \mathbb{E}\Big[\Big\|\frac{1}{b}\sum_{i\in I_j}\xi_i\Big\|^2 \Big| \mathcal{F}_j\Big] - \Big\|\frac{1}{n}\sum_{i=1}^{n}\xi_i\Big\|^2 \\
&\quad = \frac{1}{b^2}\mathbb{E}\Big[\sum_{i\in I_j}\sum_{k\in I_j}\xi_i^T\xi_k \Big| \mathcal{F}_j\Big] - \frac{1}{n^2}\sum_{i=1}^{n}\sum_{k=1}^{n}\xi_i^T\xi_k \\
&\quad = \frac{1}{b^2}\mathbb{E}\Big[\sum_{i\neq k\in I_j}\xi_i^T\xi_k + \sum_{i\in I_j}\xi_i^T\xi_i \Big| \mathcal{F}_j\Big] - \frac{1}{n^2}\sum_{i=1}^{n}\sum_{k=1}^{n}\xi_i^T\xi_k \\
&\quad = \frac{1}{b^2}\Big[\frac{b}{n}\frac{(b-1)}{(n-1)}\sum_{i\neq k}\xi_i^T\xi_k + \frac{b}{n}\sum_{i=1}^{n}\xi_i^T\xi_i\Big] - \frac{1}{n^2}\sum_{i=1}^{n}\sum_{k=1}^{n}\xi_i^T\xi_k \\
&\quad = \frac{1}{b^2}\Big[\frac{b}{n}\frac{(b-1)}{(n-1)}\sum_{i=1}^{n}\sum_{k=1}^{n}\xi_i^T\xi_k + \Big(\frac{b}{n} - \frac{b}{n}\frac{(b-1)}{(n-1)}\Big)\sum_{i=1}^{n}\xi_i^T\xi_i\Big] - \frac{1}{n^2}\sum_{i=1}^{n}\sum_{k=1}^{n}\xi_i^T\xi_k \\
&\quad = \frac{1}{bn}\Big[\Big(\frac{(b-1)}{(n-1)} - \frac{b}{n}\Big)\sum_{i=1}^{n}\sum_{k=1}^{n}\xi_i^T\xi_k + \frac{(n-b)}{(n-1)}\sum_{i=1}^{n}\xi_i^T\xi_i\Big]
\end{aligned}$$

$$
\begin{aligned}
&= \frac{1}{bn}\left(\frac{n-b}{n-1}\right)\Big[-\frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{n}\xi_i^T\xi_k + \sum_{i=1}^{n}\xi_i^T\xi_i\Big] \\
&= \frac{1}{bn}\left(\frac{n-b}{n-1}\right)\Big[-n\Big\|\frac{1}{n}\sum_{i=1}^{n}\xi_i\Big\|^2 + \sum_{i=1}^{n}\|\xi_i\|^2\Big] \\
&\leq \frac{1}{b}\left(\frac{n-b}{n-1}\right)\frac{1}{n}\sum_{i=1}^{n}\|\xi_i\|^2 \\
&\overset{(5.9)}{=} \frac{1}{b}\left(\frac{n-b}{n-1}\right)\frac{1}{n}\sum_{i=1}^{n}\|\nabla f_i(w_j) - \nabla f_i(w_{j-1})\|^2 \\
&\overset{(5.5)}{\leq} \frac{1}{b}\left(\frac{n-b}{n-1}\right)L^2\eta^2\frac{1}{n}\sum_{i=1}^{n}\|v_{j-1}\|^2 \\
&= \frac{1}{b}\left(\frac{n-b}{n-1}\right)L^2\eta^2\|v_{j-1}\|^2.
\end{aligned}
$$

Hence, by taking expectation, we have

$$
\mathbb{E}[\|v_j - v_{j-1}\|^2] - \mathbb{E}[\|\nabla F(w_j) - \nabla F(w_{j-1})\|^2] \leq \frac{1}{b}\left(\frac{n-b}{n-1}\right)L^2\eta^2\mathbb{E}[\|v_{j-1}\|^2].
$$

By Lemma 5.3.2, for $t \geq 1$,

$$
\begin{aligned}
\mathbb{E}[\|\nabla F(w_t) - v_t\|^2] &= \sum_{j=1}^{t}\mathbb{E}[\|v_j - v_{j-1}\|^2] - \sum_{j=1}^{t}\mathbb{E}[\|\nabla F(w_j) - \nabla F(w_{j-1})\|^2] \\
&\leq \frac{1}{b}\left(\frac{n-b}{n-1}\right)L^2\eta^2\sum_{j=1}^{t}\mathbb{E}[\|v_{j-1}\|^2].
\end{aligned}
$$

This completes the proof.

However, the result simply follows for the case when $b = 1$ by the alternative proof. We have

$$
\|v_t - v_{t-1}\|^2 \overset{(6.4)}{=} \|\nabla f_{i_t}(w_t) - \nabla f_{i_t}(w_{t-1})\|^2 \overset{(5.5)}{\leq} L^2\|w_t - w_{t-1}\|^2 = L^2\eta^2\|v_{t-1}\|^2,\ t \geq 1. \tag{5.10}
$$

Hence, by Lemma 5.3.2, we have

$$
\mathbb{E}[\|\nabla F(w_t) - v_t\|^2] \leq \sum_{j=1}^{t}\mathbb{E}[\|v_j - v_{j-1}\|^2] \overset{(6.30)}{\leq} L^2\eta^2\sum_{j=1}^{t}\mathbb{E}[\|v_{j-1}\|^2].
$$

□

Using the above lemmas, we are now able to obtain the following convergence rate result for SARAH-IN.

Theorem 5.3.1. *Suppose that Assumption 5.3.1 holds. Consider SARAH-IN (SARAH within a single outer loop in Figure 5.2) with*

$$\eta \leq \frac{2}{L\left(\sqrt{1+\frac{4m}{b}\left(\frac{n-b}{n-1}\right)}+1\right)}. \tag{5.11}$$

Then we have

$$\mathbb{E}[\|\nabla F(\tilde{w})\|^2] \leq \frac{2}{\eta(m+1)}[F(w_0)-F(w_*)],$$

where w_* is a global minimizer of F, and $\tilde{w}=w_t$, where t is chosen uniformly at random from $\{0,1,\ldots,m\}$.

Proof. By Lemma 6.3.11, we have

$$\mathbb{E}[\|\nabla F(w_t)-v_t\|^2] \leq \frac{1}{b}\left(\frac{n-b}{n-1}\right)L^2\eta^2\sum_{j=1}^{t}\mathbb{E}[\|v_{j-1}\|^2].$$

Note that $\|\nabla F(w_0)-v_0\|^2=0$. Hence, by summing over $t=0,\ldots,m$ $(m\geq 1)$, we have

$$\sum_{t=0}^{m}\mathbb{E}\|v_t-\nabla F(w_t)\|^2 \leq \frac{1}{b}\left(\frac{n-b}{n-1}\right)L^2\eta^2\Big[m\mathbb{E}\|v_0\|^2+(m-1)\mathbb{E}\|v_1\|^2+\cdots+\mathbb{E}\|v_{m-1}\|^2\Big].$$

We have

$$\begin{aligned}
&\sum_{t=0}^{m}\mathbb{E}[\|\nabla F(w_t)-v_t\|^2]-(1-L\eta)\sum_{t=0}^{m}\mathbb{E}[\|v_t\|^2]\\
&\leq \frac{1}{b}\left(\frac{n-b}{n-1}\right)L^2\eta^2\Big[m\mathbb{E}\|v_0\|^2+(m-1)\mathbb{E}\|v_1\|^2+\cdots+\mathbb{E}\|v_{m-1}\|^2\Big]\\
&-(1-L\eta)\Big[\mathbb{E}\|v_0\|^2+\mathbb{E}\|v_1\|^2+\cdots+\mathbb{E}\|v_m\|^2\Big]\\
&\leq \Big[\frac{1}{b}\left(\frac{n-b}{n-1}\right)L^2\eta^2 m-(1-L\eta)\Big]\sum_{t=1}^{m}\mathbb{E}[\|v_{t-1}\|^2] \overset{(5.11)}{\leq} 0,
\end{aligned} \tag{5.12}$$

since

$$\eta = \frac{2}{L\left(\sqrt{1+\frac{4m}{b}\left(\frac{n-b}{n-1}\right)}+1\right)}$$

is a root of equation

$$\frac{1}{b}\left(\frac{n-b}{n-1}\right)L^2\eta^2 m - (1-L\eta) = 0.$$

Therefore, by Lemma 5.3.1, we have

$$\begin{aligned}\sum_{t=0}^{m}\mathbb{E}[\|\nabla F(w_t)\|^2] &\leq \frac{2}{\eta}[F(w_0)-F(w_*)] + \sum_{t=0}^{m}\mathbb{E}[\|\nabla F(w_t)-v_t\|^2] - (1-L\eta)\sum_{t=0}^{m}\mathbb{E}[\|v_t\|^2] \\ &\overset{(5.12)}{\leq} \frac{2}{\eta}[F(w_0)-F(w_*)].\end{aligned}$$

If $\tilde{w} = w_t$, where t is chosen uniformly at random from $\{0, 1, \ldots, m\}$, then

$$\mathbb{E}[\|\nabla F(\tilde{w})\|^2] = \frac{1}{m+1}\sum_{t=0}^{m}\mathbb{E}[\|\nabla F(w_t)\|^2] \leq \frac{2}{\eta(m+1)}[F(w_0)-F(w_*)].$$

□

This result shows a sublinear convergence rate for SARAH-IN with increasing m. Consequently, with $b = 1$ and $\eta = \frac{2}{L(\sqrt{1+4m}+1)}$, to obtain

$$\mathbb{E}[\|\nabla F(\tilde{w})\|^2] \leq \frac{L(\sqrt{1+4m}+1)}{(m+1)}[F(w_0)-F(w_*)] \leq \epsilon,$$

it is sufficient to make $m = \mathcal{O}(L^2/\epsilon^2)$. Hence, the total complexity to achieve an ϵ-accurate solution is $(n + 2m) = \mathcal{O}(n + L^2/\epsilon^2)$. Therefore, we have the following conclusion for complexity bound.

Corollary 5.3.1. *Suppose that Assumption 5.3.1 holds. Consider SARAH within a single outer iteration with batch size $b = 1$ and the learning rate $\eta = \mathcal{O}(1/(L\sqrt{m}))$ where m is the total number of iterations, then $\|\nabla F(w_t)\|^2$ converges sublinearly in expectation with a rate of $\mathcal{O}(L/\sqrt{m})$, and therefore, the total complexity to achieve an ϵ-accurate solution defined*

in (6.6) *is* $\mathcal{O}(n + L^2/\epsilon^2)$.

Finally, we present the result for SARAH with multiple outer iterations in application to the class of gradient dominated functions defined in (5.7).

Theorem 5.3.2. *Suppose that Assumptions 5.3.1 and 5.3.2 hold. Consider SARAH (in Figure 5.1) with* η *and* m *such that*

$$\eta \leq \frac{2}{L\left(\sqrt{1+\frac{4m}{b}\left(\frac{n-b}{n-1}\right)}+1\right)} \qquad \textit{and} \qquad \frac{\eta(m+1)}{2} > \tau.$$

Then we have

$$\mathbb{E}[\|\nabla F(\tilde{w}_s)\|^2] \leq (\bar{\gamma}_m)^s \|\nabla F(\tilde{w}_0)\|^2,$$

where

$$\bar{\gamma}_m = \frac{2\tau}{\eta(m+1)} < 1.$$

Proof. Note that $\tilde{w}_s = \tilde{w}$ and $w_0 = \tilde{w}_{s-1}$, $s \geq 1$. By Theorem 5.3.1, we have

$$\begin{aligned}
\mathbb{E}[\|\nabla F(\tilde{w}_s)\|^2 | \tilde{w}_{s-1}] = \mathbb{E}[\|\nabla F(\tilde{w})\|^2 | w_0] &\leq \frac{2}{\eta(m+1)}[F(w_0) - F(w_*)] \\
&\overset{(5.7)}{\leq} \frac{2\tau}{\eta(m+1)}\|\nabla F(w_0)\|^2 \\
&= \frac{2\tau}{\eta(m+1)}\|\nabla F(\tilde{w}_{s-1})\|^2.
\end{aligned}$$

Hence, taking expectation to have

$$\mathbb{E}[\|\nabla F(\tilde{w}_s)\|^2] \leq \frac{2\tau}{\eta(m+1)}\mathbb{E}[\|\nabla F(\tilde{w}_{s-1})\|^2] \leq \left[\frac{2\tau}{\eta(m+1)}\right]^s \|\nabla F(\tilde{w}_0)\|^2.$$

□

Consider the case when $b = 1$ and $\eta = \frac{2}{L(\sqrt{1+4m}+1)}$. We need $m = \mathcal{O}\left(L^2\tau^2\right)$ to satisfy

$\frac{\eta(m+1)}{2} = \frac{m+1}{L\sqrt{1+4m}+1} > \tau$. To obtain

$$\mathbb{E}[\|\nabla F(\tilde{w}_s)\|^2] \leq (\bar{\gamma}_m)^s \|\nabla F(\tilde{w}_0)\|^2 \leq \epsilon,$$

it is sufficient to have $s = \mathcal{O}(\log(1/\epsilon))$. This implies the total complexity to achieve an ϵ-accurate solution is $(n + 2m)s = \mathcal{O}\left((n + L^2\tau^2)\log(1/\epsilon)\right)$ and we can summarize the conclution as follows.

Corollary 5.3.2. *Suppose that Assumptions 5.3.1 and 5.3.2 hold. Consider SARAH with parameters from Theorem 5.3.2 with batch size $b = 1$ and the learning rate $\eta = \mathcal{O}(1/(L\sqrt{m}))$, then the total complexity to achieve an ϵ-accurate solution defined in* (6.6) *is $\mathcal{O}\left((n + L^2\tau^2)\log(1/\epsilon)\right)$.*

5.4 Discussions on the Mini-batches Sizes

Let us discuss two simple corollaries of Theorem 5.3.1.

The first corollary is obtained trivially by substituting the learning rate into the complexity bound in Theorem 5.3.1.

Corollary 5.4.1. *Suppose that Assumption 5.3.1 holds. Consider SARAH-IN (SARAH within a single outer loop in Figure 5.2) with*

$$\eta = \frac{2}{L\left(\sqrt{1 + \frac{4m}{b}\left(\frac{n-b}{n-1}\right)} + 1\right)}. \tag{5.13}$$

Then we have

$$\mathbb{E}[\|\nabla F(\tilde{w})\|^2] \leq \frac{L\left(\sqrt{1 + \frac{4m}{b}\left(\frac{n-b}{n-1}\right)} + 1\right)}{(m+1)}[F(w_0) - F(w_*)],$$

where w_* is a global minimizer of F, and $\tilde{w} = w_t$, where t is chosen uniformly at random from $\{0, 1, \ldots, m\}$.

Remark 5.4.1. *We can clearly observe that the rate of convergence for SARAH-IN depends on the size of b. For a larger value of b, we can use a more aggressive learning rate and it*

requires the smaller number of iterations to achieve an ϵ-accurate solution. In particular, when $b = n$, SARAH-IN reduces to the GD method and its convergence rate becomes that of gradient descent,

$$\mathbb{E}[\|\nabla F(\tilde{w})\|^2] \leq \frac{2L}{(m+1)}[F(w_0) - F(w_*)],$$

and the total complexity to achieve an ϵ-accurate solution is $n \cdot m = \mathcal{O}\left(\frac{nL}{\epsilon}\right)$. However, the total work in terms of IFOs increases with b. When $b \neq n$, the total complexity to achieve an ϵ-accurate solution is $(n + 2bm) = \mathcal{O}\left(n + \frac{L^2}{\epsilon^2}\left(\frac{n-b}{n-1}\right)\right)$.

Let us set $m = n - 1$ in Corollary 5.4.1, we can achieve the following result.

Corollary 5.4.2. *Suppose that Assumption 5.3.1 holds. Consider SARAH-IN with $m = n - 1$, and*

$$\eta = \frac{2}{L\left(\sqrt{4(n/b) - 3} + 1\right)}.$$

Then we have

$$\mathbb{E}[\|\nabla F(\tilde{w})\|^2] \leq \frac{L\left(\sqrt{4(n/b) - 3} + 1\right)}{n}[F(w_0) - F(w_*)],$$

where w_* is a global minimizer of F, and $\tilde{w} = w_t$, where t is chosen uniformly at random from $\{0, 1, \ldots, n-1\}$.

Remark 5.4.2. *For SARAH-IN with the number of iterations $m = n - 1$ and the learning rate $\eta = \mathcal{O}\left(1/(L\sqrt{(n/b)})\right)$, we could achieve a convergence rate of $\mathcal{O}(L/\sqrt{bn})$. We can observe that the value of b significantly affects the rate. For example, when $b = n/\beta$, $\beta > 1$ and $b = n^\alpha$, $\alpha < 1$, the convergence rates become $\mathcal{O}(L\sqrt{\beta}/n)$ and $\mathcal{O}(L/\sqrt{n^{\alpha+1}})$, respectively.*

5.5 Numerical Experiments

We now turn to the numerical study and conduct experiments on the multiclass classification problem with neural networks, which is the typical challegeing nonconvex problem in machine learning.

SARAH+ as a Practical Variant [51] proposes SARAH+ as a practical variant of SARAH. Now we propose SARAH+ for the nonconvex optimization by running Algorithm SARAH (Figure 5.1) with the following SARAH-IN algorithm (Figure 5.3). Notice that SARAH+ is different from SARAH in that the inner loop is terminated adaptively instead of using a fixed choice of the inner loop size m. This is idea is based on the fact that the norm $\|v_t\|$ converges to zero expectation, which has been both proven theoretically and verified numerically for convex optimization in [51]. Under the assumption that similar behavior happens in the nonconvex case, instead of tuning the inner loop size for SARAH, we believe that a proper choice of the ratio γ below, the automatic loop termination can give superior or competitive performances.

Input: $w_0 (= \tilde{w}_{s-1})$, the learning rate $\eta > 0$, the batch size b, and the maximum inner loop size m.

Evaluate the full gradient: $v_0 = \frac{1}{n} \sum_{i=1}^{n} \nabla f_i(w_0)$

Take a gradient descent step: $w_1 = w_0 - \eta v_0$

Iterate:

while $\|v_{t-1}\|^2 > \gamma \|v_0\|^2$ **and** $t < m$ **do**

Choose a mini-batch $I_t \subseteq [n]$ of size b uniformly at random

Update the stochastic recursive gradient:

$$v_t = \frac{1}{b} \sum_{i \in I_t} [\nabla f_i(w_t) - \nabla f_i(w_{t-1})] + v_{t-1}$$

Update the iterate and index: $w_{t+1} = w_t - \eta v_t$; $t = t + 1$

end while

$\tilde{w} = w_t$ with t chosen uniformly randomly from $\{0, 1, \ldots, m\}$

Output: $\tilde{w}$

Figure 5.3: Algorithm SARAH within a single outer loop: SARAH-IN(w_0, η, b, m)

Networks and Datasets We perform numerical experiments with neural nets with one fully connected hidden layer of n_h nodes, followed by a fully connected output layer

which feeds into the softmax regression and cross entropy objective, with the weight decay regularizer (ℓ_2-regularizer) with parameter λ. We test the performance on the datasets MNIST [36] [2] and CIFAR10 [33] [3] with $n_h = 300, \lambda$ =1e-04 and $n_h = 100, \lambda$ =1e-03, respectively. Both datasets have 10 classes, i.e. 10 softmax output nodes in the network, and are normalized to interval $[0, 1]$ as a simple data pre-processing. This network of MNIST achieves the best performance for neural nets with a single hidden layer. Information on both datasets is also available in Table 5.2.

Optimization Details We compare the efficiency of SARAH, SARAH+ [51], SVRG [65], AdaGrad [18] and SGD-M (momentum SGD [61, 76]) [4] numerically in terms of number of effective data passes, where the last two algorithms are efficient SGD variants available in the Google open-source library *Tensorflow* [5]. As the choice of initialization for the weight parameters is very important, we apply a widely used mechanism called *normalized initialization* [23] where the weight parameters between layers j and $j+1$ are sampled uniformly from $\left[-\sqrt{6/(n_j+n_{j+1})}, \sqrt{6/(n_j+n_{j+1})}\right]$. In addition, we use mini-batch size $b = 10$ in all the algorithms.

Table 5.2: Summary of statistics and best parameters of all the algorithms for the two datasets.

Dataset	Number of Samples $(n_{\text{train}}, n_{\text{test}})$	Dimensions (d)	SARAH (m^*, η^*)	SARAH+ (η^*)	SVRG (m^*, η^*)	AdaGrad (δ^*, η^*)	SGD-M (γ^*, η^*)
MNIST	(60,000, 10,000)	784	(0.1n, 0.08)	0.2	(0.4n, 0.08)	(0.01, 0.1)	(0.7, 0.01)
CIFAR10	(50,000, 10,000)	3072	(0.4n, 0.03)	0.02	(0.8n, 0.02)	(0.05, 1.0)	(0.7, 0.001)

Performance and Comparison We present the optimal choices of optimization parameters for the mentioned algorithms in Table 5.2, as well as their performance in Figure 5.4. As for the optimization parameters we consistently use the ratio 0.7 in SARAH+, while for all the others, we need to tune two parameters, including η^* for optimal learning rates, m^*

[2] Available at http://yann.lecun.com/exdb/mnist/.

[3] Available at https://www.cs.toronto.edu/~kriz/cifar.html.

[4] While SARAH, SVRG, SGD have been proven effective for nonconvex optimization, as far as we know, the SGD variants AdaGrad and SGD-M do not have theoretical convergence for nonconvex optimization.

[5] See https://www.tensorflow.org.

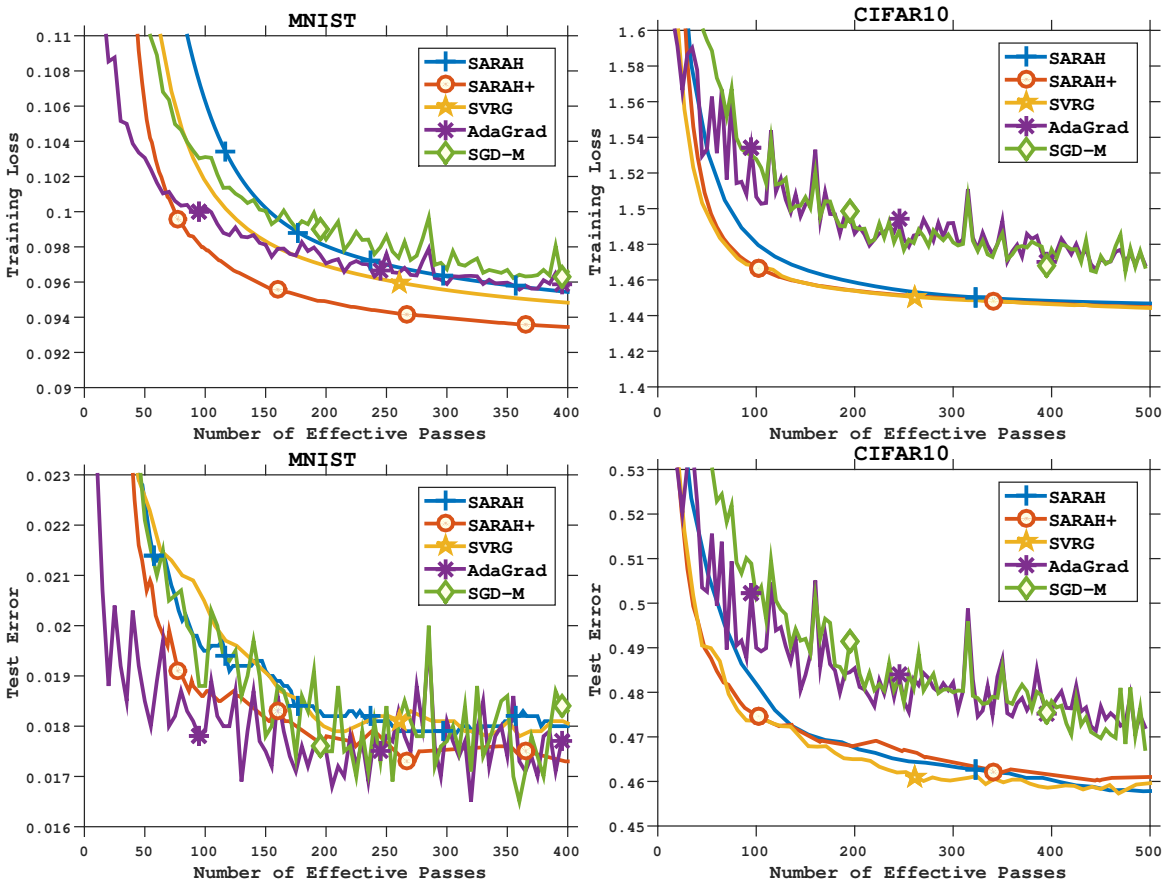


Figure 5.4: An example of ℓ_2-regularized neural nets on *MNIST* and *CIFAR10* training/testing datasets for SARAH, SARAH+, SVRG, AdaGrad and SGD-M.

for optimal inner loop size, δ^* for the optimal initial accumulator and γ^* for the optimal momentum. For the tuning of the parameters, reasonable ranges for the parameters have been scanned and we selected the best parameters in terms ofthe training error reduction.

Figure 5.4 compares the training losses (top) and the test errors (bottom), obtained by the tested algorithms on *MNIST* and *CIFAR10*, in terms of the number of effective passes through the data. On the *MNIST* dataset, which is deemed to be easier for traning, all the methods achieve similar performance in the end; however, SARAH(+) and SVRG stabilize faster than AdaGrad and SGD-M - the two of the most popular SGD variants; meanwhile, SARAH+ has shown superior performance in minimizing the training loss. For the other, more difficult, *CIFAR10* dataset, SARAH(+) and SVRG improve upon the training accuracy considerably in comparison with AdaGrad and SGD-M, and as a result, a similar advantage can be seen in the test error reduction.

5.6 Conclusion

In this chapter of work, we study and extend SARAH framework to nonconvex optimization, also admitting the practical variant, SARAH+. For smooth nonconvex functions, the inner loop of SARAH achieves the best sublinear convergence rate in the literature, while the full variant of SARAH has the same linear convergence rate and same as SVRG, for a special class of gradient dominated functions. In addition, we also analyze the dependence of the convergence of SARAH on the size of the mini-batches. In the end, we validate SARAH(+) numerically in comparison with SVRG, AdaGrad and SGD-M, with the popular nonconvex application of neural networks.

Chapter 6

Inexact SARAH

In this chapter, we develop and analyze an extension of the SARAH algorithm which can be applied to stochastic optimization problems rather than finite sum problems. The original SARAH algorithm, as well its predecessor, SVRG, cannot be applied to stochastic problems since they require exact gradient information. The inexact version of SARAH, which we develop here requires only stochastic gradient information computed on a mini-batch of sufficient size. Hence the proposed combines variance reduction via sample size selection as well as iterative stochastic gradient updates. We analyze the convergence rate of the algorithms for strongly convex, convex, and nonconvex cases with appropriate mini-batch size selected for each case.

6.1 Introduction

We consider the problem of stochastic optimization

$$\min_{w \in \mathbb{R}^d} \{F(w) = \mathbb{E}[f(w;\xi)]\}, \tag{6.1}$$

where ξ is a random variable. One of the most popular applications of this problem is expected risk minimization in supervised learning. In this case random variable ξ represents a random data sample (x, y), or a set of such samples $\{(x_i, y_i)\}_{i \in I}$. When can then consider a set of realization $\{\xi_{[i]}\}_{i=1}^n$ of ξ corresponding to a set of random samples $\{(x_i, y_i)\}_{i=1}^n$,

and define $f_i(w) := f(w; \xi_{[i]})$. Then the sample average approximation of $F(w)$, known as empirical risk in supervised learning, is written as

$$\min_{w \in \mathbb{R}^d} \left\{ F(w) = \frac{1}{n} \sum_{i=1}^{n} f_i(w) \right\}. \tag{6.2}$$

Throughout the chapter, we assume the existence of unbiased gradient estimator, that is $\mathbb{E}[\nabla f(w; \xi)] = \nabla F(w)$ for any fixed $w \in \mathbb{R}^d$. In addition we assume that there exists a lower bound of function F.

In recent years, a class of variance reduction methods [34, 16, 28, 51] has been proposed for problem (6.2) which have smaller computational complexity than both, the full gradient descent method and the stochastic gradient method. All these methods rely on the finite sum form of (6.2) and are, thus, not readily extendable to (6.1). In particular, SVRG [28] and SARAH [51] are two similar methods that consist of an outer loop, which includes one exact gradient computation at each outer iteration and an inner loop with multiple iterative stochastic gradient updates. The only difference between SVRG and SARAH is how the iterative updates are performed in the inner loop. The advantage of SARAH is that the inner loop itself results in a convergent stochastic gradient algorithm. Hence, it is possible to apply only one loop of SARAH with sufficiently large number of steps to obtain an approximately optimal solution (in expectation). The convergence behavior of one-loop SARAH is similar to that or the standard stochastic gradient method [51]. The multiple-loop SARAH algorithm matches convergence rates of SVRG, however, due to its convergent inner loop has an additional practical advantage of being able to use an adaptive inner loop size (see [51] for details).

A version of SVRG algorithm, SCSG, which drops the exact gradient requirement and replaces it with a mini-batch of stochastic gradients and inner loop size is generated randomly from geometric distribution with a success probability based on the mini-batch size, has been recently proposed and analyzed in [37, 38]. While this method has been developed for (6.2) it can be directly applied to (6.1). In this chapter, we propose and analyze an inexact version of SARAH (iSARAH) which can be applied to solve (6.1). Instead of exact gradient computation, a sample gradient is sufficient for inexact SARAH, with appropri-

ately chosen sample size. We develop total sample complexity analysis for this method under various convexity assumptions on $F(w)$. These complexity results are summarized in Tables 6.1-6.3 and are compared to the result for SCSG from [37, 38] when applied to (6.1). We also list the complexity bounds for SVRG, SARAH and SCSG when applied to finite sum problem (6.2).

All of the complexity results that we compare in Tables 6.1-6.3 are developed under the assumption that $F(w)$ is L-smooth. Table 6.1 shows the complexity results in the case when $F(w)$ is strongly convex case, with κ denoting the condition number. We observe that iSARAH achieves the best complexity bounds among the methods applicable to stochastic problems. The general convex case is summarized in Table 6.2. In this case, iSARAH (multiple loop) again achieves the best convergence rate among the stochastic methods, but under an additional (reasonable) assumption (Assumption 6.3.4). In the nonconvex case, SCSG achieves the best convergence rate under the bounded variance assumption, which requires that $\mathbb{E}[\|\nabla f(w;\xi) - \nabla F(w)\|^2] \leq C$, for some $C > 0$ and $\forall w \in \mathbb{R}^d$. While convergence rate of iSARAH (multiple loop) for nonconvex remains an open question, we were able to derive a convergence rates for iSARAH (one loop) without the bounded variance assumption. This convergence rate is naturally slower, since the one-loop iSARAH method is not a variance reduction method.

Table 6.1: Comparison results (Strongly convex)

Method	**Bound**	**Problem type**
SARAH (multiple loop) [51]	$\mathcal{O}\left((n+\kappa)\log\left(\frac{1}{\epsilon}\right)\right)$	Finite-sum
SVRG [28, 65]	$\mathcal{O}\left((n+\kappa)\log\left(\frac{1}{\epsilon}\right)\right)$	Finite-sum
SCSG [37, 38]	$\mathcal{O}\left(\left(\min\left\{\frac{\kappa}{\epsilon}, n\right\}+\kappa\right)\log\left(\frac{1}{\epsilon}\right)\right)$	Finite-sum
SCSG	$\mathcal{O}\left(\left(\frac{\kappa}{\epsilon}+\kappa\right)\log\left(\frac{1}{\epsilon}\right)\right)$	Expectation
SGD [53]	$\mathcal{O}\left(\frac{1}{\epsilon}\right)$	Expectation
iSARAH (multiple loop)	$\mathcal{O}\left(\left(\max\left\{\frac{1}{\epsilon}, \kappa\right\}+\kappa\right)\log\left(\frac{1}{\epsilon}\right)\right)$	Expectation

6.1.1 Organization

The rest of the chapter is organized as follows. In Section 6.2, we describe Inexact SARAH (iSARAH) algorithm in detail. We provide the convergence analysis of iSARAH in Section 6.3 including one-loop results in the strongly convex, convex, and nonconvex cases; and multiple-loop results in the strongly convex and convex cases. Section 3.5 contains our

Table 6.2: Comparison results (General convex)

Method	Bound	Problem type	Additional assumption
SARAH (one loop) [51, 54]	$\mathcal{O}\left(n+\frac{1}{\epsilon^2}\right)$	Finite-sum	None
SARAH (multiple loop) [51]	$\mathcal{O}\left(\left(n+\frac{1}{\epsilon}\right)\log\left(\frac{1}{\epsilon}\right)\right)$	Finite-sum	Assumptions 6.3.4
SVRG [28, 65]	$\mathcal{O}\left(n+\frac{\sqrt{n}}{\epsilon}\right)$	Finite-sum	None
SCSG [37, 38]	$\mathcal{O}\left(\min\left\{\frac{1}{\epsilon^2},\frac{n}{\epsilon}\right\}\right)$	Finite-sum	None
SCSG	$\mathcal{O}\left(\frac{1}{\epsilon^2}\right)$	Expectation	None
SGD	$\mathcal{O}\left(\frac{1}{\epsilon^2}\right)$	Expectation	Bounded variance
iSARAH (one loop)	$\mathcal{O}\left(\frac{1}{\epsilon^2}\right)$	Expectation	None
iSARAH (multiple loop)	$\mathcal{O}\left(\frac{1}{\epsilon}\log\left(\frac{1}{\epsilon}\right)\right)$	Expectation	Assumption 6.3.4

Table 6.3: Comparison results (Nonconvex)

Method	Bound	Problem type	Additional assumption
SARAH (one loop) [51, 54]	$\mathcal{O}\left(n+\frac{1}{\epsilon^2}\right)$	Finite-sum	None
SVRG [28, 65]	$\mathcal{O}\left(n+\frac{n^{2/3}}{\epsilon}\right)$	Finite-sum	None
SCSG [37, 38]	$\mathcal{O}\left(\min\left\{\frac{1}{\epsilon^{5/3}},\frac{n^{2/3}}{\epsilon}\right\}\right)$	Finite-sum	Bounded variance
SCSG	$\mathcal{O}\left(\frac{1}{\epsilon^{5/3}}\right)$	Expectation	Bounded variance
SGD	$\mathcal{O}\left(\frac{1}{\epsilon^2}\right)$	Expectation	Bounded variance
iSARAH (one loop)	$\mathcal{O}\left(\frac{1}{\epsilon^2}\right)$	Expectation	None

numerical experiments for logistic regression and neural networks. A discussion of the results and future work is in Section 2.4.

6.1.2 Basic Notation

Symbol $\mathbb{R}$ denotes the set of real number. $\mathbb{R}^d$ denotes the d-dimensional vector space. The standard Euclidean norm of a vector $w \in \mathbb{R}^d$ is denoted $\|w\|$. We denote $\mathbb{E}[\cdot|\mathcal{F}]$ as the conditional expectation with condition on σ-algebra $\mathcal{F}$. We denote ∇f is the gradient of function f. Notation $a^T b = \langle a, b\rangle$ denotes the dot product of two vectors a and b.

6.2 The Algorithm

In this section, we describe iSARAH algorithm (Algorithm 6). We first define the vector

$$v_0 = \frac{1}{b}\sum_{i=1}^{b} \nabla f(w_0;\zeta_i), \tag{6.3}$$

where $\{\zeta_i\}_{i=1}^b$ are i.i.d.[1] with $\mathbb{E}[\nabla f(w_0;\zeta_i)|w_0] = \nabla F(w_0)$. We have $\mathbb{E}[v_0|w_0] = \frac{1}{b}\sum_{i=1}^b \nabla F(w_0) = \nabla F(w_0)$.

The key step of the algorithm is a recursive update of the stochastic gradient estimate *(SARAH update)*

$$v_t = \nabla f(w_t;\xi_t) - \nabla f(w_{t-1};\xi_t) + v_{t-1}, \tag{6.4}$$

followed by the iterate update

$$w_{t+1} = w_t - \eta v_t. \tag{6.5}$$

Let $\mathcal{F}_t = \sigma(w_0, w_1, \ldots, w_t)$ be the σ-algebra generated by $w_0, w_1, \ldots, w_t$. We note that ξ_t is independent of $\mathcal{F}_t$. Hence, we have a *biased estimator* of the gradient

$$\mathbb{E}[v_t|\mathcal{F}_t] = \nabla F(w_t) - \nabla F(w_{t-1}) + v_{t-1}.$$

Algorithm 6 Inexact SARAH (iSARAH)

Parameters: the learning rate $\eta > 0$ and the inner loop size m.
Initialize: $\tilde{w}_0$
Iterate:
for $s = 1, 2, \ldots$ **do**
  $\tilde{w}_s = \text{iSARAH-IN}(\tilde{w}_{s-1}, \eta, m)$
end for
Output: $\tilde{w}_s$

Algorithm 7 iSARAH-IN(w_0, η, m)

Input: $w_0 (= \tilde{w}_{s-1})$ the learning rate $\eta > 0$ and the inner loop size m.
Generate random variables $\{\zeta_i\}_{i=1}^b$ i.i.d.
Compute $v_0 = \frac{1}{b}\sum_{i=1}^b \nabla f(w_0;\zeta_i)$
$w_1 = w_0 - \eta v_0$
Iterate:
for $t = 1, \ldots, m-1$ **do**
  Generate a random variable ξ_t
  $v_t = \nabla f(w_t;\xi_t) - \nabla f(w_{t-1};\xi_t) + v_{t-1}$
  $w_{t+1} = w_t - \eta v_t$
end for
Set $\tilde{w} = w_t$ with t chosen uniformly at random from $\{0, 1, \ldots, m\}$
Output: $\tilde{w}$

[1]Independent and identically distributed random variables. We note from probability theory that if $\zeta_1, \ldots, \zeta_b$ are i.i.d. random variables then $g(\zeta_1), \ldots, g(\zeta_b)$ are also i.i.d. random variables if g is measurable function.

iSARAH's iterations are divided into the outer loop where an inexact gradient with a batch size b is computed and the inner loop where only stochastic gradient is computed. A special property of iSARAH is that we could analyze the convergence results for a single outer loop (iSARAH-IN in Algorithm 7). In the next section, we will also provide the analysis for both iSARAH-IN (one-loop) and iSARAH (multiple-loop).

Convergence criteria. Our iteration complexity analysis aims to bound the number of outer iterations $\mathcal{T}$ (or total number of stochastic gradient evaluations) which is needed to guarantee that $\|\nabla F(w_{\mathcal{T}})\|^2 \leq \epsilon$. In this case we will say that $w_{\mathcal{T}}$ is an ϵ-accurate solution. However, as is common practice for stochastic gradient algorithms, we aim to obtain the bound on the number of iterations, which is required to guarantee the bound on the expected squared norm of a gradient, i.e.,

$$\mathbb{E}[\|\nabla F(w_{\mathcal{T}})\|^2] \leq \epsilon. \tag{6.6}$$

6.3 Convergence Analysis of iSARAH

6.3.1 Basic Assumptions

To proceed with the analysis of the proposed algorithm, we will make the following common assumptions.

Assumption 6.3.1 (L-smooth)**.** *$f(w;\xi)$ is L-smooth for every realization of ξ, i.e., there exists a constant $L > 0$ such that*

$$\|\nabla f(w;\xi) - \nabla f(w';\xi)\| \leq L\|w - w'\|, \ \forall w, w' \in \mathbb{R}^d. \tag{6.7}$$

Note that this assumption implies that $F(w) = \mathbb{E}[f(w;\xi)]$ is also *L-smooth*. The following strong convexity assumption will be made for the appropriate parts of the analysis, otherwise, it would be dropped.

Assumption 6.3.2a (μ-strongly convex)**.** *The function $F : \mathbb{R}^d \to \mathbb{R}$, is μ-strongly convex,*

i.e., there exists a constant $\mu > 0$ *such that* $\forall w, w' \in \mathbb{R}^d$,

$$F(w) \geq F(w') + \nabla F(w')^T(w - w') + \tfrac{\mu}{2}\|w - w'\|^2.$$

Another, stronger, assumption of μ-strong convexity for (6.2) will also be imposed when required in our analysis. Note that Assumption 6.3.2b implies Assumption 6.3.2a but not vice versa.

Assumption 6.3.2b. $f(w;\xi)$ *is strongly convex with* $\mu > 0$ *for every realization of* ξ.

Under Assumption 6.3.2a, let us define the (unique) optimal solution of (6.2) as w_*, Then strong convexity of F implies that

$$2\mu[F(w) - F(w_*)] \leq \|\nabla F(w)\|^2, \ \forall w \in \mathbb{R}^d. \tag{6.8}$$

We note here, for future use, that for strongly convex functions of the form (6.2), arising in machine learning applications, the condition number is defined as $\kappa \overset{\text{def}}{=} L/\mu$. Furthermore, we should also notice that Assumptions 6.3.2a and 6.3.2b both cover a wide range of problems, e.g. l_2-regularized empirical risk minimization problems with convex losses.

Finally, as a special case of the strong convexity with $\mu = 0$, we state the general convexity assumption, which we will use for convergence analysis.

Assumption 6.3.3. $f(w;\xi)$ *is convex for every realization of* ξ, *i.e.,* $\forall w, w' \in \mathbb{R}^d$,

$$f(w;\xi) \geq f(w';\xi) + \nabla f(w';\xi)^T(w - w').$$

Again, we note that Assumption 6.3.2b implies Assumption 6.3.3, but Assumption 6.3.2a does not. Hence in our analysis, depending on the result we aim at, we will require Assumption 6.3.3 to hold by itself, or Assumption 6.3.2a and Assumption 6.3.3 to hold together, or Assumption 6.3.2b to hold by itself. We will always use Assumption 6.3.1.

6.3.2 Existing Results

In this section, we provide some well-known results from the existing literature that support our theoretical analyses.

Lemma 6.3.1 (Theorem 2.1.5 in [49]). *Suppose that f is convex and L-smooth. Then, for any w, $w' \in \mathbb{R}^d$,*

$$f(w) \leq f(w') + \nabla f(w')^T(w - w') + \frac{L}{2}\|w - w'\|^2, \tag{6.9}$$

$$f(w) \geq f(w') + \nabla f(w')^T(w - w') + \frac{1}{2L}\|\nabla f(w) - \nabla f(w')\|^2, \tag{6.10}$$

$$(\nabla f(w) - \nabla f(w'))^T(w - w') \geq \frac{1}{L}\|\nabla f(w) - \nabla f(w')\|^2. \tag{6.11}$$

Note that (6.9) does not require the convexity of f.

Lemma 6.3.2 (Theorem 2.1.11 in [49]). *Suppose that f is μ-strongly convex and L-smooth. Then, for any w, $w' \in \mathbb{R}^d$,*

$$(\nabla f(w) - \nabla f(w'))^T(w - w') \geq \frac{\mu L}{\mu + L}\|w - w'\|^2 + \frac{1}{\mu + L}\|\nabla f(w) - \nabla f(w')\|^2. \tag{6.12}$$

Lemma 6.3.3 ([28]). *Suppose that Assumptions 6.3.1 and 6.3.3 hold. Then, $\forall w \in \mathbb{R}^d$,*

$$\mathbb{E}[\|\nabla f(w;\xi) - \nabla f(w_*;\xi)\|^2] \leq 2L[F(w) - F(w_*)], \tag{6.13}$$

where w_* is any optimal solution of $F(w)$.

Lemma 6.3.4 (Lemma 1 in [53]). *Suppose that Assumptions 6.3.1 and 6.3.3 hold. Then, for $\forall w \in \mathbb{R}^d$,*

$$\mathbb{E}[\|\nabla f(w;\xi)\|^2] \leq 4L[F(w) - F(w_*)] + 2\mathbb{E}[\|\nabla f(w_*;\xi)\|^2], \tag{6.14}$$

where w_* is any optimal solution of $F(w)$.

Lemma 6.3.5 (Lemma 1 in [52]). *Let ξ and $\{\xi_i\}_{i=1}^b$ be i.i.d. random variables with*

$\mathbb{E}[\nabla f(w;\xi_i)] = \nabla F(w)$, $i = 1, \ldots, b$, *for all* $w \in \mathbb{R}^d$. *Then,*

$$\mathbb{E}\left[\left\|\frac{1}{b}\sum_{i=1}^{b}\nabla f(w;\xi_i) - \nabla F(w)\right\|^2\right] = \frac{\mathbb{E}[\|\nabla f(w;\xi)\|^2] - \|\nabla F(w)\|^2}{b}. \tag{6.15}$$

Lemmata 6.3.4 and 6.3.5 clearly imply the following result.

Corollary 6.3.1. *Suppose that Assumptions 6.3.1 and 6.3.3 hold. Let* $\{\xi_i\}_{i=1}^b$ *be i.i.d. random variables with* $\mathbb{E}[\nabla f(w;\xi_i)] = \nabla F(w)$, $i = 1, \ldots, b$, *for all* $w \in \mathbb{R}^d$. *Then,*

$$\mathbb{E}\left[\left\|\frac{1}{b}\sum_{i=1}^{b}\nabla f(w;\xi_i) - \nabla F(w)\right\|^2\right] \leq \frac{4L[F(w) - F(w_*)] + 2\mathbb{E}[\|\nabla f(w_*;\xi)\|^2] - \|\nabla F(w)\|^2}{b}, \tag{6.16}$$

where w_* *is any optimal solution of* $F(w)$.

6.3.3 Special Property of SARAH Update

The most important property of the SVRG algorithm is the variance reduction of the steps. This property holds as the number of outer iteration grows, but it does not hold, if only the number of inner iterations increases. In other words, if we simply run the inner loop for many iterations (without executing additional outer loops), the variance of the steps does not reduce in the case of SVRG, while it goes to zero in the case of SARAH with large learning rate in the strongly convex case. We recall the SARAH update as follows.

$$v_t = \nabla f(w_t;\xi_t) - \nabla f(w_{t-1};\xi_t) + v_{t-1}, \tag{6.17}$$

followed by the iterate update:

$$w_{t+1} = w_t - \eta v_t. \tag{6.18}$$

We will now show that $\|v_t\|^2$ is going to zero in expectation in the *strongly convex* case. These results substantiate our conclusion that SARAH uses more stable stochastic gradient estimates than SVRG.

Theorem 6.3.1a. *Suppose that Assumptions 6.3.1, 6.3.2a and 6.3.3 hold. Consider* v_t

defined by (6.17) *with* $\eta < 2/L$ *and any given* v_0. *Then, for any* $t \geq 1$,

$$\begin{aligned}\mathbb{E}[\|v_t\|^2] &\leq \left[1 - \left(\tfrac{2}{\eta L} - 1\right)\mu^2\eta^2\right]\mathbb{E}[\|v_{t-1}\|^2] \\ &\leq \left[1 - \left(\tfrac{2}{\eta L} - 1\right)\mu^2\eta^2\right]^t \|v_0\|^2.\end{aligned}$$

Proof. For $t \geq 1$, we have

$$\begin{aligned}\|\nabla F(w_t) - \nabla F(w_{t-1})\|^2 &= \Big\|\mathbb{E}[\nabla f(w_t;\xi_t) - \nabla f(w_{t-1};\xi_t)|\mathcal{F}_t]\Big\|^2 \\ &\leq \mathbb{E}[\|\nabla f(w_t;\xi_t) - \nabla f(w_{t-1};\xi_t)\|^2|\mathcal{F}_t]. \qquad (6.19)\end{aligned}$$

For $t \geq 1$, we have

$$\begin{aligned}&\mathbb{E}[\|v_t\|^2|\mathcal{F}_t] \\ &= \mathbb{E}[\|v_{t-1} - (\nabla f(w_{t-1};\xi_t) - \nabla f(w_t;\xi_t)\|^2|\mathcal{F}_t] \\ &= \|v_{t-1}\|^2 \\ &\quad + \mathbb{E}\Big[\|\nabla f(w_{t-1};\xi_t) - \nabla f(w_t;\xi_t)\|^2 - \tfrac{2}{\eta}(\nabla f(w_{t-1};\xi_t) - \nabla f(w_t;\xi_t))^T(w_{t-1} - w_t)|\mathcal{F}_t\Big] \\ &\overset{(6.11)}{\leq} \|v_{t-1}\|^2 + \mathbb{E}\Big[\|\nabla f(w_{t-1};\xi_t) - \nabla f(w_t;\xi_t)\|^2 - \tfrac{2}{L\eta}\|\nabla f(w_{t-1};\xi_t) - \nabla f(w_t;\xi_t)\|^2|\mathcal{F}_t\Big] \\ &= \|v_{t-1}\|^2 - \left(\tfrac{2}{\eta L} - 1\right)\mathbb{E}[\|\nabla f(w_{t-1};\xi_t) - \nabla f(w_t;\xi_t)\|^2|\mathcal{F}_t] \\ &\overset{(6.19)}{\leq} \|v_{t-1}\|^2 - \left(\tfrac{2}{\eta L} - 1\right)\|\nabla F(w_t) - \nabla F(w_{t-1})\|^2 \\ &\leq \|v_{t-1}\|^2 - \left(\tfrac{2}{\eta L} - 1\right)\mu^2\eta^2\|v_{t-1}\|^2.\end{aligned}$$

Note that $\frac{2}{\eta L} - 1 > 0$ since $\eta < 2/L$. The last inequality follows by the strong convexity of F, that is, $\mu\|w_t - w_{t-1}\| \leq \|\nabla F(w_t) - \nabla F(w_{t-1})\|$ and the fact that $w_t = w_{t-1} - \eta v_{t-1}$. By taking the expectation and applying recursively, we have

$$\begin{aligned}\mathbb{E}[\|v_t\|^2] &\leq \left[1 - \left(\tfrac{2}{\eta L} - 1\right)\mu^2\eta^2\right]\mathbb{E}[\|v_{t-1}\|^2] \\ &\leq \left[1 - \left(\tfrac{2}{\eta L} - 1\right)\mu^2\eta^2\right]^t \mathbb{E}[\|v_0\|^2].\end{aligned}$$

For any given v_0, we achieve the desired result. □

This result implies that by choosing $\eta = \mathcal{O}(1/L)$, we obtain the linear convergence of $\|v_t\|^2$ in expectation with the rate $(1 - 1/\kappa^2)$. Below we show that a better convergence rate can be obtained under a stronger convexity assumption.

Theorem 6.3.1b. *Suppose that Assumptions 6.3.1 and 6.3.2b hold. Consider v_t defined by (6.17) with $\eta \leq 2/(\mu + L)$ and any given v_0. Then the following bound holds, $\forall\ t \geq 1$,*

$$\begin{aligned}\mathbb{E}[\|v_t\|^2] &\leq \left(1 - \tfrac{2\mu L\eta}{\mu+L}\right) \mathbb{E}[\|v_{t-1}\|^2] \\ &\leq \left(1 - \tfrac{2\mu L\eta}{\mu+L}\right)^t \|v_0\|^2.\end{aligned}$$

Proof. For $t \geq 1$, we have

$$\begin{aligned}
&\mathbb{E}[\|v_t\|^2|\mathcal{F}_t] \\
&\leq \|v_{t-1}\|^2 \\
&\quad + \mathbb{E}\Big[\|\nabla f(w_{t-1};\xi_t) - \nabla f(w_t;\xi_t)\|^2 - \tfrac{2}{\eta}(\nabla f(w_{t-1};\xi_t) - \nabla f(w_t;\xi_t))^T(w_{t-1} - w_t)|\mathcal{F}_t\Big] \\
&\overset{(6.12)}{\leq} \|v_{t-1}\|^2 - \tfrac{2\mu L\eta}{\mu+L}\|v_{t-1}\|^2 + \mathbb{E}[\|\nabla f(w_{t-1};\xi_t) - \nabla f(w_t;\xi_t)\|^2|\mathcal{F}_t] \\
&\quad - \tfrac{2}{\eta}\cdot\tfrac{1}{\mu+L}\mathbb{E}[\|\nabla f(w_{t-1};\xi_t) - \nabla f(w_t;\xi_t)\|^2|\mathcal{F}_t] \\
&= (1 - \tfrac{2\mu L\eta}{\mu+L})\|v_{t-1}\|^2 + (1 - \tfrac{2}{\eta}\cdot\tfrac{1}{\mu+L})\mathbb{E}[\|\nabla f(w_{t-1};\xi_t) - \nabla f(w_t;\xi_t)\|^2|\mathcal{F}_t] \\
&\leq \left(1 - \tfrac{2\mu L\eta}{\mu+L}\right)\|v_{t-1}\|^2, \qquad (6.20)
\end{aligned}$$

where in last inequality we have used that $\eta \leq 2/(\mu + L)$. By taking the expectation and applying recursively for any given v_0, the desired result is achieved. □

Again, by setting $\eta = \mathcal{O}(1/L)$, we derive the linear convergence with the rate of $(1-1/\kappa)$, which is a significant improvement over the result of Theorem 6.3.1a, when the problem is severely ill-conditioned.

We will provide our convergence analyses in detail in next sub-section. We will divide our results into two parts, which are *one-loop* results corresponding to iSARAH-IN (Algorithm 7) and *multiple-loop* results corresponding to iSARAH (Algorithm 6).

6.3.4 One-loop (iSARAH-IN) Results

We begin with providing two useful lemmata that do not require convexity assumption. Lemma 6.3.6 bounds the sum of expected of $\|\nabla F(w_t)\|^2$; and Lemma 6.3.7 expands the value of $\mathbb{E}[\|\nabla F(w_t) - v_t\|^2]$.

Lemma 6.3.6. *Suppose that Assumption 6.3.1 holds. Consider iSARAH-IN (Algorithm 7). Then, we have*

$$\sum_{t=0}^{m} \mathbb{E}[\|\nabla F(w_t)\|^2] \leq \frac{2}{\eta}\mathbb{E}[F(w_0) - F(w_*)] + \sum_{t=0}^{m} \mathbb{E}[\|\nabla F(w_t) - v_t\|^2] - (1 - L\eta) \sum_{t=0}^{m} \mathbb{E}[\|v_t\|^2], \tag{6.21}$$

where $w_* = \arg\min_w F(w)$.

Proof. By Assumption 6.3.1 and $w_{t+1} = w_t - \eta v_t$, we have

$$\begin{aligned}\mathbb{E}[F(w_{t+1})] &\overset{(6.9)}{\leq} \mathbb{E}[F(w_t)] - \eta\mathbb{E}[\nabla F(w_t)^T v_t] + \frac{L\eta^2}{2}\mathbb{E}[\|v_t\|^2] \\ &= \mathbb{E}[F(w_t)] - \frac{\eta}{2}\mathbb{E}[\|\nabla F(w_t)\|^2] + \frac{\eta}{2}\mathbb{E}[\|\nabla F(w_t) - v_t\|^2] - \left(\frac{\eta}{2} - \frac{L\eta^2}{2}\right)\mathbb{E}[\|v_t\|^2],\end{aligned}$$

where the last equality follows from the fact $a^T b = \frac{1}{2}\left[\|a\|^2 + \|b\|^2 - \|a - b\|^2\right]$.

By summing over $t = 0, \ldots, m$, we have

$$\begin{aligned}\mathbb{E}[F(w_{m+1})] \leq \mathbb{E}[F(w_0)] - \frac{\eta}{2}\sum_{t=0}^{m} \mathbb{E}[\|\nabla F(w_t)\|^2] + \frac{\eta}{2}\sum_{t=0}^{m} \mathbb{E}[\|\nabla F(w_t) - v_t\|^2] \\ - \left(\frac{\eta}{2} - \frac{L\eta^2}{2}\right)\sum_{t=0}^{m} \mathbb{E}[\|v_t\|^2],\end{aligned}$$

which is equivalent to ($\eta > 0$):

$$\begin{aligned}\sum_{t=0}^{m} \mathbb{E}[\|\nabla F(w_t)\|^2] &\leq \frac{2}{\eta}\mathbb{E}[F(w_0) - F(w_m)] + \sum_{t=0}^{m} \mathbb{E}[\|\nabla F(w_t) - v_t\|^2] - (1 - L\eta)\sum_{t=0}^{m} \mathbb{E}[\|v_t\|^2] \\ &\leq \frac{2}{\eta}\mathbb{E}[F(w_0) - F(w_*)] + \sum_{t=0}^{m} \mathbb{E}[\|\nabla F(w_t) - v_t\|^2] - (1 - L\eta)\sum_{t=0}^{m} \mathbb{E}[\|v_t\|^2],\end{aligned}$$

where the last inequality follows since $w_* = \arg\min_w F(w)$. □

Lemma 6.3.7. *Suppose that Assumption 6.3.1 holds. Consider v_t defined by* (6.4) *in iSARAH-IN (Algorithm 7). Then for any $t \geq 1$,*

$$\begin{aligned}
&\mathbb{E}[\|\nabla F(w_t) - v_t\|^2] \\
&= \mathbb{E}[\|\nabla F(w_0) - v_0\|^2] + \sum_{j=1}^{t} \mathbb{E}[\|v_j - v_{j-1}\|^2] - \sum_{j=1}^{t} \mathbb{E}[\|\nabla F(w_j) - \nabla F(w_{j-1})\|^2].
\end{aligned}$$

Proof. Let $\mathcal{F}_j = \sigma(w_0, w_1, \ldots, w_j)$ be the σ-algebra generated by $w_0, w_1, \ldots, w_j$[2]. We note that ξ_j is independent of $\mathcal{F}_j$. For $j \geq 1$, we have

$$\begin{aligned}
&\mathbb{E}[\|\nabla F(w_j) - v_j\|^2 | \mathcal{F}_j] \\
&= \mathbb{E}[\|[\nabla F(w_{j-1}) - v_{j-1}] + [\nabla F(w_j) - \nabla F(w_{j-1})] - [v_j - v_{j-1}]\|^2 | \mathcal{F}_j] \\
&= \|\nabla F(w_{j-1}) - v_{j-1}\|^2 + \|\nabla F(w_j) - \nabla F(w_{j-1})\|^2 + \mathbb{E}[\|v_j - v_{j-1}\|^2 | \mathcal{F}_j] \\
&\quad + 2(\nabla F(w_{j-1}) - v_{j-1})^T (\nabla F(w_j) - \nabla F(w_{j-1})) \\
&\quad - 2(\nabla F(w_{j-1}) - v_{j-1})^T \mathbb{E}[v_j - v_{j-1} | \mathcal{F}_j] \\
&\quad - 2(\nabla F(w_j) - \nabla F(w_{j-1}))^T \mathbb{E}[v_j - v_{j-1} | \mathcal{F}_j] \\
&= \|\nabla F(w_{j-1}) - v_{j-1}\|^2 - \|\nabla F(w_j) - \nabla F(w_{j-1})\|^2 + \mathbb{E}[\|v_j - v_{j-1}\|^2 | \mathcal{F}_j],
\end{aligned}$$

where the last equality follows from

$$\mathbb{E}[v_j - v_{j-1} | \mathcal{F}_j] \overset{(6.4)}{=} \mathbb{E}[\nabla f(w_j; \xi_j) - \nabla f(w_{j-1}; \xi_j) | \mathcal{F}_j] = \nabla F(w_j) - \nabla F(w_{j-1}).$$

By taking expectation for the above equation, we have

$$\begin{aligned}
&\mathbb{E}[\|\nabla F(w_j) - v_j\|^2] \\
&= \mathbb{E}[\|\nabla F(w_{j-1}) - v_{j-1}\|^2] - \mathbb{E}[\|\nabla F(w_j) - \nabla F(w_{j-1})\|^2] + \mathbb{E}[\|v_j - v_{j-1}\|^2].
\end{aligned}$$

By summing over $j = 1, \ldots, t$ ($t \geq 1$), we have

$$\mathbb{E}[\|\nabla F(w_t) - v_t\|^2]$$

[2] $\mathcal{F}_j$ contains all the information of $w_0, \ldots, w_j$ as well as $v_0, \ldots, v_{j-1}$

$$= \mathbb{E}[\|\nabla F(w_0) - v_0\|^2] + \sum_{j=1}^{t} \mathbb{E}[\|v_j - v_{j-1}\|^2] - \sum_{j=1}^{t} \mathbb{E}[\|\nabla F(w_j) - \nabla F(w_{j-1})\|^2].$$

□

General Convex Cases

In this subsection, we analyze one-loop results of Inexact SARAH (Algorithm 7) in the general convex case. We first derive the bound for $\mathbb{E}[\|\nabla F(w_t) - v_t\|^2]$.

Lemma 6.3.8. *Suppose that Assumptions 6.3.1 and 6.3.3 hold. Consider v_t defined as* (6.4) *in SARAH (Algorithm 6) with $\eta < 2/L$. Then we have that for any $t \geq 1$,*

$$\mathbb{E}[\|\nabla F(w_t) - v_t\|^2] \leq \frac{\eta L}{2 - \eta L}\Big[\mathbb{E}[\|v_0\|^2] - \mathbb{E}[\|v_t\|^2]\Big] + \mathbb{E}[\|\nabla F(w_0) - v_0\|^2]. \tag{6.22}$$

Proof. For $j \geq 1$, we have

$$\begin{aligned}
&\mathbb{E}[\|v_j\|^2|\mathcal{F}_j] \\
&= \mathbb{E}[\|v_{j-1} - (\nabla f(w_{j-1};\xi_j) - \nabla f(w_j;\xi_j)\|^2|\mathcal{F}_j] \\
&= \|v_{j-1}\|^2 \\
&\quad + \mathbb{E}\Big[\|\nabla f(w_{j-1};\xi_j) - \nabla f(w_j;\xi_j)\|^2 - \tfrac{2}{\eta}(\nabla f(w_{j-1};\xi_j) - \nabla f(w_j;\xi_j))^T(w_{j-1} - w_j)|\mathcal{F}_j\Big] \\
&\overset{(6.11)}{\leq} \|v_{j-1}\|^2 + \mathbb{E}\Big[\|\nabla f(w_{j-1};\xi_j) - \nabla f(w_j;\xi_j)\|^2 - \tfrac{2}{L\eta}\|\nabla f(w_{j-1};\xi_j) - \nabla f(w_j;\xi_j)\|^2|\mathcal{F}_j\Big] \\
&= \|v_{j-1}\|^2 + \Big(1 - \tfrac{2}{\eta L}\Big)\mathbb{E}[\|\nabla f(w_{j-1};\xi_j) - \nabla f(w_j;\xi_j)\|^2|\mathcal{F}_j] \\
&\overset{(6.4)}{=} \|v_{j-1}\|^2 + \Big(1 - \tfrac{2}{\eta L}\Big)\mathbb{E}[\|v_j - v_{j-1}\|^2|\mathcal{F}_j],
\end{aligned}$$

which, if we take expectation, implies that

$$\mathbb{E}[\|v_j - v_{j-1}\|^2] \leq \frac{\eta L}{2 - \eta L}\Big[\mathbb{E}[\|v_{j-1}\|^2] - \mathbb{E}[\|v_j\|^2]\Big],$$

when $\eta < 2/L$.

By summing the above inequality over $j = 1, \ldots, t$ ($t \geq 1$), we have

$$\sum_{j=1}^{t} \mathbb{E}[\|v_j - v_{j-1}\|^2] \leq \frac{\eta L}{2 - \eta L} \Big[\mathbb{E}[\|v_0\|^2] - \mathbb{E}[\|v_t\|^2] \Big]. \tag{6.23}$$

By Lemma 6.3.7, we have

$$\begin{aligned} \mathbb{E}[\|\nabla F(w_t) - v_t\|^2] &\leq \sum_{j=1}^{t} \mathbb{E}[\|v_j - v_{j-1}\|^2] + \mathbb{E}[\|\nabla F(w_0) - v_0\|^2] \\ &\overset{(6.23)}{\leq} \frac{\eta L}{2 - \eta L} \Big[\mathbb{E}[\|v_0\|^2] - \mathbb{E}[\|v_t\|^2] \Big] + \mathbb{E}[\|\nabla F(w_0) - v_0\|^2]. \end{aligned}$$

□

Lemma 6.3.9. *Suppose that Assumptions 6.3.1 and 6.3.3 hold. Consider v_0 defined as* (6.3) *in SARAH (Algorithm 6). Then we have,*

$$\begin{aligned} &\frac{\eta L}{2 - \eta L} \mathbb{E}[\|v_0\|^2] + \mathbb{E}[\|\nabla F(w_0) - v_0\|^2] \\ &\qquad \leq \frac{2}{2 - \eta L} \left(\frac{4L\mathbb{E}[F(w_0) - F(w_*)] + 2\mathbb{E}\left[\|\nabla f(w_*; \xi)\|^2\right] - \mathbb{E}[\|\nabla F(w_0)\|^2]}{b} \right) \\ &\qquad\quad + \frac{\eta L}{2 - \eta L} \mathbb{E}[\|\nabla F(w_0)\|^2]. \end{aligned} \tag{6.24}$$

Proof. By Corollary 6.3.1, we have

$$\begin{aligned} &\frac{\eta L}{2 - \eta L} \mathbb{E}[\|v_0\|^2 | w_0] - \frac{\eta L}{2 - \eta L} \|\nabla F(w_0)\|^2 + \mathbb{E}[\|\nabla F(w_0) - v_0\|^2 | w_0] \\ &\qquad = \frac{2}{2 - \eta L} \Big[\mathbb{E}[\|v_0\|^2 | w_0] - \|\nabla F(w_0)\|^2 \Big] \\ &\qquad = \frac{2}{2 - \eta L} \Big[\mathbb{E}[\|v_0 - \nabla F(w_0)\|^2 | w_0 \Big] \\ &\qquad \overset{(6.16)}{\leq} \frac{2}{2 - \eta L} \left(\frac{4L[F(w_0) - F(w_*)] + 2\mathbb{E}\left[\|\nabla f(w_*; \xi)\|^2\right] - \|\nabla F(w_0)\|^2}{b} \right). \end{aligned}$$

Taking the expectation and adding $\frac{\eta L}{2 - \eta L} \mathbb{E}[\|\nabla F(w_0)\|^2]$ for both sides, the desired result is achieved. □

We then derive this basic result for the convex case by using Lemmata 6.3.8 and 6.3.9.

Lemma 6.3.10. *Suppose that Assumptions 6.3.1 and 6.3.3 hold. Consider iSARAH-IN (Algorithm 7) with $\eta \leq 1/L$. Then, we have*

$$\mathbb{E}[\|\nabla F(\tilde{w})\|^2] \leq \frac{2}{\eta(m+1)}\mathbb{E}[F(w_0) - F(w_*)] + \frac{\eta L}{2-\eta L}\mathbb{E}[\|\nabla F(w_0)\|^2] + \frac{2}{2-\eta L}\left(\frac{4L\mathbb{E}[F(w_0) - F(w_*)] + 2\mathbb{E}\left[\|\nabla f(w_*;\xi)\|^2\right] - \mathbb{E}[\|\nabla F(w_0)\|^2]}{b}\right), \tag{6.25}$$

where w_* is any optimal solution of $F(w)$; and ξ is some random variable.

Proof. By Lemma 6.3.8, we have

$$\sum_{t=0}^{m}\mathbb{E}[\|\nabla F(w_t) - v_t\|^2] \leq \frac{m\eta L}{2-\eta L}\mathbb{E}[\|v_0\|^2] + (m+1)\mathbb{E}[\|\nabla F(w_0) - v_0\|^2]. \tag{6.26}$$

Hence, by Lemma 6.3.6 with $\eta \leq 1/L$, we have

$$\begin{aligned}\sum_{t=0}^{m}\mathbb{E}[\|\nabla F(w_t)\|^2] &\leq \frac{2}{\eta}\mathbb{E}[F(w_0) - F(w_*)] + \sum_{t=0}^{m}\mathbb{E}[\|\nabla F(w_t) - v_t\|^2] \\ &\overset{(6.26)}{\leq} \frac{2}{\eta}\mathbb{E}[F(w_0) - F(w_*)] + \frac{m\eta L}{2-\eta L}\mathbb{E}[\|v_0\|^2] + (m+1)\mathbb{E}[\|\nabla F(w_0) - v_0\|^2].\end{aligned} \tag{6.27}$$

Since $\tilde{w} = w_t$, where t is picked uniformly at random from $\{0, 1, \ldots, m\}$. The following holds,

$$\begin{aligned}\mathbb{E}[\|\nabla F(\tilde{w})\|^2] &= \frac{1}{m+1}\sum_{t=0}^{m}\mathbb{E}[\|\nabla F(w_t)\|^2] \\ &\overset{(6.27)}{\leq} \frac{2}{\eta(m+1)}\mathbb{E}[F(w_0) - F(w_*)] + \frac{\eta L}{2-\eta L}\mathbb{E}[\|v_0\|^2] + \mathbb{E}[\|\nabla F(w_0) - v_0\|^2] \\ &\overset{(6.24)}{\leq} \frac{2}{\eta(m+1)}\mathbb{E}[F(w_0) - F(w_*)] + \frac{\eta L}{2-\eta L}\mathbb{E}[\|\nabla F(w_0)\|^2] \\ &\quad + \frac{2}{2-\eta L}\left(\frac{4L\mathbb{E}[F(w_0) - F(w_*)] + 2\mathbb{E}\left[\|\nabla f(w_*;\xi)\|^2\right] - \mathbb{E}[\|\nabla F(w_0)\|^2]}{b}\right).\end{aligned}$$

□

This expected bound for $\|\nabla F(\tilde{w})\|^2$ will be used for deriving both one-loop and multiple-

loop results in the convex case.

Lemma 6.3.10 can be used to get the following result for general convex.

Theorem 6.3.2. *Suppose that Assumptions 6.3.1 and 6.3.3 hold. Consider iSARAH-IN (Algorithm 7) with* $\eta = \frac{1}{L\sqrt{m+1}} \leq \frac{1}{L}$, $b = 2\sqrt{m+1}$ *and a given* w_0. *Then we have,*

$$\mathbb{E}[\|\nabla F(\tilde{w})\|^2] \leq \frac{2}{\eta(m+1)}[F(w_0) - F(w_*)] + \frac{1}{\sqrt{m+1}}\left[4L[F(w_0) - F(w_*)] + 2\mathbb{E}[\|\nabla f(w_*;\xi)\|^2]\right], \tag{6.28}$$

where w_* *is any optimal solution of* $F(w)$; *and* ξ *is some random variable.*

Proof. By Lemma 6.3.10, for any given w_0, we have

$$\begin{aligned}
\mathbb{E}[\|\nabla F(\tilde{w})\|^2] &\leq \frac{2}{\eta(m+1)}[F(w_0) - F(w_*)] + \frac{\eta L}{2-\eta L}\|\nabla F(w_0)\|^2 \\
&\quad + \frac{2}{2-\eta L}\left(\frac{4L[F(w_0) - F(w_*)] + 2\mathbb{E}\left[\|\nabla f(w_*;\xi)\|^2\right] - \|\nabla F(w_0)\|^2}{b}\right) \\
&\leq \frac{2}{\eta(m+1)}[F(w_0) - F(w_*)] \\
&\quad + \frac{1}{2-\eta L}\frac{4L}{\sqrt{m+1}}[F(w_0) - F(w_*)] + \frac{2}{2-\eta L}\frac{\mathbb{E}\left[\|\nabla f(w_*;\xi)\|^2\right]}{\sqrt{m+1}} \\
&\leq \frac{2}{\eta(m+1)}[F(w_0) - F(w_*)] \\
&\quad + \frac{1}{\sqrt{m+1}}\left[4L[F(w_0) - F(w_*)] + 2\mathbb{E}[\|\nabla f(w_*;\xi)\|^2]\right].
\end{aligned}$$

The last inequality follows since $\eta \leq \frac{1}{L}$, which implies $\frac{1}{2-\eta L} \leq 1$. The second last inequality follows since $\eta \leq \frac{1}{L\sqrt{m+1}}$ and $b = 2\sqrt{m+1}$. □

Corollary 6.3.2. *Suppose that Assumptions 6.3.1 and 6.3.3 hold. Consider iSARAH-IN (Algorithm 7) with the learning rate* $\eta = \mathcal{O}\left(\frac{1}{\sqrt{m+1}}\right)$ *and the number of samples* $b = 2\sqrt{m+1}$, *where* m *is the total number of iterations, then* $\|\nabla F(\tilde{w})\|^2$ *converges sublinearly in expectation with a rate of* $\mathcal{O}\left(\sqrt{\frac{1}{m+1}}\right)$, *and therefore, the total complexity to achieve an* ϵ*-accurate solution is* $\mathcal{O}(1/\epsilon^2)$.

Proof. It is easy to see that to achieve $\mathbb{E}[\|\nabla F(\tilde{w})\|^2] \leq \epsilon$ we need $m = \mathcal{O}(1/\epsilon^2)$ and hence the total work is $2\sqrt{m} + 2m = \mathcal{O}\left(\frac{1}{\epsilon} + \frac{1}{\epsilon^2}\right) = \mathcal{O}\left(\frac{1}{\epsilon^2}\right)$. □

Nonconvex Cases

We then move to nonconvex case. We start with some lemmata for bounding $\mathbb{E}[\|\nabla F(w_t) - v_t\|^2]$.

Lemma 6.3.11. *Suppose that Assumption 6.3.1 holds. Consider v_t defined as* (6.4) *in iSARAH-IN (Algorithm 7). Then for any $t \geq 1$,*

$$\mathbb{E}[\|\nabla F(w_t) - v_t\|^2] \leq \mathbb{E}[\|\nabla F(w_0) - v_0\|^2] + L^2\eta^2 \sum_{j=1}^{t} \mathbb{E}[\|v_{j-1}\|^2]. \tag{6.29}$$

Proof. We have, for $t \geq 1$,

$$\|v_t - v_{t-1}\|^2 \overset{(6.4)}{=} \|\nabla f(w_t;\xi_t) - \nabla f(w_{t-1};\xi_t)\|^2 \overset{(6.7)}{\leq} L^2\|w_t - w_{t-1}\|^2 = L^2\eta^2\|v_{t-1}\|^2. \tag{6.30}$$

Hence, by Lemma 6.3.7,

$$\begin{aligned}\mathbb{E}[\|\nabla F(w_t) - v_t\|^2] &\leq \mathbb{E}[\|\nabla F(w_0) - v_0\|^2] + \sum_{j=1}^{t} \mathbb{E}[\|v_j - v_{j-1}\|^2] \\ &\overset{(6.30)}{\leq} \mathbb{E}[\|\nabla F(w_0) - v_0\|^2] + L^2\eta^2 \sum_{j=1}^{t} \mathbb{E}[\|v_{j-1}\|^2].\end{aligned}$$

□

Lemma 6.3.12. *Suppose that Assumption 6.3.1 holds. Consider v_t defined as* (6.4) *in iSARAH-IN (Algorithm 7) with $\eta \leq \frac{2}{L(\sqrt{1+4m}+1)}$. Then we have*

$$L^2\eta^2 \sum_{t=0}^{m}\sum_{j=1}^{t} \mathbb{E}[\|v_{j-1}\|^2] - (1 - L\eta)\sum_{t=0}^{m} \mathbb{E}[\|v_t\|^2] \leq 0. \tag{6.31}$$

Proof. For $\eta \leq \frac{2}{L(\sqrt{1+4m}+1)}$, we have

$$\begin{aligned}&L^2\eta^2 \sum_{t=0}^{m}\sum_{j=1}^{t} \mathbb{E}[\|v_{j-1}\|^2] - (1 - L\eta)\sum_{t=0}^{m} \mathbb{E}[\|v_t\|^2] \\ &\qquad = L^2\eta^2\Big[m\mathbb{E}\|v_0\|^2 + (m-1)\mathbb{E}\|v_1\|^2 + \cdots + \mathbb{E}\|v_{m-1}\|^2\Big] \\ &\qquad\quad - (1 - L\eta)\Big[\mathbb{E}\|v_0\|^2 + \mathbb{E}\|v_1\|^2 + \cdots + \mathbb{E}\|v_m\|^2\Big]\end{aligned}$$

$$\leq [L^2\eta^2 m - (1 - L\eta)] \sum_{t=1}^{m} \mathbb{E}[\|v_{t-1}\|^2] \leq 0,$$

since $\eta = \frac{2}{L(\sqrt{1+4m}+1)}$ is a root of the equation $L^2\eta^2 m - (1 - L\eta) = 0$. □

With the help of the above lemmata, we are able to derive our result for nonconvex.

Theorem 6.3.3. *Suppose that Assumption 6.3.1 holds. Consider iSARAH-IN (Algorithm 7) with* $\eta \leq \frac{2}{L(\sqrt{1+4m}+1)} \leq \frac{1}{L}$, $b = \sqrt{m+1}$ *and a given* w_0. *Then we have,*

$$\mathbb{E}[\|\nabla F(\tilde{w})\|^2] \leq \frac{2}{\eta(m+1)}[F(w_0) - F^*] + \frac{1}{\sqrt{m+1}}\Big(\mathbb{E}[\|\nabla f(w_0;\xi)\|^2]\Big), \tag{6.32}$$

where F^* *is any lower bound of* F*; and* ξ *is some random variable.*

Proof. Let F^* be any lower bound of F. By Lemma 6.3.6 and since $\tilde{w} = w_t$, where t is picked uniformly at random from $\{0, 1, \ldots, m\}$, we have

$$\begin{aligned}
\mathbb{E}[\|\nabla F(\tilde{w})\|^2] &= \frac{1}{m+1}\sum_{t=0}^{m} \mathbb{E}[\|\nabla F(w_t)\|^2] \\
&\leq \frac{2}{\eta(m+1)}\mathbb{E}[F(w_0) - F^*] + \frac{1}{m+1}\left(\sum_{t=0}^{m} \mathbb{E}[\|\nabla F(w_t) - v_t\|^2] - (1 - L\eta)\sum_{t=0}^{m} \mathbb{E}[\|v_t\|^2]\right) \\
&\overset{(6.29)}{\leq} \frac{2}{\eta(m+1)}\mathbb{E}[F(w_0) - F^*] + \mathbb{E}[\|\nabla F(w_0) - v_0\|^2] \\
&\qquad + \frac{1}{m+1}\left(L^2\eta^2 \sum_{t=0}^{m}\sum_{j=1}^{t} \mathbb{E}[\|v_{j-1}\|^2] - (1 - L\eta)\sum_{t=0}^{m} \mathbb{E}[\|v_t\|^2]\right) \\
&\overset{(6.31)}{\leq} \frac{2}{\eta(m+1)}\mathbb{E}[F(w_0) - F^*] + \mathbb{E}[\|\nabla F(w_0) - v_0\|^2] \\
&\overset{(6.15)}{\leq} \frac{2}{\eta(m+1)}\mathbb{E}[F(w_0) - F^*] + \frac{1}{b}\mathbb{E}[\|\nabla f(w_0;\xi)\|^2].
\end{aligned}$$

For any given w_0 and $b = \sqrt{m+1}$, we could achieve the desired result. □

Corollary 6.3.3. *Suppose that Assumption 6.3.1 holds. Consider iSARAH-IN (Algorithm 7) with the learning rate* $\eta = \mathcal{O}\left(\frac{1}{\sqrt{m+1}}\right)$ *and the number of samples* $b = \sqrt{m+1}$*, where* m *is the total number of iterations, then* $\|\nabla F(\tilde{w})\|^2$ *converges sublinearly in expectation with a rate of* $\mathcal{O}\left(\sqrt{\frac{1}{m+1}}\right)$*, and therefore, the total complexity to achieve an* ϵ*-accurate solution is* $\mathcal{O}(1/\epsilon^2)$.

Proof. Same as general convex case, to achieve $\mathbb{E}[\|\nabla F(\tilde{w})\|^2] \leq \epsilon$ we need $m = \mathcal{O}(1/\epsilon^2)$ and hence the total work is $\sqrt{m} + 2m = \mathcal{O}\left(\frac{1}{\epsilon} + \frac{1}{\epsilon^2}\right) = \mathcal{O}\left(\frac{1}{\epsilon^2}\right)$. □

6.3.5 Multiple-loop (iSARAH) Results

In this section, we analyze multiple-loop results of Inexact SARAH (Algorithm 6).

Strongly Convex Cases

Theorem 6.3.4. *Suppose that Assumptions 6.3.1, 6.3.2a and 6.3.3 hold. Consider iSARAH (Algorithm 6) with the choice of η, m, and b such that*

$$\alpha = \frac{1}{\mu\eta(m+1)} + \frac{\eta L}{2-\eta L} + \frac{4\kappa - 2}{b(2-\eta L)} < 1.$$

(Note that $\kappa = L/\mu$.) Then, we have

$$\mathbb{E}[\|\nabla F(\tilde{w}_s)\|^2] - \Delta \leq \alpha^s(\|\nabla F(\tilde{w}_0)\|^2 - \Delta), \tag{6.33}$$

where

$$\Delta = \frac{\delta}{1-\alpha} \text{ and } \delta = \frac{4}{b(2-\eta L)}\mathbb{E}\left[\|\nabla f(w_*;\xi)\|^2\right].$$

Proof. By Lemma 6.3.10, with $\tilde{w} = \tilde{w}_s$ and $w_0 = \tilde{w}_{s-1}$, we have

$$\begin{aligned}
&\mathbb{E}[\|\nabla F(\tilde{w}_s)\|^2] \\
&\leq \frac{2}{\eta(m+1)}\mathbb{E}[F(\tilde{w}_{s-1}) - F(w_*)] + \frac{\eta L}{2-\eta L}\mathbb{E}[\|\nabla F(\tilde{w}_{s-1})\|^2] \\
&+ \frac{2}{2-\eta L}\left(\frac{4L\mathbb{E}[F(\tilde{w}_{s-1}) - F(w_*)] + 2\mathbb{E}\left[\|\nabla f(w_*;\xi)\|^2\right] - \mathbb{E}[\|\nabla F(\tilde{w}_{s-1})\|^2]}{b}\right) \\
&\overset{(6.8)}{\leq} \left(\frac{1}{\mu\eta(m+1)} + \frac{\eta L}{2-\eta L} + \frac{4\kappa - 2}{b(2-\eta L)}\right)\mathbb{E}[\|\nabla F(\tilde{w}_{s-1})\|^2] \\
&\quad + \frac{4}{b(2-\eta L)}\mathbb{E}\left[\|\nabla f(w_*;\xi)\|^2\right] \qquad (6.34) \\
&= \alpha\mathbb{E}[\|\nabla F(\tilde{w}_{s-1})\|^2] + \delta \\
&\leq \alpha^s\|\nabla F(\tilde{w}_0)\|^2 + \alpha^{s-1}\delta + \cdots + \alpha\delta + \delta
\end{aligned}$$

$$
\begin{aligned}
&\leq \alpha^s \|\nabla F(\tilde{w}_0)\|^2 + \delta \frac{1-\alpha^s}{1-\alpha} \\
&= \alpha^s \|\nabla F(\tilde{w}_0)\|^2 + \Delta(1-\alpha^s) \\
&= \alpha^s (\|\nabla F(\tilde{w}_0)\|^2 - \Delta) + \Delta.
\end{aligned}
$$

By adding $-\Delta$ to both sides, we achieve the desired result. □

Corollary 6.3.4. *Let $\eta = \mathcal{O}\left(\frac{1}{L}\right)$, $m = \mathcal{O}(\kappa)$, $b = \mathcal{O}\left(\max\left\{\frac{1}{\epsilon}, \kappa\right\}\right)$ and $s = \mathcal{O}\left(\log\left(\frac{1}{\epsilon}\right)\right)$ in Theorem 6.3.4. Then, the total work complexity to achieve $\mathbb{E}[\|\nabla F(\tilde{w}_s)\|^2] \leq \epsilon$ is $\mathcal{O}\left(\left(\max\left\{\frac{1}{\epsilon}, \kappa\right\} + \kappa\right) \log\left(\frac{1}{\epsilon}\right)\right)$.*

Proof. For example, let $\eta = \frac{2}{5L}$, $m = 20\kappa - 1$, and $b = \max\left\{20\kappa - 10, \frac{20\mathbb{E}\left[\|\nabla f(w_*;\xi)\|^2\right]}{\epsilon}\right\}$. From (6.34), we have

$$
\begin{aligned}
\mathbb{E}[\|\nabla F(\tilde{w}_s)\|^2] &\leq \left(\frac{1}{8} + \frac{1}{4} + \frac{1}{8}\right) \mathbb{E}[\|\nabla F(\tilde{w}_{s-1})\|^2] + \frac{\epsilon}{8} \\
&\leq \frac{1}{2}\mathbb{E}[\|\nabla F(\tilde{w}_{s-1})\|^2] + \frac{\epsilon}{8} \\
&\leq \frac{1}{2^s}\|\nabla F(\tilde{w}_0)\|^2 + \frac{\epsilon}{4}.
\end{aligned}
$$

To guarantee that $\mathbb{E}[\|\nabla F(\tilde{w}_s)\|^2] \leq \epsilon$, it is sufficient to make $\frac{1}{2^s}\|\nabla F(\tilde{w}_0)\|^2 = \frac{3}{4}\epsilon$ or equivalently $s = \log\left(\frac{\|\nabla F(\tilde{w}_0)\|^2}{\frac{3}{4}\epsilon}\right)$. This implies the total complexity to achieve an ϵ-accuracy solution is $(b+m)s = \mathcal{O}\left(\left(\max\left\{\frac{1}{\epsilon}, \kappa\right\} + \kappa\right) \log\left(\frac{1}{\epsilon}\right)\right)$. □

General Convex Cases

Assumption 6.3.4. *Let $\tilde{w}_0, \ldots, \tilde{w}_s$ be the (outer) iterations of Algorithm 6. We assume that there exist $M_1 > 0$ and $N_1 > 0$ such that, $\forall k \geq 0$,*

$$
F(\tilde{w}_k) - F(w_*) \leq M_1 \|\nabla F(\tilde{w}_k)\|^2 + N_1. \tag{6.35}
$$

Theorem 6.3.5. *Suppose that Assumptions 6.3.1, 6.3.3, and 6.3.4 hold. Consider iSARAH (Algorithm 6) with the choice of η, m, and b such that*

$$
\alpha_c = \frac{2M_1}{\eta(m+1)} + \frac{\eta L}{2-\eta L} + \frac{8LM_1 - 1}{b(2-\eta L)} < 1.
$$

Then, we have

$$\mathbb{E}[\|\nabla F(\tilde{w}_s)\|^2] - \Delta_c \leq \alpha^s(\|\nabla F(\tilde{w}_0)\|^2 - \Delta_c), \tag{6.36}$$

where

$$\Delta_c = \frac{\delta_c}{1-\alpha_c} \text{ and } \delta_c = \frac{2N_1}{\eta(m+1)} + \frac{8LN_1}{b(2-\eta L)} + \frac{4\mathbb{E}\left[\|\nabla f(w_*;\xi)\|^2\right]}{b(2-\eta L)}.$$

Proof. By Lemma 6.3.10, with $\tilde{w} = \tilde{w}_s$ and $w_0 = \tilde{w}_{s-1}$, we have

$$\begin{aligned}
&\mathbb{E}[\|\nabla F(\tilde{w}_s)\|^2] \\
&\leq \frac{2}{\eta(m+1)}\mathbb{E}[F(\tilde{w}_{s-1}) - F(w_*)] + \frac{\eta L}{2-\eta L}\mathbb{E}[\|\nabla F(\tilde{w}_{s-1})\|^2] \\
&+ \frac{2}{2-\eta L}\left(\frac{4L\mathbb{E}[F(\tilde{w}_{s-1}) - F(w_*)] + 2\mathbb{E}\left[\|\nabla f(w_*;\xi)\|^2\right] - \mathbb{E}[\|\nabla F(\tilde{w}_{s-1})\|^2]}{b}\right) \\
&\overset{(6.35)}{\leq} \left(\frac{2M_1}{\eta(m+1)} + \frac{\eta L}{2-\eta L} + \frac{8LM_1 - 1}{b(2-\eta L)}\right)\mathbb{E}[\|\nabla F(\tilde{w}_{s-1})\|^2] \\
&\quad + \frac{2N_1}{\eta(m+1)} + \frac{8LN_1}{b(2-\eta L)} + \frac{4\mathbb{E}\left[\|\nabla f(w_*;\xi)\|^2\right]}{b(2-\eta L)} \\
&= \alpha_c\mathbb{E}[\|\nabla F(\tilde{w}_{s-1})\|^2] + \delta_c \\
&\leq \alpha_c^s(\|\nabla F(\tilde{w}_0)\|^2 - \Delta_c) + \Delta_c.
\end{aligned}$$

□

Applying the same procedure as strongly convex case above, we can achieve the following complexity result.

Corollary 6.3.5. *Let $\eta = \mathcal{O}\left(\frac{1}{L}\right)$, $m = \mathcal{O}\left(\frac{1}{\epsilon}\right)$, $b = \mathcal{O}\left(\frac{1}{\epsilon}\right)$ and $s = \mathcal{O}\left(\log\left(\frac{1}{\epsilon}\right)\right)$ in Theorem 6.3.5. Then, the total work complexity to achieve $\mathbb{E}[\|\nabla F(\tilde{w}_s)\|^2] \leq \epsilon$ is $(b+m)s = \mathcal{O}\left(\frac{1}{\epsilon}\log\left(\frac{1}{\epsilon}\right)\right)$.*

We can observe that, with the help of Assumption 6.3.4, iSARAH could achieve the best known complexity among stochastic methods (without involving of exact gradient or accelerated trick) in the general convex case.

Conclusion

We study a feedback-based agent invitation scheme for a model with randomly behaving agents and possible abandonment of customers and agents. This model is motivated by a variety of existing and emerging applications. We derived some sufficient local stability conditions, using the machinery of switched linear systems and common quadratic Lyapunov functions. Our simulation and numerical experiments show good overall performance of the feedback scheme, when the local stability conditions hold. They also suggest that, for our model, the local stability is in fact sufficient for the global stability of fluid limits.

We have demonstrated that based on the behavior of the stochastic gradient estimates at or near the stationary points, SGD with fixed step size converges with the same rate as full gradient descent of the variance reduction methods, until it reaches the accuracy where the variance in the stochastic gradient estimates starts to dominate and prevents further convergence.

We have provided the analysis of stochastic gradient algorithms with a diminishing step size in the strongly convex case under the condition of Lipschitz continuity of the individual function realizations, but without requiring any bounds on the stochastic gradients. We showed almost sure convergence of SGD and provided sublinear upper bounds for the expected convergence rate of a general recursion which includes Hogwild! for inconsistent reads and writes as a special case.

We propose the SARAH algorithm for solving finite-sum minimization problems. The linear convergence rate of SARAH is proven under a strong convexity assumption. We also prove a linear convergence rate in the strongly convex case for an inner loop of SARAH, a property that SVRG does not possess. Moreover, we provide a sublinear convergence rate

(to stationary points) for general convex and nonconvex functions. We also consider the SARAH algorithm with inexactness. Instead of computing a full gradient at each outer iteration, we only compute a subset of samples.

Bibliography

[1] Alekh Agarwal and Leon Bottou. A lower bound for the optimization of finite sums. In *ICML*, pages 78–86, 2015.

[2] Z. Aksin, M. Armony, and V. Mehrotra. The modern call center: a multi-disciplinary perspective on operations management research. *Production and Operations Management*, 16(6):665–688, 2007.

[3] Zeyuan Allen-Zhu. Katyusha: The First Direct Acceleration of Stochastic Gradient Methods. *Proceedings of the 49th Annual ACM on Symposium on Theory of Computing (to appear)*, 2017.

[4] Zeyuan Allen Zhu. Natasha: Faster non-convex stochastic optimization via strongly non-convex parameter. *arXiv preprint arXiv:1702.00763*, 2017.

[5] Zeyuan Allen-Zhu and Elad Hazan. Variance reduction for faster non-convex optimization. In *ICML*, pages 699–707, 2016.

[6] Zeyuan Allen-Zhu and Yang Yuan. Improved SVRG for Non-Strongly-Convex or Sum-of-Non-Convex Objectives. In *ICML*, pages 1080–1089, 2016.

[7] American Telemedicine Association. *Core Operational Guidelines for Telehealth Services Involving Provider-Patient Interactions*, 2014. http://www.americantelemed.org/docs/default-source/standards/core-operational-guidelines-for-telehealth-services.pdf?sfvrsn=6.

[8] Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imaging Sciences*, 2(1):183–202, 2009.

[9] S. Bengtson. Generating better results with crowdsourcing: Leverage a network of high-quality professionals for customer service. *White paper*, 2014.

[10] Dimitri P. Bertsekas. Incremental gradient, subgradient, and proximal methods for convex optimization: A survey, 2015.

[11] Léon Bottou. Online learning and stochastic approximations. In David Saad, editor, *Online Learning in Neural Networks*, pages 9–42. Cambridge University Press, New York, NY, USA, 1998.

[12] Leon Bottou and Olivier Bousquet. The tradeoffs of large scale learning. In *Proceedings of the 20th International Conference on Neural Information Processing Systems*, NIPS'07, pages 161–168, USA, 2007. Curran Associates Inc.

[13] Léon Bottou, Frank E Curtis, and Jorge Nocedal. Optimization methods for large-scale machine learning. *arXiv:1606.04838*, 2016.

[14] Andrew Cotter, Ohad Shamir, Nati Srebro, and Karthik Sridharan. Better mini-batch algorithms via accelerated gradient methods. In *NIPS*, pages 1647–1655, 2011.

[15] Christopher M De Sa, Ce Zhang, Kunle Olukotun, and Christopher Ré. Taming the wild: A unified analysis of hogwild-style algorithms. In *Advances in neural information processing systems*, pages 2674–2682, 2015.

[16] Aaron Defazio, Francis Bach, and Simon Lacoste-Julien. SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives. In *NIPS*, pages 1646–1654, 2014.

[17] Aaron Defazio, Justin Domke, and Tibério Caetano. A faster, permutable incremental gradient method for big data problems. In *ICML*, pages 1125–1133, 2014.

[18] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12:2121–2159, 2011.

[19] Stewart N. Ethier and Thomas G. Kurtz. *Markov Processes: Characterization and Convergence*. Wiley, 1986.

[20] P. Formisano. Flexibility for changing business needs: Improve customer service and drive more revenue with a virtual crowdsourcing solution. *White paper*, 2014.

[21] O. Garnet, A. Mandelbaum, and M. Reiman. Designing a call center with impatient customers. *Manufacturing and Service Operations Management*, 4(3):208–227, 2002.

[22] Saeed Ghadimi and Guanghui Lan. Stochastic first- and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4):2341–2368, 2013.

[23] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *AISTATS*, 2010.

[24] Itai Gurvich and Amy Ward. On the dynamic control of matching queues. *Stochastic Systems*, 4(2):479–523, 2014.

[25] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Series in Statistics, 2nd edition, 2009.

[26] Elad Hazan and Satyen Kale. Beyond the regret minimization barrier: Optimal algorithms for stochastic strongly-convex optimization. *Journal of Machine Learning Research*, 15:2489–2512, 2014.

[27] J. P. Hespanha. Uniform stability of switched linear systems: extensions of LaSalle's invariance principle. *IEEE Transactions on Automatic Control*, 49(4):470–482, 2004.

[28] Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *NIPS*, pages 315–323, 2013.

[29] B. R. K. Kashyap. The double-ended queue with bulk service and limited waiting space. *Operations Research*, 14(5):822–834, 1966.

[30] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

[31] Jakub Konečný, Jie Liu, Peter Richtárik, and Martin Takáč. Mini-batch semi-stochastic gradient descent in the proximal setting. *IEEE Journal of Selected Topics in Signal Processing*, 10:242–255, 2016.

[32] Jakub Konečný and Peter Richtárik. Semi-stochastic gradient descent methods. *arXiv:1312.1666*, 2013.

[33] A. Krizhevsky and G. Hinton. Learning multiple layers of features from tiny images. *Master's thesis, Department of Computer Science, University of Toronto*, 2009.

[34] Nicolas Le Roux, Mark Schmidt, and Francis Bach. A stochastic gradient method with an exponential convergence rate for finite training sets. In *NIPS*, pages 2663–2671, 2012.

[35] Remi Leblond, Fabian Pedregosa, and Simon Lacoste-Julien. Improved asynchronous parallel optimization analysis for stochastic incremental methods. *arXiv:1801.03749*, 2018.

[36] Yann Lecun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. In *Proceedings of the IEEE*, pages 2278–2324, 1998.

[37] Lihua Lei and Michael Jordan. Less than a Single Pass: Stochastically Controlled Stochastic Gradient. In Aarti Singh and Jerry Zhu, editors, *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pages 148–156, Fort Lauderdale, FL, USA, 20–22 Apr 2017. PMLR.

[38] Lihua Lei, Cheng Ju, Jianbo Chen, and Michael I Jordan. Non-convex finite-sum optimization via scsg methods. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 2348–2358. Curran Associates, Inc., 2017.

[39] Hai Lin and Panos J. Antsaklis. Stability and stabilizability of switched linear systems: A survey of recent results. *IEEE Transactions on Automatic Control*, 54(2):308–322, 2009.

[40] D. C. Liu and J. Nocedal. On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45:503–528, 1989.

[41] X. Liu, Q. Gong, and V. G. Kulkarni. Diffusion models for doubly-ended queues with renewal arrival processes. Forthcoming in *Stochastic Systems*. DOI: 10.1214/13-SSY113, 2014.

[42] Julien Mairal. Optimization with first-order surrogate functions. In *ICML*, pages 783–791, 2013.

[43] Julien Mairal. Optimization with first-order surrogate functions. In *ICML*, pages 783–791, 2013.

[44] Horia Mania, Xinghao Pan, Dimitris Papailiopoulos, Benjamin Recht, Kannan Ramchandran, and Michael I Jordan. Perturbed Iterate Analysis for Asynchronous Stochastic Optimization. *arXiv preprint arXiv:1507.06970*, 2015.

[45] S. McGee-Smith. Why companies are choosing to deploy the LiveOps cloud-based contact center, 2010. http://www.liveops.com/sites/default/files/uploads/lo_wp_mcgee-smith_analytics.pdf.

[46] Aryan Mokhtari, Mert Gürbüzbalaban, and Alejandro Ribeiro. A double incremental aggregated gradient method with linear convergence rate for large-scale optimization. *Proceedings of IEEE International Conference on Acoustic, Speech and Signal Processing (to appear)*, 2017.

[47] Eric Moulines and Francis R. Bach. Non-asymptotic analysis of stochastic approximation algorithms for machine learning. In J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. Pereira, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 451–459. Curran Associates, Inc., 2011.

[48] A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM J. on Optimization*, 19(4):1574–1609, January 2009.

[49] Yurii Nesterov. *Introductory lectures on convex optimization : a basic course*. Applied optimization. Kluwer Academic Publ., Boston, Dordrecht, London, 2004.

[50] Yurii Nesterov and Boris T Polyak. Cubic regularization of newton method and its global performance. *Mathematical Programming*, 108(1):177–205, 2006.

[51] Lam Nguyen, Jie Liu, Katya Scheinberg, and Martin Takáč. SARAH: A novel method for machine learning problems using stochastic recursive gradient. *ICML*, 2017.

[52] Lam Nguyen, Nam Nguyen, Dzung Phan, Jayant Kalagnanam, and Katya Scheinberg. When does stochastic gradient algorithm work well? *arXiv:1801.06159*, 2018.

[53] Lam Nguyen, Phuong Ha Nguyen, Marten van Dijk, Peter Richtárik, Katya Scheinberg, and Martin Takáč. SGD and Hogwild! convergence without the bounded gradients assumption. *arXiv:1802.03801*, 2018.

[54] Lam M. Nguyen, Jie Liu, Katya Scheinberg, and Martin Takác. Stochastic recursive gradient algorithm for nonconvex optimization. *CoRR*, abs/1705.07261, 2017.

[55] Lam M. Nguyen and Alexander L. Stolyar. A service system with randomly behaving on-demand agents. *SIGMETRICS Perform. Eval. Rev.*, 44(1):365–366, 2016.

[56] Lam M. Nguyen and Alexander L. Stolyar. A queueing system with on-demand servers: local stability of fluid limits. *Queueing Systems*, Nov 2017.

[57] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer, New York, 2nd edition, 2006.

[58] Guodong Pang and Alexander L. Stolyar. A service system with on-demand agent invitations. *Queueing Syst. Theory Appl.*, 82(3-4):259–283, 2016.

[59] Guodong Pang, Rishi Talreja, and Ward Whitt. Martingale proofs of many-server heavy-traffic limits for Markovian queues. *Probability Surveys*, 4:193–267, 2007.

[60] Boris T Polyak. Gradient methods for the minimisation of functionals. *USSR Computational Mathematics and Mathematical Physics*, 3(4):864–878, 1963.

[61] Boris T. Polyak. Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics*, 4(5):1–17, 1964.

[62] L.S. Pontryagin. *Ordinary Differential Equations*. Adiwes international series in mathematics. Addison-Wesley, 1962.

[63] Alexander Rakhlin, Ohad Shamir, and Karthik Sridharan. Making gradient descent optimal for strongly convex stochastic optimization. In *ICML*. icml.cc / Omnipress, 2012.

[64] Benjamin Recht, Christopher Re, Stephen Wright, and Feng Niu. Hogwild!: A Lock-Free Approach to Parallelizing Stochastic Gradient Descent. In J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. Pereira, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 693–701. Curran Associates, Inc., 2011.

[65] Sashank J. Reddi, Ahmed Hefny, Suvrit Sra, Barnabás Póczos, and Alexander J. Smola. Stochastic variance reduction for nonconvex optimization. In *ICML*, pages 314–323, 2016.

[66] Herbert Robbins and Sutton Monro. A stochastic approximation method. *The Annals of Mathematical Statistics*, 22(3):400–407, 1951.

[67] Walter Rudin. *Principles of Mathematical Analysis*. McGraw-Hill, New York, 1976.

[68] Mark Schmidt, Nicolas Le Roux, and Francis Bach. Minimizing finite sums with the stochastic average gradient. *Mathematical Programming*, pages 1–30, 2016.

[69] Shai Shalev-Shwartz, Yoram Singer, and Nathan Srebro. Pegasos: Primal estimated sub-gradient solver for svm. In *Proceedings of the 24th International Conference on Machine Learning*, ICML '07, pages 807–814, New York, NY, USA, 2007. ACM.

[70] Shai Shalev-Shwartz, Yoram Singer, and Nathan Srebro. Pegasos: Primal estimated sub-gradient solver for SVM. In *ICML*, pages 807–814, 2007.

[71] Shai Shalev-Shwartz, Yoram Singer, Nathan Srebro, and Andrew Cotter. Pegasos: Primal estimated sub-gradient solver for SVM. *Mathematical Programming*, 127(1):3–30, 2011.

[72] Shai Shalev-Shwartz and Tong Zhang. Stochastic dual coordinate ascent methods for regularized loss. *Journal of Machine Learning Research*, 14(1):567–599, 2013.

[73] R. Shorten, O. Mason, F. O'Cairbre, and P. Curran. A unifying framework for the siso circle criterion and other quadratic stabilty criteria. *International Journal of Control*, 77(1):1–9, 2004.

[74] Robert Shorten, Fabian Wirth, Oliver Mason, Kai Wulff, and Christopher King. Stability criteria for switched and hybrid systems. *SIAM Review*, 49(4):545–592, 2007.

[75] A. Stolyar, M.I. Reiman, N. Korolev, V. Mezhibovsky, and H. Ristock. Pacing in knowledge worker engagement, 2010. *United States Patent Application 20100266116-A1.*

[76] Ilya Sutskever, James Martens, George Dahl, and Geoffrey Hinton. On the importance of initialization and momentum in deep learning. In *ICML*, pages 1139–1147, 2013.

[77] Martin Takáč, Avleen Singh Bijral, Peter Richtárik, and Nathan Srebro. Mini-batch primal and dual methods for SVMs. In *ICML*, pages 1022–1030, 2013.

[78] Tijmen Tieleman and Geoffrey Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. Technical report, 2012.

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

[80] Sergey Zeltyn and Avishai Mandelbaum. Call centers with impatient customers: Many-server asymptotics of the m/m/n + g queue. *Queueing Systems*, 51(3):361–402, 2005.

Biography

Lam Nguyen was born in Hanoi, Vietnam in October 1986. He grew up in Moscow, Russia and got his B.S. degree in Applied Mathematics and Computer Science from Faculty (Department) of Computational Mathematics and Cybernetics, Lomonosov Moscow State University in June 2008. After his graduation, he came back to Vietnam and worked as a Software Engineer at FPT Software Company. He came to the United States in January 2011 with his wife and received his M.B.A. degree from McNeese State University, Louisiana in December 2013. He joined as a PhD student in the Department of Industrial and Systems Engineering at Lehigh University in August 2014. During the first two years of his Ph.D., he was working with Dr. Alexander Stolyar in the area of Applied Probability, Stochastic Models and Optimal Control. He have been working with Dr. Katya Scheinberg and Dr. Martin Takáč in the area of Large Scale Optimization Problems for Machine Learning since September 2016. Since June 2017, he has been also working as a Research Intern at the IBM T.J. Watson Research Center, where he is doing research in Machine Learning/Deep Learning problems. Lam will be joining as a Post Doctoral Fellow at IBM T.J. Watson Research Center in September 2018.

Lam met his wife Dung Nguyen in Hanoi, Vietnam in August 2005 and got married with her in January 2015. Their beloved daughter Sarah Nguyen was born in March 2016. Lam has proposed a new algorithm for machine learning problems called SARAH (which is named after his daughter's name Sarah Nguyen) for solving convex and nonconvex large scale optimization problems in 2017 and the work is published at ICML 2017.