Asynchronous Methods in Gradient Descent

DISSERTATION SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

Master of Technology

in

Computer Science

by

Koushik Ghosh

[Roll No: CS-1906]

under the guidance of

Dr. Swagatam Das

Associate Professor

Electronics and Communication Sciences Unit

Indian Statistical Institute

Kolkata-700108, India

July, 2021

To my Parents

CERTIFICATE

This is to certify that the dissertation entitled "**Asychronous Methods in Gradient Descent** " submitted by **Koushik Ghosh** to Indian Statistical Institute, Kolkata, in partial fulfillment for the award of the degree of **Master of Technology in Computer Science** is a bonafide record of work carried out by him under my supervision and guidance. The dissertation has fulfilled all the requirements as per the regulations of the institute and, in my opinion, has reached the standard needed for submission.

07/09/2021

Swagatam Das
Associate Professor,
Electronics and Communication Sciences Unit,
Indian Statistical Institute,
Kolkata-700108, INDIA

Acknowledgements

I would take this oppurtunity to thank my advisor , *Dr. Swagatam Das*, Associate Professor, Electronics and Communication Sciences Unit, Indian Statistical Institute, Kolkata, for guiding me through the project. This work would not have been possible without his help and ideas.

I also thank my friend Arnab Roy, Abhirup, Sohan Ghosh, Agnip Majumadr among others for their encouragement for the work.

Koushik Ghosh
Indian Statistical Institute
Kolkata - 700108 , India

07/09/2021

Contents

Abstract

Su, Boyd and Candes '14 [1] showed that if we make the stepsizes smaller and smaller, Nesterov Accelerated Gradient Descent converges to a 2nd order ODE. On the other hand, arjevani has shown recently some convergence results on delayed vanilla Gradient descent . Our idea is to take a delayed version of Nesterov Accelerated Gradient Descent and derive it's corresponding ODE and prove convergence for the convex case.

Keywords: Nesterov Accelerated Gradient Descent, Asynchronous

Chapter 1

Introduction

Optimization is the workhorse of deep learning with large data. And Optimizers like Stohcastic Gradient Descent and Adam are the most important. Parallelism and acceleration are two important aspects of optimization. Vanilla gradient descent is a Greedy approach that takes step in the direction of steepest descent . However, this approach turns out to be slow and there are ways to boost convergence by having some sort of momentum that pushes the ball in the direction of descent. Nesterov Accelerated Gradient Descent(NAG)[8] is one such method that improve the rate of convergence under same assumptions as vanilla Gradient Descent.One vital point is NAG is not a descent method i.e the function value does not necessarily decrease with each update as is the case in vanilla Gradient Descent. Acceleration is still mysterious and Su Boyd Candes [1] derived an ODE for NAG [8] in the continuous case which sheds some physical intution on it in terms of ball moving with momentum along with friction . These optimizers serve to train deep neural networks.Another way of speeding up the training is parallelism using multiple machines . Mini-batch methods can be easily parallelized.These are synchronous methods. We can also consider asynchronous methods. And these tend to work well in practice. However, theoretically analyzing them is extremely difficult.

Chapter 2

Related Works

Optimization broadly deals with the problem $\min f(x)$ where f is a function that may be convex or non-convex, smooth or non-smooth. Several methods have been proposed since Newton's method to deal with these optimization problems. These days data size is becoming larger and larger and to deal with these acceleration of optimization is being studied. First order methods are more popular due to lesser computational cost and memory efficiency. Nesterov 30 years back in [8] proposed the following algorithm

- $x_k = y_{k-1} - \epsilon \nabla f(y_{k-1})$
- $y_k = x_k + \beta_{k-1}(x_k - x_{k-})$

where ϵ is the stepsize and β_k is the momentum term . Let L be the smoothness constant of the function f i.e $\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$. Let x_* be the minima of the convex function f and $e_0 = \|x_0 - x_*\|$ where x_0 is the starting point of the algorithm .

Theorem 2.0.1. *In [8],Nesterov proved if the* $\epsilon \leq \frac{1}{L}$ *, f is convex with smoothness parameter L , then* $f(x_k) - f(x_*) \leq O(\frac{e_0^2}{\epsilon k^2})$

This rate is considered to be optimal among first order methods. This method has applications in training deep architectures as well as classical machine learning problems like sparse regression, compressed sensing. Connections between Ordinary differential equations have been an object of study for a long time. If the step sizes are taken a limit to 0 , then the path converges to solution of an ODE. This provides

some intuition into the gradient descent methods since analysis of continuous ODE's is a lot easier.

2.1 2nd order ODE for Nesterov Accelerated Gradient Descent

In Su Boyd Candes [1], the authors derive an ODE which is a continuized version of Nesterov Accelerated Gradient Descent. This is in the spirit of the continuous version of gradient descent known as Gradient Flow given by the equation $\dot{Y}(t) = -\nabla f(Y(t))$ where $t \in 0, \infty)$ Unlike the latter which is a first order differential equation, this happens to be a 2nd order ODE This is interesting because deriving the convergences of the continuous version is generally easier than the discrete version under the usual assumptions of convexity and strong convexity the following ODE is derived as the continuous version of discrete NAG $\ddot{X} + \frac{3}{t}\dot{X} + \nabla f(X) = 0$ ....(1) in the case of convexity , i.e f is assumed to be convex, convergence rate is $O(\frac{1}{t^2})$ compared to the case of gradient flow in which case there is a convergence rate of $O(\frac{1}{t})$, analogy of acceleration of convergence extends onto the continuous case.

in the case f is not assumed to be differentiable , using subgradients a similiar rate of convergence $O(\frac{1}{t^2})$ is shown

Theorem 2.1.1. *Consider $f \in \mathcal{F}_\infty := \cup_{L>0}\mathcal{F}_L$ and any x0 R , the ODE (1) with boundary conditions $X(0) = x_0$,$X^{'}(0) = 0$ has a unique global solution $X \in C^2((0,+\infty);\mathbb{R}^n)\bigcup C^1([0,+\infty);\mathbb{R}^n)$).*

Theorem 2.1.2. *For any $f \in \mathcal{F}_\infty$, as the step size $\epsilon \to 0$, Nesterov's algorithm converges to the ODE (1) in the following sense : $\lim_{\epsilon\to 0} \max_{0\le k\le\sqrt{T}} \|x_k - X(k\sqrt{\epsilon})\| = 0$*

Theorem 2.1.3. *Consider the space $\mathcal{F}_\infty = \cup\mathcal{F}_L$ where $\mathcal{F}_L$ denotes the set of convex smooth functions with smoothness parameter L . Let $X(t)$ be global soltion to (1). Then $f(X(t)) - f(x_*) \le 2\frac{e_0^2}{t^2}$*

. Proof Idea: Consider the Lyanopunov function $E(t) = t^2(f(X(t)) - f(x_*)) + 2\|X(t) + \frac{t}{2}\dot{X} - x_*\|^2$. Show the derivative $\dot{E}(t)$ is less than equal to 0 .
With the identification of $t \approx k\sqrt{\epsilon}$, analogy with the Theorem 2.0.1 is clear . Consider $\ddot{X} + \frac{r}{t} + \nabla f(X) = 0$ where the constant 3 in (1) is replaced by a general constant r.

Theorem 2.1.4. *For $r \ge 3$, $f(X(t)) - f(x_*) \le (r-1)^2\frac{e_0^2}{2t^2}$*

Proof Idea: Take the Lyanpunov function $E(t) = 2t^2(f(X(t)) - f(x_*)) + (r-1)^2\|X(t) + \frac{t}{2}\dot{X} - x_*\|^2$ Show the derivative $\dot{E}(t)$ is less than equal to 0 .then $E(t) \leq E(0)$ from which the result follows .

Theorem 2.1.5. *For $r \leq 3$ and if $(f(x)-f(x_*))^{\frac{r-1}{2}}$ is a convex function , $f(X(t))-f(x_*) \leq (r-1)^2\frac{e_0^2}{2t^2}$*

Definition 1. *For any convex function f , we can define directional derivative in the direction p at point $x \in \mathbb{R}^d$ $f^{'}(x;p) = \lim t \to 0+\frac{f(x+tp)-f(x)}{t}$. This limit can be shown to exist for any convex function though it may be unbounded.*

f has a derivative at x $x \Leftrightarrow f^{'}(x;p)$ is linear in p

$f^{'}(x;p) = \sup_{g\in\partial f(x)} g^T p$ where $\partial f(x)$ is set of subgradients of f

Theorem 2.1.6. *If f is a convex function with directional derivated $G_f(x,p)$ and consider the following 2nd order ODE, $\ddot{X} + \frac{r}{t} + G_f(X,\dot{X}) = 0$ has a solution $X(t)$ on $t \in [0,\gamma)$. for $t \in [0,\gamma)$, the following holds, $f(X(t)) - f(x_*) \leq 2\frac{e_0^2}{t^2}$ Here $r \geq 3$*

2.2 Delayed Vanilla Gradient Descent

Yossi Arjevani [2] considers a simple version of Asynchronous Stochastic Gradient Descent and proves the following theorems $x_{k+1} = X_k - \epsilon(\nabla f(x_{k-\tau}) + \xi_k)$ where $x_0 = x_1 = ...x_\tau$ f is assumed to be convex or strongly convex function so $f(x) = \frac{1}{2}x^T Mx + b^T x + c$ where $M \in \mathbb{R}^{d\times d}$.

Theorem 2.2.1. *for the delay $\tau \geq 1$and $k \geq (\tau+1)log(2(\tau+1))$ above version of delayed gradient descent in the deterministic case where $\psi_k = 0$. L is the smoothness parameter and μ is the strong convexity parameter and $\epsilon = \frac{1}{\mu\tau}$, then $f(x_k) - f(x_*) \leq O(Le_0^2 \exp(-\frac{k\mu}{L\tau}))$ and in case of smooth L convex function $f(x_k) - f(x_*) \leq O(\frac{L\tau e_0^2}{k})$*

Theorem 2.2.2. *$\mathcal{E}[f(x_k) - f(x_*)] \leq O(Le_0^2 \exp(-\frac{k\mu}{L\tau}) + \frac{\sigma^2}{\mu k})$ for the strongly convex case under the same assumptions as Thm 2.2.1 and $\mathcal{E}$ is the expectation . and for the convex smooth case, then $\mathcal{E}[f(x_k) - f(x_*)] \leq O(\frac{L\tau e_0^2}{k} + \frac{\sigma e_0}{\sqrt{k}})$ assuming second moment of the noise is bounded by σ^2*

As we can see , in the case $\sigma = 0$, Thm 2.2.2 is equivalent to thm 2.2.1 . In case of deterministic case, we can see the bounds linearly worsen with the delay. however, in the delayed SGD case, dominant terms in both convex and strongly convex case are dominated by terms that are free of delay . so we can overcome the effects of delay in the delayed SGD. Proof technique follows the method of generating functions . , then treating the delayed GD updates as recursion we can solve for the coefficients in terms of generating functions standard for solving things like fibonacci series to get a closed form .Thus we recover $x_k = [z^k]\frac{x_0}{1-z+\epsilon M z^{\tau+1}}$. now using some complex analysis to bound the roots of the polynomial and the fact that operator norm of a polynomial in operator is bounded by values of the polynomial on one of its eigenvalues, we can prove the results mentioned in Thm 2.2.1 and Thm 2.2.2

2.3 Delay Compensated Stochastic Gradient Descent

Stich's paper [3]extends the results of Arjevani's paper [2] to more general case of functions where f is not assumed to be quadratic. This requires more challenges.

And authors deal with these with the theory of Delay Compensation they have developed over several previous papers. Equivalent rates of convergence in Thm 2.2.1 and Thm 2.2.2 in case of general convex + smooth and strongly convex + smooth are proven .

Definition 2. *Assumption 1 (μ quasi convexity wrt x^*) of a function f when the following holds at all points in $x \in \mathbb{R}^d$:*
$f(x) - f^\star + \frac{\mu}{2}\|x - x^\star\|^2 \leq \langle \nabla f(x), x - x^\star \rangle$

Definition 3. *f is μ strongly convex when the following holds for for all $x, y \in \mathbb{R}^d$:*
$f(x) - f(y) + \frac{\mu}{2}\|x - x^\star\|^2 \leq \langle \nabla f(x), x - y \rangle$

Definition 4. *Polyak Lojasiewicz inequality holds when $\|\nabla f(x)\|^2 \geq 2\mu(f(x) - f(x^\star)$ for all points $x \in \mathbb{R}^d$. This is weaker than quasi convexity since*
$f(x) - f^\star + \frac{\mu}{2}\|x - x^\star\|^2 \leq \langle \nabla f(x), x - x^\star \rangle \leq \frac{1}{2\mu}\|\nabla f(x)\|^2 + \frac{\mu}{2}\|x - x^*\|^2$

Definition 5. *Assumption 2 f is L smooth when*
$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$ *for all x,y. this implies Nesterov 2004, lemma 1.2.3*
$f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + \frac{L}{2}\|y - x\|^2$ *for all $x, y \in \mathbb{R}^d$*

Definition 6. *Assumption 3a of (M, σ^2) bounded noise hold when . $g = \nabla f(x) + \xi$ for some differentiable function f and conditionally independent noise ξ , then there exists constants M,$\sigma^2 > 0$ s.t $\mathbb{E}[\xi|x] = 0$ and $\mathbb{E}[\|\xi\|^2|x] \leq M\|\nabla f(x)\|^2 + \sigma^2$*

Definition 7. *Assumption 3b of (M, σ^2) bounded noise hold when . $g = \nabla f(x) + \xi$ for some L smooth quasi convex f and conditionally independent noise ξ , then there exists constants M,$\sigma^2 > 0$ s.t $\mathbb{E}[\xi|x] = 0$ and $\mathbb{E}[\|\xi\|^2|x] \leq 2LM(f(x) - f^\star) + \sigma^2$*

$f(x) = |x|(1 - \exp(-|x|))$ is quasi convex but not convex. following the ideas of perturbed iterate analysis .

Consider delay compensated algorithms of the form :

$x_{t+1} = x_t - v_t$

$e_{t+1} = e_t + \gamma_t g_t - v_t$

After T such updates, we output $x^{out} \in (x_t)_{t=0}^{T-1}$ where x_t is chosen according to probabilities proportional to w_t for a sequence of positive weights w_t

For delayed SGD we can recover it from the above algorithm by putting $v_t = \gamma_{t-\tau} g_{t-\tau}$ for $t \geq$, and $v_t = 0$ o.w; with $e_t := \sum_{i=1}^{i=\tau} \gamma_{t-i} g_{t-i}$

Theorem 2.3.1. *Let x_t for $t \geq 0$ be the iterates of delayed SGD with stepsize $\gamma_t = \gamma$ constant for all $t \geq 0$ on a function f that satisfies Assumptions 2 , 3. Then,*

- *if f also satisfy Assumption 1 for some positive μ , then with an appropriate γ less than $\frac{1}{10L(\tau+M)}$*

  $E[f(x^{out}) - f(x^*)] = \mathcal{O}(L(\tau + M)\|x_0 - x_*\|^2 \exp(\frac{-\mu T}{-10L(\tau+M)}) + \frac{\sigma^2}{\mu T})$ *where w_t is proportional to $(1 - \frac{\mu\gamma}{2})^{-t}$*

- *If f satisfies Assumption 1 for $\mu = 0$, and γ less than $\frac{1}{10L(\tau+M)}$, then $E[f(x^{out})-f(x^*)] = \mathcal{O}(\frac{L(\tau+M)\|x_0-x_*\|^2}{T} + \frac{\sigma\|x_0-x_*\|}{\sqrt{T}})$ where $w_t = 1$ for all t*
- *take an arbitrary non-convex function, then for some stepsize small enough , $E[\|\nabla f(x^{out})\|^2] = \mathcal{O}(\frac{L(\tau+M)(f(x_0)-f(x_*))}{T} + \frac{\sigma}{\sqrt{T}}\sqrt{L(f(x_0)-f(x_*))})$ where $w_t = 1$*

Definition 8. *δ -approximate compressor.. A random operator $C : \mathcal{R}^d \to \mathcal{R}^d$ that satisifes for δ positive, $E_C[\|x - C(x)\|^2] \leq (1-\delta)\|x\|^2$*

To reduce communication costs we consider a compressor C . In this case we only have a single worker. here $v_t = C(e_t + \gamma_t g_t)$ and $e_{t+1} := e_t + \gamma_t g_t - v_t$

Theorem 2.3.2. *Let x_t for $t \geq 0$ be the iterates of the above delay compensated compressed sgd with stepsize $\gamma_t = \gamma$ constant for all $t \geq 0$ on a function f that satisfies Assumptions 2 , 3. Then,*

- *if f also satisfy Assumption 1 for some positive μ , then with an appropriate γ less than $\frac{1}{10L(\frac{2}{\delta}+M)}$*

  $E[f(x^{out}) - f(x^*)] = \mathcal{O}(L(\frac{1}{\delta}+M)\|x_0 - x_*\|^2 \exp(\frac{-\mu T}{-10L(\frac{2}{\delta}+M)}) + \frac{\sigma^2}{\mu T})$ *where w_t is proportional to $(1-\frac{\mu\gamma}{2})^{-t}$*
- *If f satisfies Assumption 1 for $\mu = 0$, and γ less than $\frac{1}{10L(\tau+M)}$, then $E[f(x^{out})-f(x^*)] = \mathcal{O}(\frac{L(\frac{1}{\delta}+M)\|x_0-x_*\|^2}{T} + \frac{\sigma\|x_0-x_*\|}{\sqrt{T}})$ where $w_t = 1$ for all t*
- *take an arbitrary non-convex function, then for some stepsize small enough , $E[\|\nabla f(x^{out})\|^2] = \mathcal{O}(\frac{L(\frac{1}{\delta}+M)(f(x_0)-f(x_*))}{T} + \frac{\sigma}{\sqrt{T}}\sqrt{L(f(x_0)-f(x_*))})$ where $w_t = 1$*

in case of Local SGD , $x_{t+1}^k = \frac{1}{K}\sum_{k=1}^K (x_t^k - \gamma_t g_t^k)$ if $\tau|(t+1)$ else $x_{t+1}^k = x_t^k - \gamma_t g_t^k$ here the sequences evolve parallely and it synchronized in every τ th step

Theorem 2.3.3. *Let x_t^k for $t \geq 0$ be the iterates of Local SGD with stepsize $\gamma_t = \gamma$ constant for all $t \geq 0$ on a function f that satisfies Assumptions 2 , 3. Then,*

- *if f also satisfy Assumption 1 for some positive μ , then with an appropriate γ less than $\frac{1}{10L(\tau K+M)}$*

  $E[f(x^{out}) - f(x^*)] = \mathcal{O}(L(\tau K+M)\|x_0 - x_*\|^2 \exp(\frac{-\mu T}{-10L(\tau K+M)}) + \frac{\sigma^2}{\mu T})$ *where w_t is proportional to $(1-\frac{\mu\gamma}{2})^{-t}$*
- *If f satisfies Assumption 1 for $\mu = 0$, and γ less than $\frac{1}{10L(\tau K+M)}$, then $E[f(x^{out}) - f(x^*)] = \mathcal{O}(\frac{L(\tau K+M)\|x_0-x_*\|^2}{T} + \frac{\sigma\|x_0-x_*\|}{\sqrt{TK}})$ where $w_t = 1$ for all t*
- *take an arbitrary non-convex function, then for some stepsize small enough , $E[\|\nabla f(x^{out})\|^2] = \mathcal{O}(\frac{L(\tau K+M)(f(x_0)-f(x_*))}{T} + \frac{\sigma}{\sqrt{KT}}\sqrt{L(f(x_0)-f(x_*))})$ where $w_t = 1$*

2.4 Nesterov Acceleration in distributed settings

Distributed Nesterov Gradient descent is studied in [8]. This method has a sequence $(x_i(k), y_i(k))$ at each node i of a network. Each node does updates after communicating with its neighboring nodes and computes the gradient step with respect to its own value.. The update equations are

$x_i(k+1) = \sum_{j \in N_i} W_{ij} y_j(k) - \alpha_k \nabla f_i(y_i(k))$

$y_i(k+1) = x_{k+1} + \beta_k (x_i(k+1) - x_i(k))$

where W_{ij} is the weight of the edge connecting nodes i and j and N_i is the neighbourhood of node i . choose $\alpha_k = \frac{c}{k+1}$ and $\beta_k = \frac{k}{k+3}$ At each update , each node sends $y_i(k)$ to its neighbors and each node as a result also receive that value from its neighbours. Then the values are weighed according to the weight of the edge of the network and then substracts gradient with regard to its component function f_i . In this method, Distributed Nesterov Gradient, achieves rates O (log K/K) and O (log k/k), where per-node communications is K and the per-node gradient evaluations k Another method is developed in [8] Distributed Nesterov gradient with Consensus iterations. This requires apriori knownledge of smoothness parameter of f as well as of the largest singular value of the weight matrix W. Here the convergence rates are improved to $O(\frac{1}{K^{2-\epsilon}})$ and $O(\frac{1}{k^2})$. Boundedness of the gradients is assumed here.

Based on these papers, the next target could be to extend the results of Stich and Arjevani to the accelerated case with delay.

2.5 Problem Statement

Asynchronous methods have been extremely useful for training deep neural nets. Arjevani's paper[2] considered a simple version of the delayed gradient descent namely with constant delay . In that case, Arjevani was able to show Delayed SGD is able to overcome the effects of delay . Extending these results to accelerated gradient descent methods is still an open problem . In this thesis, we take a simpler problem, namely we derive a 2nd order ODE associated to delayed version of NAG in the spirit of Su,Boyd,Candes [1] paper. We are able to show that for convex function not neccesarily differentiable a convergence rate of $O(\frac{\tau^2}{t^2})$. We hope in the same spirit $O(\frac{\tau^2}{\epsilon k^2})$ would extend into the discrete delayed NAG. however , that would involve more complex methods and we keep that for the future.

Chapter 3

Mini-Batch vs Asynchronous Gradient Descent

Parallelization of data for the purpose of optimization to reduce computational time is widely used. It is used to reduce training time of deep neural networks. Joeri R. Hermans[7] compares minibatch to asynchronous methods. In mini-batch data parallelism , work is divided among n workers each of which compute gradients for m data points.

. In the beginning , all the components get a copy of the parameter θ_0 of the central machine. After each of the worker completes it's task, central machine updates the parameter θ_0 using average of the received computations and produces a new update of the central parameter. Then the workers can update themselves with the new parameter and begin to work again. This leads to the phenomenon of locking in that all the workers are not always active as some of them finish early and some finish later and have to wait for the others to complete before the central worker can begin to work . The visualization for this is given in Fig 1 In order to stop wasteful waiting and speed up the processes asynchronous methods are used. In asynchronous method of updates, whoever finishes the computation updates the central parameter first. this leads to problem that gradients being sent to the central machine gets delayed. although this generally works well in practice, analyzing them theoretically is hard. The Visualization for this is given in Fig 2 .

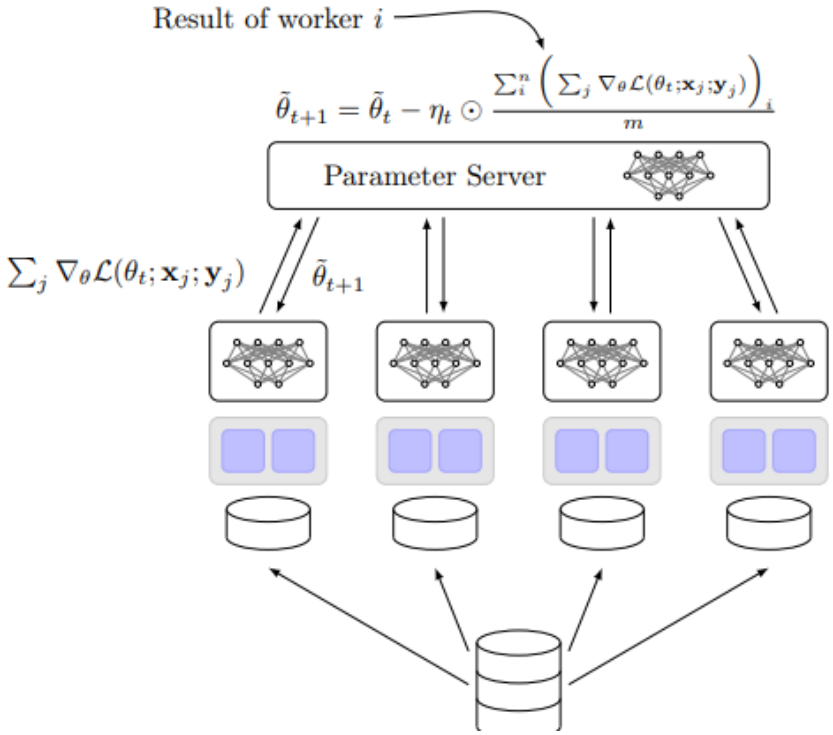

Figure 3.1: Mini-Batch Parallelism

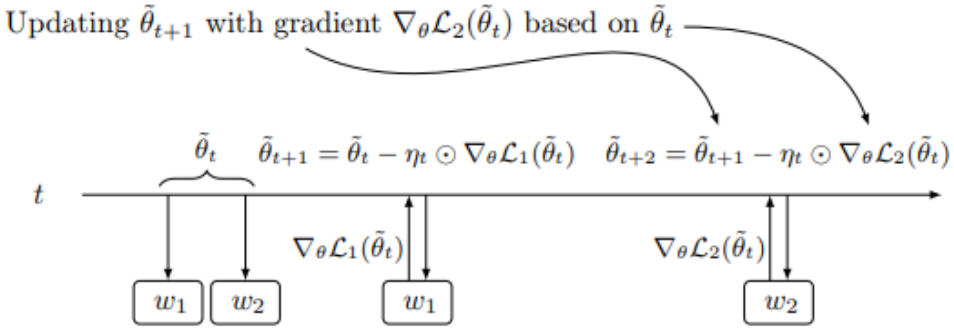

Figure 3.2: Asynchronous Parallelism

3.1 hogwild

Niu [4] came up with a decentralized gradient descent scheme called Hogwild! that does updates in parallel on CPUs. Processors can access a common memory without locking the parameters.This method can only work for sparse data so that in each update , the scheme will only modify a small part of parameter space.The authors then prove prove optimal rates of convergence for this algorithm.

3.2 DownPour SGD

Downpour SGD in Dean [5] is an asynchronous version of Stochastic Gradient Descent by Dean in their DistBelief framework . It runs multiple copies of a algorithm parallely on different parts of the training set. These models send their updates to a the central server containing the parameters, which is divided among different machines. However,this method has the risk of divergence.

3.3 Delay-tolerant Algorithms for SGD

McMahan and Streeter [6] has extended AdaGrad to the distributed setting by developing delay-tolerant algorithms that as in adagrad adapts past gradients and also likewise update delays . This experimentally works well.

Chapter 4

Convergence of Gradient Flow

4.1 Heuristic derivation of the ODE for Gradient Descent

Let's derive a 1st order ODE for gradient descent in continuous time. Throughout we will assume the functions are sufficiently smooth Consider the differential equation.

$\frac{dx}{dt} = -\nabla f(x)$
Consider the flow of $x(t)$ along the ODE . then for small δ,
$x_{t+\delta} = x_t + \delta\dot{x} + O(\delta^2)$ ....(1)

as the velocity $\dot{x} = -\nabla f(x)$, by substituting it in (1) , we get $x_{t+\delta} = x_t - \delta f(x_t)$
Define $X_k := x_{\delta k}$
then $X_{k+1} = X_k - \delta\nabla f(x)$. –(2)
thus we recover the vanilla gradient descent in (2) by discretization of (1). So let's prove some convergences for gradient flow equation which was shown above to be a continous version of the Vanilla Gradient Descent :

$\frac{dx}{dt} = -\nabla f(x(t))$ ...(1) This equation is the gradient flow equation It is easy to demonstrate that if a particle flows along the above ODE, then the value of the function decreases.

4.2 Proof of Convergence of Gradient Flow

Lemma 4.2.1. $\frac{df(x(t))}{dt} \leq 0$

Proof : $\frac{df(x(t))}{dt} = \langle \nabla f(x(t)), \frac{dx(t)}{dt} \rangle = -\|\nabla f(x(t))\|^2 \leq 0$ where the second

equality follows from (1) and first equality follows from application of chain rule if f is assumed to be greater than equal to some bounded constant , then along with monotonicity we get convergence of $f(x(t))$. However, to get convergence of $x(t)$ requires the method of Lynapunov functions that we will discuss next.

Convergence in the case f is a convex function

Definition 9. *Convexity means f is bounded below by its tangent which means* $f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle$ *where the R.H.S is the value of tangent equation at x evaluated at y.*

Theorem 4.2.2. *:Assuming f to be convex, gradient flow has a convergence rate of* $O(\frac{1}{t})$. *Then* $f(x_t) - f(x_*) \leq \frac{\|x - x_0\|^2}{t}$

Proof Technique : Define $E(t) = t(f(x_t) - f(x_*)) + \frac{1}{2}\|x(t) - x_*\|^2$. $E(t)$ can be shown to be a decreasing function of time . Such functions are known as Lynapunov functions.

4.3 Convergence in the case f is a strongly convex function

Definition 10. *If f is* μ *strongly convex, it means that* $f(x) \geq f(y) + < \nabla f(y), x - y > + \mu \frac{\|x-y\|^2}{2}$ *which means that f is lower bounded by a quadratic function.*

Using these assumptions, it can be shown that gradient flow converges linearly which is an improvement over sublinear convergence as in the linear case.

Theorem 4.3.1. *: if f is* μ *strongly convex, gradient flow converges at* $f(x(t)) - f(x_*) \leq O(exp(-2\mu t))$

Lemma 4.3.2. *: if f is* μ *strongly convex then f also follows PL-inequality* $\frac{1}{2}\|\nabla f(x)\|^2 \geq \mu(f(x) - f(x^*))$

Proof : Using the assumptions of strong convexity we get

$$f(y) \geq f(x) + \nabla f(x)^T (y - x) + \frac{\mu}{2}\|y - x\|^2$$

...(3)

Now let's try to minimize both sides the inequality in (3) wrt y. :

$$\min_y \{f(y)\} = f(x^*)$$

Since the R.H.S is a quadratic in y , we do simple differentiation to get the minima

$$\frac{\partial R.H.S}{\partial y} = \nabla f(x) + \mu(y - x) = 0 \iff y = x - \frac{1}{\mu}\nabla f(x)$$

....(4) Putting (4) back into the R.H.S , we get $f(x) - \frac{1}{\mu}\|\nabla f(x)\|^2 + \frac{1}{2\mu}\|\nabla f(x)\|^2$ and hence we get $f(x^*) \geq f(x) - \frac{1}{2\mu}\|\nabla f(x)\|^2$ Proof of the Theorem : $\frac{d(f(x(t))-f^*)}{dt} = \langle\ \nabla f(x(t)), \frac{dx(t)}{dt}\rangle = -\|\nabla f(x(t))\|^2 \leq 2\mu(f^* - f(x(t)))$ thus we can get linear convergence for the strongly convex case

Chapter 5

ODE for Nesterov Accelerated Gradient Descent method

Nesterov Accelerated Gradient Descent aka NAG

$x_{k+1} = y_k - \epsilon \nabla f(y_k)$ ..(1)

$y_{k+1} = x_{k+1} + \beta_k(x_{k+1} - x_k)$...(2) where the ϵ is the size of each step and β_k is the parameter that denotes the momentum

Theorem 5.0.1. *in [8],nesterov proposed the following : Assuming f to be convex and $\frac{1}{\epsilon}$ smooth, and taking the $\beta_k = \frac{k}{k+3}$, we can get a convergence rate of $O(\frac{1}{\epsilon k^2})$*

This is an improvement over $O(\frac{1}{\epsilon k})$ convergence in case of vanilla gradient descent The following results in this chapter are from Su,Boyd,Candes [1]

5.1 ODE associated to NAG

Theorem 5.1.1. *as ϵ the stepsize tends to 0, the above scheme converges to $\ddot{X}(t) + 3/t\dot{X}(t) + \nabla f(X) = 0$*

Main ingredient that we will use here is the Taylor expansion Proof : For a function u, taylor expansion around a point x_0 is given by

$u(x_0 + \delta x) = u(x_0) + \delta x \frac{\partial u}{\partial x}|_{x=x_0} + \frac{1}{2}(\delta x)^2 \frac{\partial^2 u}{\partial x^2}|_{x=x_0} + o(\delta x)$ where $o(\delta x)$ contains higher order difference terms. the above gives a 2nd order approximation which is a quadratic in δx from the eqn (2), we can plug in $k = k - 1$

$y_k = x_k + \beta_{k-1}(x_k - x_{k-1}) - \epsilon \nabla f(y_k)$ ...(3)

Plugging (3) into (1), $x_{k+1} = x_k + \beta_{k-1}(x_k - x_{k-1}) - \epsilon \nabla f(y_k)$...(4)

from (4) we get

$x_{k+1} - x_k = \beta_{k-1}(x_k - x_{k-1}) - \epsilon \nabla f(y_k)$

next we will try to derive an ODE of NAG in the spirit of the derivation of ODE

we carried out for the vanilla gradient descent in the previous chapter. Correspondingly, one would expect $O(\frac{1}{t^2})$ rates of convergence for the continuous case.
with the heuristic of $t = \delta k$, $\delta = \sqrt{\epsilon}$ and $t = \delta k = \sqrt{\epsilon}k$ so with these approximations x_{k+1} is identified to $X(t + \sqrt{\epsilon})$, x_k with $X(t)$ and x_{k-1} with $X(t - \sqrt{\epsilon})$
so $x_{k+1} - x_k = X(t) + \sqrt{\epsilon}\dot{X}(t) + \frac{\epsilon}{2}\ddot{X}(t) - X(t) = \sqrt{\epsilon}\dot{X}(t) + \frac{\epsilon}{2}\ddot{X}(t) + o(\epsilon)$
similarly $x_k - x_{k-1} = -(X(t) - \sqrt{\epsilon}\dot{X}(t) + \frac{\epsilon}{2}\ddot{X}(t)) + X(t) = \sqrt{\epsilon}\dot{X}(t) - \frac{\epsilon}{2}\ddot{X}(t) + o(\epsilon)$
Plugging these into (4) , we get $\sqrt{\epsilon}\dot{X}(t) + \frac{\epsilon}{2}\ddot{X}(t) + o(\epsilon) = \beta_{k-1}(\sqrt{\epsilon}\dot{X(t)} - \frac{\alpha}{2}\ddot{X}(t) + o(\epsilon) - \epsilon\nabla f(y_k)$
so collecting the coefficients of ϵ we get $\frac{\epsilon}{2}(1 + \beta_{k-1})\ddot{X}(t) + \sqrt{\epsilon}(1 - \beta_{k-1})\dot{X}(t) + \alpha\nabla f(y_k) + o(\sqrt{\epsilon}) = 0$ ....(5) using the paul teng parameters $\beta_{k-1} = \frac{k-1}{k+2} = 1 - \frac{3}{k+2}$ which is appoximately $1 - \frac{3}{k} = 1 - 3\frac{\sqrt{\epsilon}}{t}$
we can also approximate y_k with $X(t)$ plugging in these form of β_k and y_k in (5) and comparing the coefficients of ϵ we can derive the following differential equation . $\ddot{X} + \frac{3}{t} + \nabla f(X) = 0$ ...(5) the constant 3 can also be replaced with a general constant r the paper also proves similiar convergence rates for the case the constant 3 is replace with a general r .

Theorem 5.1.2. *if $X(t)$ flows along the the ODE (5), we get a $O(\frac{1}{t^2})$ rate of convergence*

Proof : Take the Lyanpunov function to be $E(t) = 2t^2(f(X(t) - f^*) + \|X(t) + t\dot{X}(t) - X^*\|^2$ and show using convexity that $\dot{E}(t) \leq 0$. Hence $E(t) \leq E(0)$ from where the result follows

Chapter 6

New Contributions

We have already seen the benefits of accelerated gradient descent methods in the synchronous case. However, we want the benefits of acceleration in the synchronous case to carry into asynchronous case. As in the case of Vanilla SGD with delay, analysis of accelerated methods with delay is much harder. We use the setup of Yossi Arjevani's paper [2] which assumes constant delay of the gradients to reach the central machine.

Theorem 6.0.1. *consider the following delayed NAG .*
$x_k = y_{k-\tau-1} - \epsilon\nabla f(y_{t-\tau-1})$ *..(1)*
$y_k = x_k - \frac{k-1}{k+2}(x_k - x_{k-\tau-1}) = 0$ *...(2)*

as ϵ the stepsize tends to 0, the above scheme converges to $\ddot{X}(t) + 3/t\dot{X}(t) + \frac{1}{(\tau+1)^2}\nabla f(X) = 0$

Proof : $t \approx k\sqrt{\epsilon}$ and $\hat{\tau} \approx \tau\sqrt{\epsilon}$ using the same heuristic as the non delayed case. using (1) and (2), first put $k \to k+1$, in both (1) and (2) , then put $k \to -\tau$ in (2) and plug it back to (1) we can get,
$x_{k+1} - x_{k-\tau} - \frac{k-\tau-1}{k-\tau+2}(x_{k-\tau} - x_{k-2\tau-1}) + \epsilon\nabla f(y_{k-\tau}) = 0$ ...(3)
Consider the taylor expansion
$(x_{k+1} - x_{k-\tau}) = \dot{X}(t-\hat{\tau})\sqrt{\epsilon} + \frac{1}{2}\epsilon(1+\tau)^2\ddot{X}(t-\hat{\tau}) + o(\epsilon)$...(4)
similiarly consider the taylor expansion of
$(x_{k-\tau} - x_{k-2\tau-1}) = \dot{X}(t-\hat{\tau})\sqrt{\epsilon}(1+\tau) - \frac{1}{2}\ddot{X}\epsilon(1+\tau)^2 + o(\epsilon)$
also, $\frac{k-1}{k+2} = 1 - \frac{3}{k+2} \approx 1 - \frac{3}{k} \approx 1 - 3\frac{\sqrt{\epsilon}}{t}$ we can plug in the taylor expansions into (3) ,consider only the coefficients of ϵ and derive the following
ODE : $\ddot{X} + \frac{3}{t} + \frac{1}{(1+\tau)^2}\nabla f(X) = 0$

we can also consider the general case where the constant 3 is replaced by r .now $\ddot{X} +$

$\frac{3}{t} + \frac{1}{(1+\tau)^2}\nabla f(X) = 0$ is equivalent to $\ddot{X} + \frac{3}{t} + \nabla f(X) = 0$ as far as proving existence of global solution is concerned under appropriate differentiablity assumptions on f. so Thm 1 of Su,Boyd,Candes [1] paper extends to this new setting and existence of global solution holds. This derivation is inspired by Su Boyd Candes paper [1] Theorem 1 in Su Boyd Candes paper [1] show the existence of global solution under appopriate condition of smoothness and Theorem 2 shows NAG converges to the above ODE.

we prove the following convergence result below analogous to thm 24 of Su,Boyd, Candes '14 [1]

Theorem 6.0.2. . *using the same setup i.e assume a not neccesarily smooth convex function f(x) with directional subgradient $G(x, p; f)$, and consider the boundary conditions to be $X(0) = x_0$ and $\dot{X}(0) = 0$. The corresponding delayed NAG is then $\ddot{X}(t) + \frac{3}{t}\dot{X}(t) + \frac{1}{(\tau+1)^2)}G(X, \dot{X}) = 0$ has a convergence like this*

$f(X(t)) - f^* \leq (\tau + 1)^2 O(\frac{\|x_0 - x_*\|^2}{t^2})$

Proof : Define $E(t) = t^2(f(X(t) - f^*) + 2(\tau + 1)^2\|X(t) + \frac{t}{2}\dot{X}(t) - X^*\|^2$. Since f is not assumed to be differentiable first part of E need not be differentiable but the second part of E is .

Consider $E(t + \delta t) - E(t)$ for small $\delta t > 0$. In $E(t)$ the second term is differentiable and its derivative is

$4(\tau + 1)^2\langle X + \frac{t}{2}\dot{X}(t) - X^*, \dot{X}(t) + \frac{1}{2}\dot{X}(t) + \frac{t}{2}\ddot{X}(t)\rangle$

$= 4(\tau + 1)^2\langle X + \frac{t}{2}\dot{X}(t) - X^*, \frac{3}{2}\dot{X}(t) + \frac{t}{2}\ddot{X}(t)\rangle$..(3)

from the ode $\frac{3}{2}\dot{X}(t) + \frac{t}{2}\ddot{X}(t) = -\frac{t}{2(\tau+1)^2}G(X, \dot{X})$.

$4(\tau+1)^2\langle X+\frac{t}{2}\dot{X}(t)-X^\star, -\frac{t}{2(\tau+1)^2}G(X, \dot{X})\rangle = -2t\langle X-X^*, G(X, \dot{X})\rangle - t^2\langle\dot{X}(t), G(X, \dot{X})\rangle$

hence the 2nd part of $E(t + \delta t) - E(t)$ is equal to

$(-2t\langle X - X^*, G(X, \dot{X})\rangle - t^2\langle\dot{X}(t), G(X, \dot{X})\rangle\delta t + O(\delta t)$

first part of $E(t+\delta t)-E(t)$ is equal to $(t+\delta t)^2(f(X(t+\delta t))-f^*)-t^2(f(X(t)-f^*)$

$= 2t(f(X(t + \delta t)) - f^*)\delta t + t^2(f(X(t + \delta t)) - f(X(t)))$

also $f(X(t + \delta t)) = f(X + \delta t\dot{X}) + O(\delta t)$

$f(X + \delta t\dot{X}) = f(X) + \langle\dot{X}, G(X, \dot{X})\rangle + O(\delta t)$

now combing both parts $E(t + \delta t) - E(t)$

$= 2t(f(X(t+\delta t))-f^*)\delta t+t^2\langle\dot{X}(t), G(X, \dot{X})\rangle+(-2t\langle X-X^*, G(X, \dot{X})\rangle-t^2\langle\dot{X}(t), G(X, \dot{X})\rangle\delta t+$

$O(\delta t) = 2t((f(X) - f^*) - \langle X - X^*, G(X, \dot{X})\rangle) + O(\delta t) \leq O(\delta t)$

since f is convex so $f(X) - f^* \geq \langle X - X^*, G(X, \dot{X})\rangle$and .

Thus $\limsup_{t\to+0} \frac{E(t+\delta t)-E(t)}{\delta t} \leq 0$

which shows $E(t)$ is a non-increasing function .

Then the required result follows from the fact that $E(t) \leq E(0)$ since t is strictly positive

Theorem 6.0.3. . *using the same setup i.e assume a not neccesarily smooth convex function f(x) with directional subgradient $G(x,p;f)$, and consider the boundary conditions to be $X(0) = x_0$ and $\dot{X}(0) = 0$. Consider the more general case of $r > 3$ $\ddot{X}(t) + \frac{r}{t}\dot{X}(t) + \frac{1}{(\tau+1)^2)}G(X,\dot{X}) = 0$ has a convergence like this*

$f(X(t)) - f^* \leq (\tau+1)^2 O(\frac{\|x_0-x_*\|^2}{t^2})$

Proof : Define $E(t) = 2t^2(f(X(t)-f^*) + (r-1)^2(\tau+1)^2\|X(t) + \frac{t}{r-1}\dot{X}(t) - X^*\|^2$. Since f is not assumed to be differentiable first part of E need not be differentiable but the second part of E is .

Consider $E(t+\delta t) - E(t)$ for small $\delta t > 0$. In $E(t)$ the second term is differentiable and its derivative is

$2(r-1)^2(\tau+1)^2\langle X + \frac{t}{r-1}\dot{X}(t) - X^*, \dot{X}(t) + \frac{1}{r-1}\dot{X}(t) + \frac{t}{r-1}\ddot{X}(t)\rangle$

$= 2(r+1)^2(\tau+1)^2\langle X + \frac{t}{2}\dot{X}(t) - X^*, \frac{r}{r-1}\dot{X}(t) + \frac{t}{r-1}\ddot{X}(t)\rangle$..(3)

from the ode $\frac{r}{r-1}\dot{X}(t) + \frac{t}{r-1}\ddot{X}(t) = -\frac{t}{(r-1)(\tau+1)^2}G(X,\dot{X})$.

$2(r-1)^2(\tau+1)^2\langle X+\frac{t}{r-1}\dot{X}(t)-X^\star, -\frac{t}{2(\tau+1)^2}G(X,\dot{X})\rangle = -t(r-1)^2\langle X-X^*, G(X,\dot{X})\rangle -$ $t^2(r-1)\langle\dot{X}(t), G(X,\dot{X})\rangle$ hence the 2nd part of $E(t+\delta t) - E(t)$ is equal to

$(-t(r-1)^2\langle X - X^*, G(X,\dot{X})\rangle - t^2(r-1)\langle\dot{X}(t), G(X,\dot{X})\rangle\delta t + O(\delta t)$

first part of $E(t+\delta t)-E(t)$ is equal to $(t+\delta t)^2(f(X(t+\delta t))-f^*) - t^2(f(X(t)-f^*)$

$= 2t(f(X(t+\delta t)) - f^*)\delta t + t^2(f(X(t+\delta t)) - f(X(t)))$

also $f(X(t+\delta t)) = f(X + \delta t\dot{X}) + O(\delta t)$

$f(X + \delta t\dot{X}) = f(X) + \langle\dot{X}, G(X,\dot{X})\rangle + O(\delta t)$

now combing both parts $E(t+\delta t) - E(t)$

$= 4t(f(X(t+\delta t)) - f^*)\delta t + 2t^2\langle\dot{X}(t), G(X,\dot{X})\rangle\delta t - t(r-1)^2\langle X - X^*, G(X,\dot{X})\rangle\delta t -$ $t^2(r-1)\langle\dot{X}(t), G(X,\dot{X})\rangle\delta t + O(\delta t) = 4t((f(X)-f^*) - \frac{(r-1)^2}{4}\langle X - X^*, G(X,\dot{X})\rangle)\delta t +$ $t^2(3-r)\langle\dot{X}(t), G(X,\dot{X})\rangle\delta t + O(\delta t) \leq 4t((f(X)-f^*) - \frac{(r-1)^2}{4}(f(X)-f^*))\delta t + t^2(3-$ $r)\langle\dot{X}(t), G(X,\dot{X})\rangle\delta t + O(\delta t) \leq O(\delta t)$

since f is convex so $f(X) - f^* \geq \langle X - X^*, G(X,\dot{X})\rangle$and $r > 3$.

Thus $\limsup_{t\to+0} \frac{E(t+\delta t)-E(t)}{\delta t} \leq O(\delta t)$

which shows $E(t)$ is a non-increasing function .

Then the required result follows from the fact that $E(t) \leq E(0)$ since t is strictly

positive

These results are my main contribution .

Chapter 7

Future Work and Conclusion

7.1 Open Problems

One direction is to extend the convergence result proved for continuous version of delayed NAG in the convex case to be extended to the discrete NAG . Nesterov Accelerated Gradient Descent unfortunately does not accelerated in the stochastic case.Different Accelerated Stochastic Gradient methods have been proposed like Natasha 2 in [6] in the non-convex setting , Katusha X in [7] in the convex setting recently. One idea could be to try to find continuous ODE of these methods and derive convergence results in a similiar fashion. We haven't derived the stochastic differential equation for delayed vanilla SGD . This could also be a future direction .

7.2 conclusion

We first take limit of step sizes of the delayed version of NAG to find a 2nd order ODE . By using previous results in [1] we are able to establish global solution for the derived differential equation. Next , we find that if a solution path along this ODE converges to the minimum of the function at a rate $(\frac{\tau^2}{t^2})$. This is the same rate of $O(1/t^2)$ as in case of ode for the NAG. except it worsens with the delay quadratically. This is bad. To solve this problem , a future direction could be to consider gradient descent/flow methods that are stochastic and we expect as in the case of [2],[3] for noise to solve this problem of converges rates being affected by the delay.

Bibliography

[1] Weijie Su,Stephen Boyd,Emmanuel J.Candes A Differential Equation for Modeling Nesterov's Accelerated Gradient Method: Theory and Insights , Journal of Machine Learning Research 17 (2016)

[2] Yossi Arjevani, Ohad Shamir, Nathan Srebro A tight convergence analysis for stochastic gradient descent with delayed updates, Algorithmic Learning Theory

[3] Sebastian U. Stich,Sai Praneeth Karimireddy The Error-Feedback Framework: Better Rates for SGD with Delayed Gradients and Compressed Updates Journal of Machine Learning Research 21 (2020)

[4] 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.

[5] Jeffrey Dean et al. "Large scale distributed deep networks". In:Advances in neural informationprocessing systems. 2012, pp. 1223–1231

[6] Adaptive bound optimization for online convex optimization. HB McMahan, M Streeter. Proceedings of the 23rd Annual Conference on Learning Theory (COLT),

[7] On Scalable Deep Learning and Parallelizing Gradient Descent Joeri R. Hermans

[8] Y. Nesterov. A method of solving a convex programming problem with convergence rate O(1/k2). In Soviet Mathematics Doklady, volume 27, pages 372–376, 1983

[9] Alekh Agarwal and John C Duchi. Distributed delayed stochastic optimization. In Advances in Neural Information Processing Systems, pages 873–881, 2011.

[10] Sorathan Chaturapruek, John C Duchi, and Christopher Ré. Asynchronous stochastic convex optimization: the noise is in the noise and sgd don't care. In Advances in Neural Information Processing Systems

[11] Andrew Cotter, Ohad Shamir, Nati Srebro, and Karthik Sridharan. Better mini-batch algorithms via accelerated gradient methods.

[12] Hamid Reza Feyzmahdavian, Arda Aytekin, and Mikael Johansson. A delayed proximal gradient method with linear convergence rate. In Machine Learning for Signal Processing (MLSP), 2014 IEEE International Workshop

[13] Rémi Leblond, Fabian Pederegosa, and Simon Lacoste-Julien. Improved asynchronous parallel optimization analysis for stochastic incremental methods. arXiv preprint

[14] 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.

[15] A Nedic, Dimitri P Bertsekas, and Vivek S Borkar. Distributed asyn ´ chronous incremental subgradient methods. Studies in Computational Mathematics, 8(C):381–407, 2001.

[16] AS Nemirovsky and DB Yudin. Problem complexity and method efficiency in optimization. 1983. Willey-Interscience, New York, 1983.

[17] Yurii Nesterov. Introductory lectures on convex optimization, volume 87. Springer Science Business Media, 2004.

[18] Benjamin Recht, Christopher Re, Stephen Wright, and Feng Niu. Hogwild: A lock-free approach to parallelizing stochastic gradient descent. In Advances in neural information processing systems, pages 693–701, 2011.

[19] Ohad Shamir and Nathan Srebro. Distributed stochastic optimization and learning. In Communication, Control, and Computing (Allerton), 2014 52nd Annual Allerton Conference on, pages 850–857. IEEE, 2014.

[20] Adaptive bound optimization for online convex optimization. HB McMahan, M Streeter. Proceedings of the 23rd Annual Conference on Learning Theory (COLT),

[21] Ohad Shamir and Tong Zhang. Stochastic gradient descent for non-smooth optimization: Convergence results and optimal averaging schemes. In International Conference on Machine Learning, pages 71– 79, 2013.

[22] Benjamin Sirb and Xiaojing Ye. Decentralized consensus algorithm with delayed and stochastic gradients. arXiv preprint arXiv:1604.05649, 20

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

[24] H.-B. D¨urr and C. Ebenbauer. On a class of smooth optimization algorithms with applications in control. Nonlinear Model Predictive Control, 4(1):291–298, 2012.

[25] H.-B. D¨urr, E. Saka, and C. Ebenbauer. A smooth vector field for quadratic programming. In 51st IEEE Conference on Decision and Control, pages 2515–2520, 2012.

[26] S. Fiori. Quasi-geodesic neural learning algorithms over the orthogonal group: A tutorial. Journal of Machine Learning Research, 6:743–781, 2005.

[27] U. Helmke and J. Moore. Optimization and dynamical systems. Proceedings of the IEEE, 84(6):907, 1996.

[28] D. Hinton. Sturm's 1836 oscillation results evolution of the theory. In Sturm-Liouville theory, pages 1–27. Birkh¨auser, Basel, 2005. J. J. Leader. Numerical Analysis and Scientific Computation. Pearson Addison Wesley, 2004.

[29] L. Lessard, B. Recht, and A. Packard. Analysis and design of optimization algorithms via integral quadratic constraints. arXiv preprint arXiv:1408.3595, 2014.

[30] R. Monteiro, C. Ortiz, and B. Svaiter. An adaptive accelerated first-order method for convex optimization. Technical report, ISyE, Gatech, 2012.

[31] Y. Nesterov. A method of solving a convex programming problem with convergence rate O(1/k2). Soviet Mathematics Doklady, 27(2):372–376, 1983.

[32] Y. Nesterov. Introductory Lectures on Convex Pptimization: A Basic Course, volume 87 of Applied Optimization. Kluwer Academic Publishers, Boston, MA, 2004.

[33] Y. Nesterov. Smooth minimization of non-smooth functions. Mathematical Programming, 103(1):127–152, 2005.

[34] Y. Nesterov. Gradient methods for minimizing composite functions. Mathematical Programming, 140(1):125–161, 2013.

[35] J. Schropp and I. Singer. A dynamical systems approach to constrained minimization. Numerical functional analysis and optimization, 21(3-4):537–551, 2000.

[36] N. Z. Shor. Minimization Methods for Non-Differentiable Functions. Springer Science and Business Media, 2012.

[37] I. Sutskever, J. Martens, G. Dahl, and G. Hinton. On the importance of initialization and momentum in deep learning. In Proceedings of the 30th International Conference on Machine Learning, pages 1139–1147, 2013.

[38] P. Tseng. On accelerated proximal gradient methods for convex-concave optimization.

http://pages.cs.wisc.edu/ brecht/cs726docs/Tseng.APG.pdf, 2008.

[39] P. Tseng. Approximation accuracy, gradient methods, and error bound for structured convex optimization. Mathematical Programming, 125(2):263–295, 2010.