arXiv:1805.08244v1 [stat.ML] 21 May 2018

Stochastic Modified Equations for the Asynchronous Stochastic Gradient Descent

Jing An
Institute for Computational and Mathematical Engineering
Stanford University
Stanford, CA 94305
jingan@stanford.edu

Jianfeng Lu
Department of Mathematics
Department of Chemistry and Department of Physics
Duke University, Box 90320
Durham, NC 27708
jianfeng@math.duke.edu

Lexing Ying
Department of Mathematics and ICME
Stanford University
Stanford, CA 94305
lexing@stanford.edu

Abstract

We propose a stochastic modified equations (SME) for modeling the asynchronous stochastic gradient descent (ASGD) algorithms. The resulting SME of Langevin type extracts more information about the ASGD dynamics and elucidates the relationship between different types of stochastic gradient algorithms. We show the convergence of ASGD to the SME in the continuous time limit, as well as the SME's precise prediction to the trajectories of ASGD with various forcing terms. As an application of the SME, we propose an optimal mini-batching strategy for ASGD via solving the optimal control problem of the associated SME.

1 Introduction

In this paper, we consider the following empirical risk minimization problem commonly encountered in machine learning:

$$\min_{x\in\mathbb{R}^d} f(x) := \frac{1}{n}\sum_{i=1}^{n} f_i(x) \tag{1}$$

where x represents the model parameters, $f_i(x) \equiv f(x; z_i)$ denotes the loss function of the training sample z_i, and n is the size of the training sample set. Since the training set for most applications is of large size, stochastic gradient descent (SGD) is the most popular algorithm used in practice. In the simplest scenario, SGD randomly samples one instance $f_i(\cdot)$ at each iteration and updates the parameter by evaluating only the gradient of the selected $f_i(\cdot)$. The stability and convergence rate of SGD have been studied in depth, for example, see [8, 16]. However, the scalability of SGD is unfortunately restricted by its inherent sequential nature. To overcome this issue and hence accelerate the convergence, there has been a line of research devoted to asynchronous parallel SGDs. In the distributed computation scenario, an asynchronous stochastic gradient descent (ASGD) method parallelizes the computation on multiple processing units by (1) calculating multiple gradients simultaneously at different processors and (2) sending the results asynchronously back to the master for updating the model parameters [1, 20].

Preprint. Work in progress.

1.1 Related Work

There has been a vast literature on the analysis of SGD, see for example Bottou et al. [2] for a comprehensive review of this subject. Some widely-used methods include AdaGrad [4], which extends SGD by adapting step sizes for different features, RMSProp [23], which resolves AdaGrad's rapidly diminishing learning rates issue, and Adam [10], which combines the advantages of both AdaGrad and RMSProp with a parameter learning rates adaption based on the average of the second moments of the gradients. On the other hand, relatively few studies are devoted to ASGDs. Most of these studies for ASGD take an optimization perspective. Hogwild! [20] assumed data sparsity in order to run parallel SGD without locking successfully. Under various smoothness conditions on f such as f being strongly convex and f_i's all Lipschitz, it showed that the convergence rate can be similar to the synchronous case. Duchi et al. [5] extended this result by developing an asynchronous dual averaging algorithm that allows problems to be non-smooth and non strongly-convex as well. Mitliagkas et al. [15] observed that a standard queuing model of asynchrony correlates to the momentum, that is, the asynchrony produces momentum in SGD updates. There are also several methods using asynchrony either in parallel or distributedly, such as asynchronous stochastic coordinate descent algorithms [13, 14, 17, 21].

Recently, Li et al. [12] introduced the concept of the stochastic modified equation for SGDs (referred as **SME-SGD** in this report), where in the continuous-time limit an SGD is approximated by an appropriate (overdamped) Langevin equation. Compared to most convergence analysis that give upper bounds for (strongly) convex objects, this new framework not only provides more precise analysis for the leading order dynamics of SGD but also suggests adaptive hyper-parameter strategies using optimal control theory.

1.2 Our Contributions

Inspired by Li et al. [12] mentioned above, we extend the application of SME here to characterize the dynamics of ASGD algorithms.

In Section 2, we first derive a stochastic modified equation for the asynchronous stochastic gradient descent, denoted shortly as **SME-ASGD**, for the case where the gradient of each loss function f_i is linear. The derivation results in a Langevin equation, which has a unique invariant distribution solution with a convergence rate dominated by the temperature factor. Meanwhile, for the momentum SGD (MSGD), a similar Langevin equation denoted as **SME-MSGD** is derived and we show that the temperature factors for both derived SME agree. This comparison gives a Langevin dynamics explanation of why an asynchronous method gives rise to similar behavior as compared to the momentum-based methods [15]. Then by introducing a new accumulative quantity, we derive a more general SME-ASGD for the general case in which the gradient of the loss function can be nonlinear. We show that the two SME-ASGDs are equivalent when the gradients are linear.

Section 3 provides some numerical analysis for SME-ASGD by providing a strong approximation estimation to the ASGD algorithm. Different from the usual convergence studies, we do not assume convexity on f and f_i but only require their gradients to be (uniformly) Lipschitz. Numerical results including non-linear forcing terms and non-convex objectives demonstrate that SME-ASGD provides much more accurate predications for the behavior of ASGD compared to SME-SGD derived in [12]. In Section 4, we apply the optimal control theory to identify the optimal mini-batch for ASGD and the numerical simulations there verify that the suggested strategy gives a significantly better performance.

2 Stochastic Modified Equations

The asynchronous stochastic gradient descent (ASGD) carries out the following update at each step:

$$x_{k+1} = x_k - \eta \nabla f_{\gamma_k}(x_{k-\tau_k}), \tag{2}$$

where η is the step size, $\{\gamma_k\}$ are i.i.d. uniform random variables taking values in $\{1, 2, \cdots, n\}$, and $x_{k-\tau_k}$ is delayed read of the parameter x used to update x_{k+1} with random staleness τ_k.

Assumption 1. *We assume that the staleness τ_k are independent and that the sample selection process γ_k is mutually independent from the staleness process τ_k. ∇f_i's are all (uniformly) Lipschitz, that is, for each $1 \leq i \leq n$, there exists $L_i > 0$ such that for any $x, y \in \mathbb{R}^d$, we have $|\nabla f_i(x) -$*

$\nabla f_i(y)| \le L_i|x-y|$. *As a consequence, by taking* $L = \frac{1}{n}\sum_{i=1}^n L_i$, f *is also (uniformly) Lipschitz:* $|\nabla f(x) - \nabla f(y)| \le L|x-y|$. *In addition, the staleness process* τ_k *follows the geometric distribution:* $\tau_k = l$ *(i.e.,* $x_{k-\tau_k} = x_{k-l}$*) with probability* $(1-\mu)\mu^l$ *for* $\mu \in (0,1)$.

Making the geometric distribution assumption here is not only to simplify the computation, but also can be justified by considering the canonical queuing model [24]. For example, the computation at each processor may involve some randomized algorithm that requires each processor to do multiple independent trials until the result is accepted, thus resulting in a geometrically distributed computation time. Our derivation of SME models can be also easily generalized to other random staleness models as long as the expectation of read delays is finite.

2.1 Linear gradients

We first show the derivation of Langevin dynamics with the linear forcing term. Suppose that, for each $1 \le i \le n$, ∇f_i is linear, or equivalently each f_i is quadratic. While this is a fairly restrictive assumption, the derivation in this simplified scenario offers a more transparent view towards the stochastic modified equation for the asynchronous algorithm.

A key quantity for our derivation is the expected read m_k given as follows as a weighted sum of x_k:

$$m_k = \mathbb{E}(x_{k-\tau_k}) = \sum\nolimits_{l=0}^{\infty} x_{k-l}(1-\mu)\mu^l.$$

Note that $m_{k+1} = \sum_{l=0}^{\infty} x_{k+1-l}(1-\mu)\mu^l = x_{k+1}(1-\mu) + \mu m_k$ and $x_{k+1} = (m_{k+1} - \mu m_k)/(1-\mu)$. Plugging this into (2), we can rewrite ASGD as

$$\frac{m_{k+1} - 2m_k + m_{k-1}}{\eta(1-\mu)} = -\frac{m_k - m_{k-1}}{\eta} - \nabla f(m_k) + (\nabla f(m_k) - \nabla f_{\gamma_k}(x_{k-\tau_k})) \tag{3}$$

by using the linearity of ∇f_i. Observe that the left hand side and the first term on the right hand side of (3) can be viewed as divided difference approximations to various time derivatives of m. The second term on the right hand side is the usual gradient. The last term $\nabla f(m_k) - \nabla f_{\gamma_k}(x_{k-\tau_k})$ can be understood as the noise due to stochastic gradient and the read delays; it has mean 0, since the expectation can be decomposed as

$$\begin{aligned}\mathbb{E}\big(\nabla f(m_k) - \nabla f_{\gamma_k}(m_k) + \nabla f_{\gamma_k}(m_k) - \nabla f_{\gamma_k}(x_{k-\tau_k})\big) &= \frac{1}{n}\sum_{i=1}^n (\nabla f(m_k) - \nabla f_i(m_k)) \\ &+ \frac{1}{n}\sum_{i=1}^n \big(\nabla f_i(\sum_{l=0}^{\infty} x_{k-l}(1-\mu)\mu^l) - \sum_{m=0}^{\infty}(1-\mu)\mu^m \nabla f_i(x_{k-m})\big) = 0.\end{aligned}$$

The covariance matrix of the noise will be denoted as

$$\Sigma_k = \mathbb{E}\big((\nabla f(m_k) - \nabla f_{\gamma_k}(x_{k-\tau_k}))(\nabla f(m_k) - \nabla f_{\gamma_k}(x_{k-\tau_k}))^T\big),$$

conditioned on $\{x_{k-l}\}_{l\ge 0}$ and we also denote the square root of Σ_k by σ_k, i.e., $\Sigma_k = \sigma_k \sigma_k^T$. Σ_k (and thus σ_k) in general depends on the previous history of the trajectory, although such dependence is omitted in our notation.

In order to arrive at a continuous time stochastic modified equation from (3), we view m_k as the evaluation of a function m at time points $t_k = k\Delta t$ where Δt is the effective time step size for the corresponding stochastic modified equation, and it is chosen as $\Delta t = \sqrt{\eta(1-\mu)}$. Let us introduce the auxiliary variables $p_k = \frac{1}{\Delta t}(m_k - m_{k-1})$ and reformulate (3) as a system of (m_k, p_k):

$$p_{k+1} = p_k - \Delta t\sqrt{(1-\mu)/\eta}\, p_k - \Delta t \nabla f(m_k) + \Delta t\left(\nabla f(m_k) - \nabla f \gamma_k(x_{k-\tau_k})\right) \tag{4}$$

$$m_{k+1} = m_k + \Delta t\, p_{k+1}. \tag{5}$$

To obtain a SME, we first model the random term by a Gaussian random noise, that is, $\Delta t\big(\nabla f(m_k) - \nabla f(x_{k-\tau_k})\big) \sim \sigma_k(\eta(1-\mu))^{1/4}\Delta B_t$, where $\Delta B_t = B_{t+\Delta t} - B_t$ is the increment of a Brownian motion (thus $\mathbb{E}(\Delta B_t) = 0$ and $\mathbb{E}(\Delta B_t)^2 = \Delta t$) and the coefficient is chosen to match the variance. Assuming that Δt is small, we arrive at a Langevin type equation:

$$\begin{aligned} dP_t &= -\nabla f(M_t)dt - \sqrt{(1-\mu)/\eta}P_t dt + \sigma(t)(\eta(1-\mu))^{1/4} dB_t \\ dM_t &= P_t dt. \end{aligned} \tag{6}$$

When f is a smooth confining potential (for example, f is a quadratic potential), the process approaches to the minimum and $\sigma(t)$ can be approximated by a constant matrix up to a first order approximation for large time t. When this constant matrix is a multiple of the identity matrix, (M_t, P_t) in the standardized model is an ergodic Markov process with stationary distribution [18]:

$$\rho_\infty(p, m) = Z^{-1} e^{-\beta\left(\frac{1}{2}|p|^2 + f(m)\right)}, \tag{7}$$

where Z is a normalization constant. In this case, the resulting friction γ is $\sqrt{(1-\mu)/\eta}$ and the temperature β^{-1} is $\frac{1}{2}\Sigma\eta$. When the constant matrix is not a multiple of identity, the stationary distribution takes such a form in a transformed coordinate system.

The reason why we care about the temperature parameter here is that it quantifies the variance of the noise, and therefore gives us more information about the asymptotic behavior of the optimization process. With such a tool, we can better analyze the connection between different stochastic gradient algorithms. Let us illustrate it by showing one example here: Mitliagkas et al. [15] argues that there is some equivalence between adding asynchrony or momentum to the SGD algorithms, and they showed it by taking expectation to a simple queuing model and finding matched coefficients. Here, we investigate such relation by looking at the corresponding Langevin dynamics, specifically the temperature for both SMEs, which offers a more detailed dynamical comparison.

Stochastic gradient descent with momentum (MSGD) introduced by [19] utilizes the velocity vector from the past updates to accelerate the gradient descent [22]:

$$\begin{aligned} v_{k+1} &= \mu' v_k - \eta' \nabla f_{\gamma_k}(x_k); \\ x_{k+1} &= x_k + v_{k+1}, \end{aligned} \tag{8}$$

with a momentum parameter $\mu' \in (0, 1)$. (8) can be also viewed as a discretization of a second-order differential equation. By following a similar derivation with effective time step $\Delta t = \sqrt{\eta'}$, and taking $p = v/\sqrt{\eta'}$ (details deferred to Appendix B), we end up with the following stochastic modified equation for MSGD (SME-MSGD)

$$\begin{aligned} dP_t &= -\nabla f(X_t)dt - \frac{1-\mu'}{\sqrt{\eta'}} P_t dt + \sigma(X_t)(\eta')^{\frac{1}{4}} dB_t \\ dX_t &= P_t dt, \end{aligned} \tag{9}$$

where friction $\gamma' = \frac{1-\mu'}{\sqrt{\eta'}}$ and temperature $\beta'^{-1} = \frac{\Sigma\eta'}{2(1-\mu')}$ dominates the convergence rate to the stationary solution. Comparing SME-ASGD with SME-MSGD results in the following interesting observation.

Proposition 2. *With $\mu' = \mu$ and $\eta' = \eta(1-\mu)$, SME-ASGD* (6) *and SME-MSGD* (9) *have the same stationary distribution.*

In Theorems 3 and 5 in Mitliagkas et al.'s paper [15], the staleness' geometric distribution parameter μ is taken to be $\mu' = 1 - \frac{1}{M}$, where M is the number of mutually independent workers and μ' is the momentum parameter. With their assumptions, when looking at (6) and (9) under the same time scale, which requires $\eta' = \eta(1-\mu)$, we can see that $\beta'^{-1} = \frac{\Sigma\eta'}{2(1-\mu')} = \frac{\Sigma\eta}{2} = \beta^{-1}$. Since the corresponding temperature for the asynchronous method and momentum method are equal, we conclude that the perspective of stochastic modified equation given above explains the observation in [15] that momentum method has certain equivalent performance as the asynchronous method.

2.2 Nonlinear gradients

We now consider the general case in which the gradient ∇f_i can be non-linear. One can still write the ASGD into a stochastic modified equation by viewing an averaged term as the position and viewing x as the momentum in the Langevin dynamics. For this, let us define a new auxiliary variable y_k which will be viewed as the position in SME as

$$y_k = -\alpha \mathbb{E}(\nabla f(x_{k-\tau_k})) = -\alpha \sum\nolimits_{l=0}^{\infty} \nabla f(x_{k-l})(1-\mu)\mu^l \tag{10}$$

where $\alpha > 0$ is to be determined. Directly following the definition, y_k satisfies the difference equation

$$\frac{y_{k+1} - y_k}{\alpha(1-\mu)} = -\frac{y_k}{\alpha} - \nabla f(x_{k+1}). \tag{11}$$

Moreover, we can rewrite the ASGD (2) into the form

$$\frac{x_{k+1}-x_k}{\eta/\alpha} = y_k + \alpha\Big(-\frac{y_k}{\alpha} - \nabla f_{\gamma_k}(x_{k-\tau_k})\Big) \tag{12}$$

The reason for us arranging terms in this way is to formulate a Langevin-type equation but moving the noise term from the momentum side (X) to the position side (Y). Notice that on the right hand side of (12), $-\frac{y_k}{\alpha} - \nabla f_{\gamma_k}(x_{k-\tau_k})$ can be viewed as a noise with mean 0, and its conditional covariance matrix depends on x_k, y_k (details deferred to Appendix B).

In order to view (11) and (12) as a time-discretization of a coupled system with the same time step size, we match $\alpha(1-\mu) = \eta/\alpha$, which requires the choice of $\alpha = \sqrt{\eta/(1-\mu)}$. Setting the step size $\Delta t = \sqrt{\eta(1-\mu)}$, same as in the linear case, and taking a Gaussian approximation to the noise $\eta\big(-\frac{y_k}{\alpha} - \nabla f_{\gamma_k}(x_{k-\tau_k})\big) \sim \sqrt{\Sigma_k}\frac{\eta^{3/4}}{(1-\mu)^{1/4}}\Delta B_t$, we arrive at the stochastic modified equation for the nonlinear case

$$\begin{aligned} dY_t &= -\nabla f(X_t)dt - \sqrt{\frac{1-\mu}{\eta}}Y_t dt \\ dX_t &= Y_t dt + \sqrt{\Sigma(t)}\frac{\eta^{3/4}}{(1-\mu)^{1/4}}dB_t \end{aligned} \tag{13}$$

Here $\Sigma(t) = \Sigma(\{X_s\}_{0\le s<t}, \{Y_s\}_{0\le s<t})$. In order to close the system of equations, we derive an explicit evolution equation for Σ

$$\begin{aligned} d\Sigma_t = &-\sqrt{\frac{1-\mu}{\eta}}\Sigma_t dt + \sqrt{\frac{1-\mu}{\eta}}\Big(\frac{1}{n}\sum_{i=1}^{n}\nabla f_i(X_t)\nabla f_i(X_t)^T + \frac{1-\mu}{\mu}\nabla f(X_t)\nabla f(X_t)^T\Big)dt \\ &+ \frac{1-\mu}{\eta\mu}\Big(\sqrt{\frac{1-\mu}{\eta}}Y_tY_t^T + \nabla f(X_t)Y_t^T + Y_t\nabla f(X_t)^T\Big)dt. \end{aligned} \tag{14}$$

The derivation of (14) is shown in Appendix B. The combined system (13)–(14) will be referred as SME-ASGD, the stochastic modified equations for asynchronous SGD, for the general nonlinear-gradient case. We should point it out that unlike the linear-gradient case (6) , (13) has no known explicit formula for invariant measure even when $\Sigma(t)$ converging to a constant matrix. Nevertheless, the ergodicity of (13) and (14) will be an interesting future direction to explore.

Remark. When the gradient ∇f is linear, (11) and (12) can be easily transformed back to (4) and (5). As a consequence, (6) and (13) are equivalent. The details of this transformation can be found in Appendix B.

3 Approximation error of the stochastic modified equation

The difference between the time-discrete ASGD and the time-continuous SME-ASGD can be rigorously quantified as follows.

Theorem 3. *Under the Assumption 1 and assume further that the variance from the asynchronous gradients is uniformly bounded, there exists $c > 0$ such that $|\sigma(t)| \le c$. Suppose all the iterates updated from the ASGD stay bounded, and the solutions for SME-ASGD and ASGD respectively up to time 0 agree, i.e., $X_{l\Delta t} = x_l, l \le 0$, with $\Delta t = \sqrt{\eta(1-\mu)}$ as given previously, then the SME-ASGD approximates the ASGD, i.e., there exists constant $K_T > 0$ depending only on T such that*

$$\sup_{n\Delta t\le T}\mathbb{E}\big\{|X_{n\Delta t} - x_n|\big\} \le K_T\frac{\Delta t}{1-\mu} \tag{15}$$

for Δt sufficiently small. Here $X_{n\Delta t}$ is the solution of (13) *at time $n\Delta t$ and x_n is from ASGD* (2)*.*

The assumption $\sigma = \sqrt{\Sigma} = O(1)$ can be justified from (14) as Σ is approximated by a constant matrix for t large. This is because when the iterate approaches to the minimizer, the gradients are close to 0, and Y_t converges to be a constant vector.

The proof of the Theorem (3) follows from viewing the AGSD as a discretization of SME-ASGD and using the analysis of strong convergence for numerical schemes for stochastic differential equations

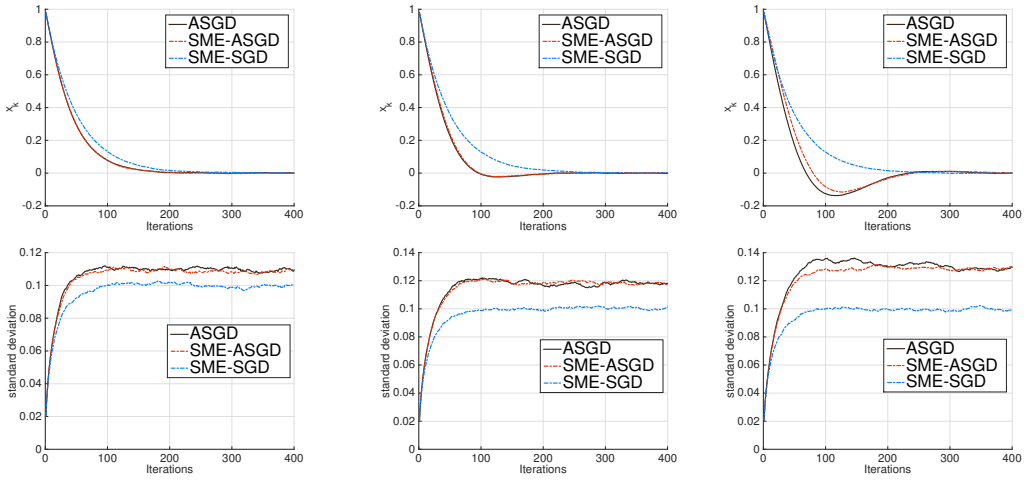
(SDEs). One interesting observation is that, contrary to the standard Euler-Maruyama method for SDEs having strong order of convergence $1/2$ [11], the above result indicates that ASGD, viewed as a discretization of SME-ASGD, has strong order 1. This is because the coefficient of the noise term in the SME-ASGD has ${\eta^{3/4}}/{(1-\mu)^{1/4}}$, which is of order $o(1)$. The SME model proposed in [12] has the same feature: the coefficient of the noise term there is of order $\sqrt{\eta}$. When $\eta \approx 1-\mu$, we can see the order equivalence between the two.

In the following, we provide numerical evidences for Theorem 3 with various loss functions f. The results are shown in Figures 1 (for linear forcing) and 2 (for general forcing). For each example, through averaging over 5000 samples, we compare the results of ASGD with the predictions from both SME-ASGD (13) and the 2nd-order weak convergent SME-SGD proposed in Li et al.'s paper [12]

$$dX_t = -\nabla(f(X_t) + \frac{\eta}{4}|\nabla f(X_t)|^2)dt + (\eta\Sigma(X_t))^{1/2}dB_t \tag{16}$$

When μ is close to 0 (i.e., the expected delay is short), one would naively expect that SME-SGD (16) would give rise to a reasonable approximation to ASGD as well. However, Figures 1 and 2 demonstrate that it is not the case: only SME-ASGD proposed here results in accurate path approximations for both the first and the second moments (in particular when μ enlarges), while the trajectories obtained from SME-SGD are way off.

A few remarks regarding the numerical results are in order here. (i) In Figure 1, the path oscillations happen to both ASGD and SME-ASGD due to a longer expected delay, but not to SME-SGD, even though we include staleness when computing $\Sigma(X_t)$ for both models. That is because our SME-ASGD model contains μ in the forcing term, while the forcing term in SME-SGD is μ-independent. (ii) The convex function whose gradient $\nabla f(x) = 4x^3 + 12x$ in Figure 2 does not satisfy the general Ito conditions; however, by having good initial data and choosing smaller time step sizes, we can still obtain the minimizer without blowing up. (iii) Even for the non-convex function (double-well function in Figure 2), our SME-ASGD model is able to give a better prediction about which minimizer that a trajectory with given initial data will fall into; the percentage that SME-ASGD shows is very close to what in the ASGD case.



$\mu = 0.9$ $\mu = 0.95$ $\mu = 0.97$

Figure 1: Apply the SME-ASGD to minimize the quadratic function $f(x) = x^2$ in different μ's, with subfunctions $f_1(x) = (x-1)^2 - 1$, and $f_2(x) = (x+1)^2 - 1$. $x_0 = 1$ and $\eta = 1e-2$. SME-ASGD achieves more accurate approximations compared to SME-SGD (16), especially when μ becomes large. However, one can also observe that when μ increases the error of the SME-ASGD approximation increases as well.

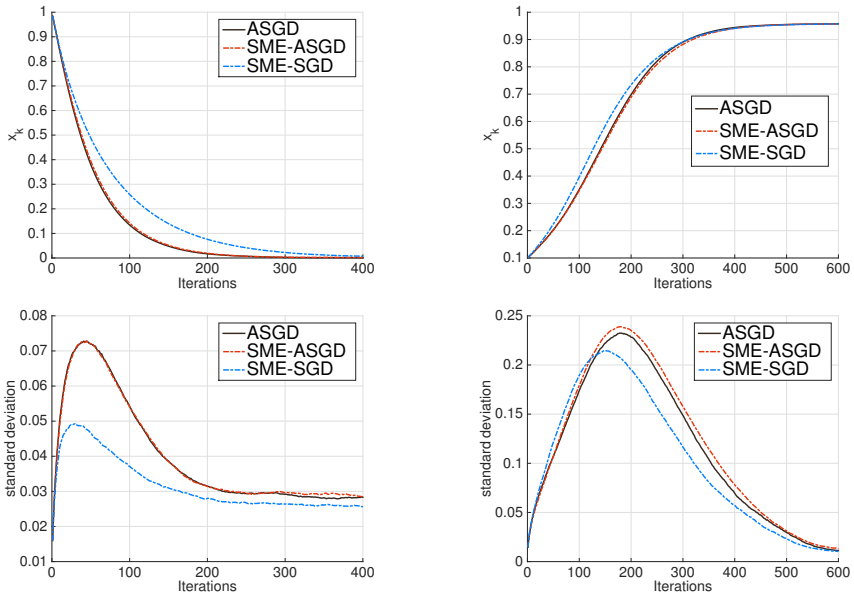


Figure 2: (Left) Apply the SME-ASGD to minimize the convex function $f(x) = x^4 + 6x^2$ with subfunctions $f_1(x) = (x-1)^4 - 1$, and $f_2(x) = (x+1)^4 - 1$. Notice that the gradients are Lipschitz locally. Here we choose $x_0 = 1$, and a smaller step size $\eta = 1e-3$. (Right) Apply the SME-ASGD to minimize the double well potential $f(x) = 1 - e^{-(x-1)^2} - e^{-(x+1)^2}$. Here $f_1 = 1 - 2e^{-(x-1)^2}$, $f_2 = 1 - 2e^{-(x+1)^2}$ and both have Lipschitz gradients. We choose $\eta = 1e-2, x_0 = 0.1$. Note that $\arg\min f(x) \approx \pm 0.9575$. In our case, due to the initial data x_0, 90.34% of ASGD path samples converge to 0.9575, while 90.50% of SME-ASGD and 88.54% of SME-SGD converge to the same minimizer. For both columns of numerical tests, we choose $\mu = 0.95$.

4 Optimal mini-batch size of ASGD

With much better understanding of dynamics of the ASGD algorithm using SME-ASGD, we are able to tune multiple hyper-parameters of ASGD using the predictions obtained from applying the stochastic optimal control theory to SME-ASGD. Here we demonstrate one such application: the optimal time-dependent mini-batch size for ASGD. By denoting the time-dependent batch size as $1 + u_k$ with $u_k \geq 0$, one can write the iteration as

$$x_{k+1} = x_k - \eta \frac{1}{1+u_k} \sum_{j=1}^{1+u_k} \nabla f_{\gamma_j}(x_{k-\tau_k}). \tag{17}$$

We may assume that the choice of mini-batch size is independent from γ_j and the staleness τ_k. This is because, even though changing the batch size will simultaneously change the clocks of all the processors, the staleness would not be changed as all the processors are impacted equally. Following the argument given in Section 2, we derive the corresponding SME

$$\begin{aligned} dY_t &= -\nabla f(X_t)dt - \sqrt{\frac{1-\mu}{\eta}} Y_t dt \\ dX_t &= Y_t dt + \frac{\sigma(t)\eta^{3/4}}{(1+u(t))^{1/2}(1-\mu)^{1/4}} dB_t. \end{aligned} \tag{18}$$

To simplify the discussion, let us consider for example the quadratic loss objective $f(x) = x^2$. By applying the Ito's formula to this SME, we obtain the following second moment equation (a similar

derivation is shown in Appendix C) for the evolution of the expected loss

$$\frac{d}{dt}\begin{bmatrix}\mathbb{E}(X_t^2)\\ \mathbb{E}(Y_t^2)\\ \mathbb{E}(X_tY_t)\end{bmatrix} = -\begin{bmatrix}0 & 0 & -2\\ 0 & 2\sqrt{(1-\mu)/\eta} & 4\\ 2 & -1 & \sqrt{(1-\mu)/\eta}\end{bmatrix}\begin{bmatrix}\mathbb{E}(X_t^2)\\ \mathbb{E}(Y_t^2)\\ \mathbb{E}(X_tY_t)\end{bmatrix} + \begin{bmatrix}\frac{\Sigma(t)\eta^{3/2}}{(1+u(t))(1-\mu)^{1/2}}\\ 0\\ 0\end{bmatrix} \tag{19}$$

As (19) is a linear system with constant coefficient matrix, its asymptotic behavior is determined by the eigenvalue of the coefficient matrix. An easy calculation shows that the eigenvalue with largest real part is given by $\lambda = -\sqrt{(1-\mu)/\eta} + \sqrt{(1-\mu-8\eta)/\eta}$, which has a negative real part, and thus the second moment of X_t decays exponentially. Moreover, (19) provides us with the stationary solution for X^2

$$z_\infty := \mathbb{E}(X_\infty^2) = \frac{\Sigma\eta}{2(1+u(t))}\Big(\frac{\eta}{1-\mu} + \frac{1}{2}\Big). \tag{20}$$

Rather than applying the optimal control subject to the full second moment equation, we shall work with a simpler evolution equation that asymptotically approximates the dynamics (imposed as a constraint). More specifically, we pose the following optimal control problem for the time-dependent mini-batch size

$$\begin{aligned}&\min_{u\in\mathcal{A}}\left\{z(T) + \frac{\gamma}{\eta}\int_0^T u(s)ds\right\} \quad \text{subject to}\\ &\frac{d}{dt}z(t) = \mathrm{Re}(\lambda)(z(t) - z_\infty) \qquad \text{with } z(0) = x_0^2,\end{aligned} \tag{21}$$

where $z(t)$ models $\mathbb{E}(X_t^2)$ – the quantity to minimize, $\mathcal{A} = \{u(t) \geq 0\}$ is an admissible control set as the mini-batch size is greater than 1, and $\gamma > 0$ is a constant measuring the unit cost for introducing extra gradient samples throughout the time. The solution to (21) is given by (detailed computation deferred to Appendix D), with $t^* = \frac{1}{\mathrm{Re}(\lambda)}\log\Big(\frac{\gamma}{\gamma^*}\Big)$, $\gamma^* = -\frac{\mathrm{Re}(\lambda)\Sigma\eta^2}{2}\big(\frac{\eta}{1-\mu} + \frac{1}{2}\big)$

$$u^*(t) = \begin{cases}0 & \text{if } \gamma > \gamma^* \text{ or } 0 \leq t \leq T - t^*\\ \sqrt{\frac{\gamma^*}{\gamma}}e^{\mathrm{Re}(\lambda)(T-t)/2} - 1 & \text{if } \gamma \leq \gamma^*, T - t^* < t \leq T.\end{cases} \tag{22}$$

In particular, (22) tells that we should use a small mini-batch size (even size 1) during the early time (for $k \leq k^* = (T - t^*)/\eta$), since during this period the gradient flow dominate the dynamics. After the transition time k^* when the noise starts to dominate, one shall apply mini-batch with size exponentially increasing in k to reduce the variance. Figure 3 demonstrates that our proposed mini-batching strategy outperforms the ASGD with a constant batch size (for example, applied in [3, 6]). Note that such strategy of increasing the batch size in later stage of training has been also suggested and used in recent works in training large neural networks, e.g., [7, 9].

5 Conclusion

In this paper, we have developed stochastic modified equations (SMEs) to model the asynchronous stochastic gradient descent (ASGD) algorithms in the continuous-time limit. When the gradient of the loss function is linear, the resulting SME can be put into a Langevin equation, whose solution is known to converge to the unique invariant measure with the convergence rate dictated by the corresponding temperature. We utilize such information to compare with the momentum SGD and prove the "asynchrony begets momentum" phenomenon. For the nonlinear gradient case, though the resulting SME does not have an explicitly known invariant measure, it still provides precise trajectory predictions for the discrete ASGD dynamics. Moreover, with SME available, we are able to find optimal hyper-parameters for ASGD algorithms by performing a moment analysis and leveraging the optimal control theory.

Acknowledgments

J.A. was partially supported by the Gene Golub Fellowship at ICME. J.L. is supported by the National Science Foundation under award DMS-1454939, and L.Y. is partially supported by the National Science Foundation under award DMS-1521830.

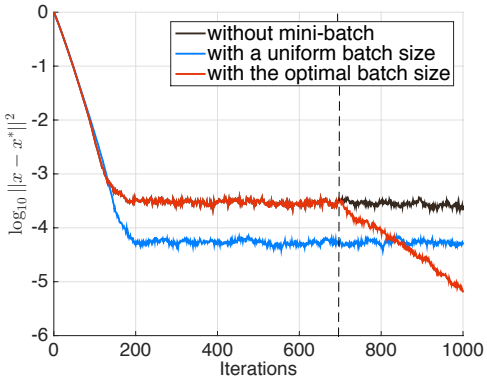


Figure 3: A comparison of performance in terms of l^2 error. We apply mini-batching over $n = 100$ subfunctions $f_i(x) = \frac{1}{2}(x - c_i)^2, c_i = -1/2 + i/(2n)$. Here we choose the step size $\eta = 0.02$ and the initial data $x_0 = 1$. The batch size for the uniform mini-batching case is 5. For the optimal mini-batching strategy, the transition happens at $k = (T - t^*)/\eta \approx 699$, and the optimized batch size at time T is 42. In practice, we can apply a more aggressive mini-batching strategy by starting to increase the batch size earlier in the flat region, and it will result in a larger batch size at T.

References

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

[2] Léon Bottou, Frank E Curtis, and Jorge Nocedal. Optimization methods for large-scale machine learning, 2016. preprint, arXiv:1606.04838.

[3] Ofer Dekel, Ran Gilad-Bachrach, Ohad Shamir, and Lin Xiao. Optimal distributed online prediction using mini-batches. *Journal of Machine Learning Research*, 13(Jan):165–202, 2012.

[4] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159, 2011.

[5] John Duchi, Michael I Jordan, and Brendan McMahan. Estimation, optimization, and parallelism when data is sparse. In *Advances in Neural Information Processing Systems*, pages 2832–2840, 2013.

[6] Kevin Gimpel, Dipanjan Das, and Noah A Smith. Distributed asynchronous online learning for natural language processing. In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning*, pages 213–222. Association for Computational Linguistics, 2010.

[7] P. Goyal, P. Dollar, R. Girshick, P. Noordhuis, L. Wesolowski, A. Kyrola, A. Tulloch, Y. Jia, and K. He. Accurate, large minibatch SGD: Training ImageNet in 1 hour, 2017. arXiv preprint, arXiv:1706.02677.

[8] Moritz Hardt, Ben Recht, and Yoram Singer. Train faster, generalize better: Stability of stochastic gradient descent. In *International Conference on Machine Learning*, pages 1225–1234, 2016.

[9] N. Keskar, D. Mudigere, J. Nocedal, M. Smelyanskiy, and P. T. P. Tang. On large-batch training for deep learning: Generalization gap and sharp minima. In *International Conference on Learning Representations*, 2017.

[10] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 2015.

[11] Peter E Kloeden and Eckhard Platen. Stochastic differential equations. In *Numerical Solution of Stochastic Differential Equations*, pages 103–160. Springer, 1992.

[12] Qianxiao Li, Cheng Tai, and E Weinan. Stochastic modified equations and adaptive stochastic gradient algorithms. In *International Conference on Machine Learning*, pages 2101–2110, 2017.

[13] Ji Liu and Stephen J Wright. Asynchronous stochastic coordinate descent: Parallelism and convergence properties. *SIAM Journal on Optimization*, 25(1):351–376, 2015.

[14] Ji Liu, Stephen J Wright, Christopher Ré, Victor Bittorf, and Srikrishna Sridhar. An asynchronous parallel stochastic coordinate descent algorithm. *The Journal of Machine Learning Research*, 16(1):285–322, 2015.

[15] Ioannis Mitliagkas, Ce Zhang, Stefan Hadjis, and Christopher Ré. Asynchrony begets momentum, with an application to deep learning. In *Communication, Control, and Computing (Allerton), 2016 54th Annual Allerton Conference on*, pages 997–1004. IEEE, 2016.

[16] Deanna Needell, Rachel Ward, and Nati Srebro. Stochastic gradient descent, weighted sampling, and the randomized kaczmarz algorithm. In *Advances in Neural Information Processing Systems*, pages 1017–1025, 2014.

[17] Yu Nesterov. Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM Journal on Optimization*, 22(2):341–362, 2012.

[18] Grigorios A Pavliotis. *Stochastic processes and applications: Diffusion processes, the Fokker-Planck and Langevin equations*, volume 60. Springer, 2014.

[19] Boris T Polyak. Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics*, 4(5):1–17, 1964.

[20] 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.

[21] Peter Richtárik and Martin Takáč. Iteration complexity of randomized block-coordinate descent methods for minimizing a composite function. *Mathematical Programming*, 144(1-2):1–38, 2014.

[22] Ilya Sutskever, James Martens, George Dahl, and Geoffrey Hinton. On the importance of initialization and momentum in deep learning. In *International conference on machine learning*, pages 1139–1147, 2013.

[23] Tijmen Tieleman and Geoffrey Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 4(2):26–31, 2012.

[24] Håkan Lorens Samir Younes. *Verification and Planning for Stochastic Processes with Asynchronous Events*. PhD thesis, Academy of Engineering Sciences, 2005.

A Appendix A: proof of Theorem 3

Proof. We look at the one step approximation in the base case, and the global approximation can be done by induction. Using the variation of constant formula, we know that the solution of

$$dY_t = -\nabla f(X_t)dt - \sqrt{\frac{1-\mu}{\eta}}Y_t dt$$

is given by

$$Y_t = e^{-\sqrt{\frac{1-\mu}{\eta}}t}Y_0 - e^{-\sqrt{\frac{1-\mu}{\eta}}t}\int_0^t \nabla f(X_s)ds$$

where $Y_0 = -\sqrt{\frac{\eta}{1-\mu}}\sum_{l=0}^{\infty}\nabla f(x_{-l})(1-\mu)\mu^l$ as defined in (10). Plugging Y_t into the integral form of $X_{\Delta t}$ gives rise to

$$X_{\Delta t} = x + \int_0^{\Delta t}\left(e^{-\sqrt{\frac{1-\mu}{\eta}}s}Y_0 - e^{-\sqrt{\frac{1-\mu}{\eta}}s}\int_0^s \nabla f(X_u)du\right)ds + \frac{\eta^{3/4}}{(1-\mu)^{1/4}}\int_0^{\Delta t}\sigma(s)dB_s \tag{23}$$

Splitting $\eta\nabla f_{\gamma_0}(v_0)$ into $\eta\nabla f_{\gamma_0}(v_0) - \eta\sum_{l=0}^{\infty}\nabla f(x_{-l})(1-\mu)\mu^l$ and $\eta\sum_{l=0}^{\infty}\nabla f(x_{-l})(1-\mu)\mu^l$, we can make the following estimate

$$\begin{aligned}
\mathbb{E}\{|X_{\Delta t} - x_1|\} &\le \left|\int_0^{\Delta t} e^{-\sqrt{\frac{1-\mu}{\eta}}s}Y_0 ds + \eta\sum_{l=0}^{\infty}\nabla f(x_{-l})(1-\mu)\mu^l\right| \\
&+ \mathbb{E}\left\{\int_0^{\Delta t} e^{-\sqrt{\frac{1-\mu}{\eta}}s}\left(\int_0^s |\nabla f(X_u) - \nabla f(x_1)|du\right)ds\right\} + |\nabla f(x_1)|\int_0^{\Delta t} e^{-\sqrt{\frac{1-\mu}{\eta}}s}sds \\
&+ \frac{\eta^{3/4}}{(1-\mu)^{1/4}}\left(\mathbb{E}\{(\int_0^{\Delta t}\sigma(s)dB_s)^2\}\right)^{1/2} + \mathbb{E}\{|\eta\nabla f_{\gamma_0}(v_0) - \eta\sum_{l=0}^{\infty}\nabla f(x_{-l})(1-\mu)\mu^l|\} \\
&\le I + II + III + \frac{\eta^{3/4}}{(1-\mu)^{1/4}}\left(\mathbb{E}\{\int_0^{\Delta t}\sigma(s)^2 ds\}\right)^{1/2} + c\eta \le I + II + III + 2c\frac{\Delta t^2}{1-\mu}.
\end{aligned}$$

In the above derivation, we have applied the Ito isometry to the fourth term and then used

$$\frac{\eta^{3/4}}{(1-\mu)^{1/4}}\left(\mathbb{E}\{\int_0^{\Delta t}\sigma(s)^2 ds\}\right)^{1/2} \le c\frac{\Delta t^2}{1-\mu},$$

since $\Delta t = \sqrt{\eta(1-\mu)}$. The fifth term, after applying the Cauchy-Schwarz inequality, is shown to be a discrete version of the covariance matrix

$$\mathbb{E}\{|\eta\nabla f_{\gamma_0}(v_0) - \eta\sum_{l=0}^{\infty}\nabla f(x_{-l})(1-\mu)\mu^l|\} \le \eta\sqrt{\Sigma_0} \le c\eta.$$

Moreover,

$$\begin{aligned}
I &= \left|\int_0^{\Delta t} e^{-\sqrt{\frac{1-\mu}{\eta}}s}Y_0 ds + \eta\sum_{l=0}^{\infty}\nabla f(x_{-l})(1-\mu)\mu^l\right| \\
&= \left|\sqrt{\frac{\eta}{1-\mu}}(e^{-\sqrt{\frac{1-\mu}{\eta}}\Delta t} - 1)\sqrt{\frac{\eta}{1-\mu}}\sum_{l=0}^{\infty}\nabla f(x_{-l})(1-\mu)\mu^l + \eta\sum_{l=0}^{\infty}\nabla f(x_{-l})(1-\mu)\mu^l\right| \\
&= \left|-\sqrt{\frac{\eta}{1-\mu}}\sum_{l=0}^{\infty}\nabla f(x_{-l})(1-\mu)\mu^l\Delta t + \eta\sum_{l=0}^{\infty}\nabla f(x_{-l})(1-\mu)\mu^l + O(\Delta t^2)\right| = O(\Delta t^2),
\end{aligned}$$

since the first two terms cancel. Because ∇f is Lipschitz, we can estimate the second term by

$$II \le L\Delta t\int_0^{\Delta t}\mathbb{E}\{|X_u - x_1|\}du \le LT\int_0^{\Delta t}\mathbb{E}\{|X_u - x_1|\}du.$$

Because x_1 stays in a bounded domain, the third term can be bounded by

$$III \leq |\nabla f(x_1)|\Delta t\sqrt{\frac{\eta}{1-\mu}}(1-e^{-\sqrt{\frac{1-\mu}{\eta}}\Delta t}) = |\nabla f(x_1)|\Delta t^2 + O(\Delta t^3) = O(\Delta t^2).$$

With these estimates available, we can choose a sufficiently large constant C (depending on c and the size of the domain containing the iterates from ASGD) such that

$$\mathbb{E}\big\{|X_{\Delta t} - x_1|\big\} \leq C\frac{\Delta t^2}{1-\mu} + LT\int_0^{\Delta t}\mathbb{E}\big\{|X_u - x_1|\big\}du.$$

By Gronwall's inequality, we have

$$\mathbb{E}\big\{|X_{\Delta t} - x_1|\big\} \leq C\frac{\Delta t^2}{1-\mu}e^{LT\Delta t} \to C\frac{\Delta t^2}{1-\mu}$$

as $\Delta t \to 0$.

The induction step is similar. With the assumption $\mathbb{E}\big\{|X_{k\Delta t} - x_k|\big\} \leq Ck\frac{\Delta t^2}{1-\mu}$, we have the following estimate

$$\begin{aligned}
\mathbb{E}\big\{|X_{(k+1)\Delta t} - x_{k+1}|\big\} \leq{}& \mathbb{E}\big\{|X_{k\Delta t} - x_k|\big\} + \left|\int_{k\Delta t}^{(k+1)\Delta t} e^{-\sqrt{\frac{1-\mu}{\eta}}s}Y_{k\Delta t}ds + \eta\sum_{l=0}^{\infty}\nabla f(x_{k-l})(1-\mu)\mu^l\right| \\
&+ \mathbb{E}\bigg\{\int_{k\Delta t}^{(k+1)\Delta t} e^{-\sqrt{\frac{1-\mu}{\eta}}s}\bigg(\int_{k\Delta t}^{s}\big|\nabla f(X_u) - \nabla f(x_{k+1})\big|du\bigg)ds\bigg\} + |\nabla f(x_{k+1})|\int_{k\Delta t}^{(k+1)\Delta t} e^{-\sqrt{\frac{1-\mu}{\eta}}s}sds \\
&+ \frac{\eta^{3/4}}{(1-\mu)^{1/4}}\bigg(\mathbb{E}\big\{\big(\int_{k\Delta t}^{(k+1)\Delta t}\sigma(s)dB_s\big)^2\big\}\bigg)^{1/2} + \mathbb{E}\big\{\big|\eta\nabla f_{\gamma_k}(v_k) - \eta\sum_{l=0}^{\infty}\nabla f(x_{k-l})(1-\mu)\mu^l\big|\big\}.
\end{aligned}$$

Here the only difference is $Y_{k\Delta t}$ in the second term, which is not given but generated from SME. Denote $y_k = -\sqrt{\frac{\eta}{1-\mu}}\sum_{l=0}^{\infty}\nabla f(x_{k-l})(1-\mu)\mu^l$. From (11), we observe that y_k is indeed an approximation of Y_t by applying the Euler discretization to the ordinary differential equation part of the SME. Because the global truncation error for the Euler method in ODE is $O(\Delta t)$, we have

$$\begin{aligned}
\left|\int_{k\Delta t}^{(k+1)\Delta t} e^{-\sqrt{\frac{1-\mu}{\eta}}s}Y_{k\Delta t}ds + \eta\sum_{l=0}^{\infty}\nabla f(x_{k-l})(1-\mu)\mu^l\right| \leq{}& \int_{k\Delta t}^{(k+1)\Delta t} e^{-\sqrt{\frac{1-\mu}{\eta}}s}\big|Y_{k\Delta t} - y_k\big|ds \\
&+ \left|\int_{k\Delta t}^{(k+1)\Delta t} e^{-\sqrt{\frac{1-\mu}{\eta}}s}y_k ds + \eta\sum_{l=0}^{\infty}\nabla f(x_{k-l})(1-\mu)\mu^l\right| \leq O(\Delta t^2)
\end{aligned}$$

as shown before. Applying the Gronwall's inequality again and letting $\Delta t \to 0$, we obtain

$$\mathbb{E}\big\{|X_{(k+1)\Delta t} - x_{k+1}|\big\} \leq Ck\frac{\Delta t^2}{1-\mu}.$$

As $n\Delta t \leq T$ for all n, one can conclude that there exist $K_T > 0$ such that

$$\mathbb{E}\big\{|X_{n\Delta t} - x_n|\big\} \leq K_T\frac{\Delta t}{1-\mu}.$$

□

B Appendix B: miscellaneous computations in SMEs

In this section, we provide the missing computations in Section 2. First, let us show that in subsection 2.2, the noise term $-\frac{y_k}{\alpha} - \nabla f_{\gamma_k}(x_{k-\tau_k})$ has mean 0

$$\begin{aligned}
\mathbb{E}\Big(-\frac{y_k}{\alpha} - \nabla f_{\gamma_k}(x_{k-\tau_k})\Big) &= \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\bigg(\sum_{l=0}^{\infty}\nabla f(x_{k-l})(1-\mu)\mu^l - \nabla f_i(x_{k-\tau_k})\bigg) \\
&= \mathbb{E}\bigg(\sum_{l=0}^{\infty}\nabla f(x_{k-l})(1-\mu)\mu^l - \nabla f(x_{k-\tau_k})\bigg) \\
&= \mathbb{E}\bigg(\sum_{m=0}^{\infty}(1-\mu)\mu^m\big(\sum_{l=0}^{\infty}\nabla f(x_{k-l})(1-\mu)\mu^l - \nabla f(x_{k-m})\big)\bigg) \\
&= \mathbb{E}\bigg(\sum_{l=0}^{\infty}\nabla f(x_{k-l})(1-\mu)\mu^l - \sum_{m=0}^{\infty}\nabla f(x_{k-m})(1-\mu)\mu^m\bigg) = 0
\end{aligned}$$

The covariance matrix conditioned on $x_{k-l}, l = 0, 1, 2, \cdots$ is given by

$$\begin{aligned}
\Sigma_k &= \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\left(\Big(-\frac{y_k}{\alpha} - \nabla f_i(x_{k-\tau_k})\Big)\Big(-\frac{y_k}{\alpha} - \nabla f_i(x_{k-\tau_k})\Big)^T\right) \\
&= \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\left(\left(\sum_{l=0}^{\infty}\nabla f(x_{k-l})(1-\mu)\mu^l - \nabla f_i(x_{k-\tau_k})\right)\left(\sum_{l=0}^{\infty}\nabla f(x_{k-l})(1-\mu)\mu^l - \nabla f_i(x_{k-\tau_k})\right)^T\right).
\end{aligned}$$

B.1 Evolution equation of Σ in (14)

First, we have

$$\Sigma_k = \mathbb{E}\bigg((-\frac{y_k}{\alpha} - \nabla f_{\gamma_k}(x_{k-\tau_k}))(-\frac{y_k}{\alpha} - \nabla f_{\gamma_k}(x_{k-\tau_k}))^T\bigg).$$

By expanding the terms in the expectation and treating them individually, we arrive at the following

$$\begin{aligned}
\Sigma_k &= \frac{1}{\alpha^2}y_k y_k^T + \frac{y_k}{\alpha}\mathbb{E}\{\nabla f_{\gamma_k}(x_{k-\tau_k})^T\} + \mathbb{E}\{\nabla f_{\gamma_k}(x_{k-\tau_k})\}\frac{y_k^T}{\alpha} + \mathbb{E}\{\nabla f_{\gamma_k}(v_k)\nabla f_{\gamma_k}(x_{k-\tau_k})^T\} \\
&= \mathbb{E}\{\nabla f_{\gamma_k}(x_{k-\tau_k})\nabla f_{\gamma_k}(x_{k-\tau_k})^T\} - \frac{1}{\alpha^2}y_k y_k^T \\
&= \mu\sum_{m=0}^{\infty}\mathbb{E}\{\nabla f_{\gamma_{k-1}}(x_{k-1-m})\nabla f_{\gamma_{k-1}}(x_{k-1-m})^T\}(1-\mu)\mu^m \\
&\quad + (1-\mu)\mathbb{E}\{\nabla f_{\gamma_k}(x_k)\nabla f_{\gamma_k}(x_k)^T\} - \frac{1}{\alpha^2}y_k y_k^T \\
&= \mu\big(\Sigma_{k-1} + \frac{1}{\alpha^2}y_{k-1}y_{k-1}^T\big) + (1-\mu)\mathbb{E}\{\nabla f_{\gamma_k}(x_k)\nabla f_{\gamma_k}(x_k)^T\} - \frac{1}{\alpha^2}y_k y_k^T \\
&= \mu\big(\Sigma_{k-1} + \frac{1}{\alpha^2}y_{k-1}y_{k-1}^T\big) + \frac{1-\mu}{n}\sum_{i=1}^{n}\nabla f_i(x_k)\nabla f_i(x_k)^T - \frac{1}{\alpha^2}y_k y_k^T.
\end{aligned} \tag{24}$$

Notice that $y_k = \mu y_{k-1} - \alpha(1-\mu)\nabla f(x_k)$, and thus we have

$$\begin{aligned}
y_{k-1}y_{k-1}^T &= \frac{1}{\mu^2}\big(y_k + \alpha(1-\mu)\nabla f(x_k)\big)\big(y_k + \alpha(1-\mu)\nabla f(x_k)\big)^T \\
&= \frac{1}{\mu^2}\bigg(y_k y_k^T + \alpha(1-\mu)y_k\nabla f(x_k)^T + \alpha(1-\mu)\nabla f(x_k)y_k^T + \alpha^2(1-\mu)^2\nabla f(x_k)\nabla f(x_k)^T\bigg).
\end{aligned}$$

Substituting it in (24), we obtain

$$\frac{\Sigma_k - \Sigma_{k-1}}{\alpha(1-\mu)} = -\frac{1}{\alpha}\Sigma_{k-1} + \frac{1}{\alpha^3\mu}y_k y_k^T + \frac{1}{\alpha^2\mu}y_k \nabla f(x_k)^T + \frac{1}{\alpha^2\mu}\nabla f(x_k) y_k^T + \frac{1-\mu}{\alpha\mu}\nabla f(x_k)\nabla f(x_k)^T + \frac{1}{\alpha n}\sum_{i=1}^{n}\nabla f_i(x_k)\nabla f_i(x_k)^T.$$

Using this and $\Delta t = \alpha(1-\mu), \alpha = \sqrt{\frac{\eta}{1-\mu}}$, we obtain the evolution equation (14).

B.2 Transformation between (6) and (13)

When ∇f is linear,

$$y_k = -\alpha \nabla f(\sum_{l=0}^{\infty} x_{k-l}(1-\mu)\mu^l) = -\alpha \nabla f(m_k).$$

Replacing y_{k+1} and y_k with the above formula and also x_{k+1} with $\frac{m_{k+1}-\mu m_k}{1-\mu}$, we can rewrite (11) as

$$-\frac{\nabla f(m_{k+1}) - \nabla f(m_k)}{1-\mu} = \nabla f(m_k) - \nabla f(\frac{m_{k+1}-\mu m_k}{1-\mu}) = -\frac{1}{1-\mu}\nabla f(m_{k+1}-m_k).$$

Since $p_{k+1} = (m_{k+1}-m_k)/\sqrt{\eta(1-\mu)}$, we have

$$\nabla f(m_{k+1}-m_k) = \nabla f(p_{k+1}\sqrt{\eta(1-\mu)}),$$

which implies (5). To show (4), we first notice that

$$\begin{aligned}\frac{x_{k+1}-x_k}{\eta/\alpha} &= \frac{m_{k+1}-(\mu+1)m_k+\mu m_{k-1}}{(1-\mu)\eta/\alpha} = \frac{m_{k+1}-2m_k+m_{k-1}}{(1-\mu)\eta/\alpha} + \frac{m_k-m_{k-1}}{\eta/\alpha} \\ &= \frac{p_{k+1}-p_k}{1-\mu} + p_k = -\alpha\nabla f(m_k) + \alpha(\nabla f(m_k) - \nabla f_{\gamma_k}(v_k)) \\ &= -\sqrt{\frac{\eta}{1-\mu}}\nabla f(m_k) + \sqrt{\frac{\eta}{1-\mu}}(\nabla f(m_k) - \nabla f_{\gamma_k}(v_k))\end{aligned}$$

by plugging in α in terms of μ, η. It is clear now that this gives (4).

B.3 SME for SGD with momentum

Recall the iteration for the SGD with a constant momentum parameter is

$$\begin{aligned}v_{k+1} &= \mu' v_k - \eta' \nabla f_{\gamma_k}(x_k) \\ x_{k+1} &= x_k + v_{k+1},\end{aligned}$$

which can be viewed as a second-order difference equation. To ensure the final equation with all terms of order $O(1)$, one needs $\eta' = (\Delta t)^2$. We can rewrite (8) as

$$\begin{aligned}\frac{v_{k+1}}{\sqrt{\eta'}} &= \frac{v_k}{\sqrt{\eta'}} + \sqrt{\eta'}\left(-\frac{1-\mu'}{\eta'}v_k - \nabla f(x_k)\right) + \sqrt{\eta'}(\nabla f(x_k) - \nabla f_{\gamma_k}(x_k)) \\ x_{k+1} &= x_k + \frac{v_{k+1}}{\sqrt{\eta'}}\sqrt{\eta'}.\end{aligned} \tag{25}$$

Let us introduce $p = v/\sqrt{\eta'}$. In order to have $\sqrt{\eta'}(\nabla f(x_k) - \nabla_{\gamma_k} f(x_k)) \sim c\Delta B_t$, we choose $c \sim \sigma(\eta')^{1/4}$. Therefore, we obtain the first order weak approximation, which can also be viewed as the Euler-Maruyama discretization of the following SDE

$$\begin{aligned}dP_t &= -\nabla f(X_t)dt - \frac{1-\mu'}{\sqrt{\eta'}}P_t dt + \sigma(X_t)(\eta')^{\frac{1}{4}}dB_t \\ dX_t &= P_t dt.\end{aligned}$$

C Appendix C: dynamics of SME-ASGD (13)

We consider the one dimensional case with $f(x) = \frac{1}{2}ax^2$. The goal here is to give an analysis of the dynamics of first and second moment of X and Y under (13). Taking expectation, we obtain

$$d\begin{bmatrix}\mathbb{E}(Y_t)\\ \mathbb{E}(X_t)\end{bmatrix} = \begin{bmatrix}-\sqrt{\frac{1-\mu}{\eta}} & -a\\ 1 & 0\end{bmatrix}\begin{bmatrix}\mathbb{E}(Y_t)\\ \mathbb{E}(X_t)\end{bmatrix}dt = A(\mu,\eta)\begin{bmatrix}\mathbb{E}(Y_t)\\ \mathbb{E}(X_t)\end{bmatrix}dt.$$

One observes that the eigenvalues of $A(\mu,\eta)$ are $\lambda_{1,2}(A) = \frac{1}{2}\bigg(-\sqrt{\frac{1-\mu}{\eta}} \pm \sqrt{\frac{1-\mu}{\eta} - 4a}\bigg)$, the real parts of both are negative as long as $a > 0$. From this, we conclude that, when $a > 0$, the expectation of X_t decays exponentially. The corresponding stationary solutions are given by

$$\mathbb{E}(X_\infty) = \mathbb{E}(Y_\infty) = 0.$$

For the second moment, we end up with the following equations by using the Ito's formula

$$\begin{aligned}
d\mathbb{E}(X_t^2) &= 2\mathbb{E}(X_tY_t)dt + \Sigma(t)\frac{\eta^{3/2}}{(1-\mu)^{1/2}}dt\\
d\mathbb{E}(Y_t^2) &= -2a\mathbb{E}(X_tY_t)dt - 2\sqrt{\frac{1-\mu}{\eta}}\mathbb{E}(Y_t^2)dt\\
d\mathbb{E}(X_tY_t) &= -a\mathbb{E}(X_t^2)dt + \mathbb{E}(Y_t^2)dt - \sqrt{\frac{1-\mu}{\eta}}\mathbb{E}(X_tY_t)dt.
\end{aligned} \tag{26}$$

In order to study the behavior of the second moments, we can rewrite (26) as

$$d\begin{bmatrix}\mathbb{E}(X_t^2)\\ \mathbb{E}(Y_t^2)\\ \mathbb{E}(X_tY_t)\end{bmatrix} = \begin{bmatrix}0 & 0 & 2\\ 0 & -2\sqrt{\frac{1-\mu}{\eta}} & -2a\\ -a & 1 & -\sqrt{\frac{1-\mu}{\eta}}\end{bmatrix}\begin{bmatrix}\mathbb{E}(X_t^2)\\ \mathbb{E}(Y_t^2)\\ \mathbb{E}(X_tY_t)\end{bmatrix}dt + \begin{bmatrix}\Sigma(t)\frac{\eta^{3/2}}{(1-\mu)^{1/2}}\\ 0\\ 0\end{bmatrix}dt. \tag{27}$$

The corresponding stationary solutions are

$$\mathbb{E}(X_\infty Y_\infty) = \frac{-\Sigma\eta^{3/2}}{2(1-\mu)^{1/2}},\ \mathbb{E}(Y_\infty^2) = \frac{a\Sigma\eta^2}{2(1-\mu)},\ \text{ and } \mathbb{E}(X_\infty^2) = \frac{\Sigma\eta^2}{2(1-\mu)} + \frac{\Sigma\eta}{2a}.$$

Let us introduce

$$B(\mu,\eta) = \begin{bmatrix}0 & 0 & 2\\ 0 & -2\sqrt{\frac{1-\mu}{\eta}} & -2a\\ -a & 1 & -\sqrt{\frac{1-\mu}{\eta}}\end{bmatrix}.$$

The eigenvalues of $B(\mu,\eta)$ are

$$\lambda_1 = -\sqrt{\frac{1-\mu}{\eta}}, \lambda_{2,3} = \lambda_\pm = -\sqrt{\frac{1-\mu}{\eta}} \pm \sqrt{\frac{1-\mu}{\eta} - 4a}.$$

We can see that the real parts of all roots are negative as long as $a > 0$. Moreover, the second moment of X_t decays exponentially, with the rate given by $\mathrm{Re}(\lambda_+)$ since λ_+ is the eigenvalue with the largest (negative) real part. We obtain the largest descent rate $\mathrm{Re}(\lambda_+)$ when the second part $\sqrt{\frac{1-\mu}{\eta} - 4a}$ in λ_+ is purely imaginary, i.e., when μ takes

$$\mu_{\text{opt}} = \max\{1 - 4a\eta, 0\}. \tag{28}$$

We note that (28) also gives a suggestion to choose optimal step size η: when μ is given, the maximal step size we can choose is $\eta_{\text{opt}} = \frac{1-\mu}{4a}$. Any step size beyond that will cause oscillations in the SME and the corresponding ASGD.

D Appendix D: detailed computation in optimal mini-batching

We first justify the derivation of (18). Notice that it is not much different from derivation of SME-ASGD (13), except for the noise term $\frac{\eta}{1+u_k}\sum_{j=1}^{1+u_k}(-\frac{y_k}{\alpha}-\nabla f_{\gamma_j}(x_{k-\tau_k}))\sim c\Delta B_t$. With an extra that $\Sigma=O(1)$, we follow the same argument used in the error analysis in Appendix A and obtain

$$\mathbb{E}\left\{\frac{\eta^2}{(1+u_k)^2}\left(\sum_{j=1}^{1+u_k}(-\frac{y_k}{\alpha}-\nabla f_{\gamma_j}(x_{k-\tau_k})\right)\left(\sum_{j=1}^{1+u_k}(-\frac{y_k}{\alpha}-\nabla f_{\gamma_j}(x_{k-\tau_k})\right)^T\right\}$$
$$=\frac{\eta^2}{(1+u_k)^2}\sum_{j=1}^{1+u_k}\mathbb{E}\left\{\left(-\frac{y_k}{\alpha}-\nabla f_{\gamma_j}(x_{k-\tau_k})\right)\left(-\frac{y_k}{\alpha}-\nabla f_{\gamma_j}(x_{k-\tau_k})\right)^T\right\}=\frac{\eta^2}{1+u_k}\Sigma\sim c^2\Delta t$$

as the cross terms vanish under the expectation. Plugging in $\Delta t=\sqrt{\eta(1-\mu)}$, one sees that the coefficient for the noise is

$$c=\frac{\sigma(t)\eta^{3/4}}{(1+u(t))^{1/2}(1-\mu)^{1/4}}.$$

So the dynamics of SME here are similar to the example in Appendix C, except that we replace all Σ by $\Sigma/(1+u)$ in the mini-batching case. Next, we show how to solve the optimal control problem (21). The value function can be defined as

$$V(z,t)=\min_{u\in\mathcal{A}}\left\{z(T)+\frac{\gamma}{\eta}\int_t^T u(s)ds\;\middle|\;\frac{d}{dt}z(t)=F(u(t),z(t)),z(t)=z\right\},\tag{29}$$

where $F(u(t),z(t))=\text{Re}(\lambda)(z(t)-z_\infty)=\text{Re}(\lambda)\big(z(t)-\frac{\Sigma\eta}{2(1+u(t))}(\frac{\eta}{1-\mu}+\frac{1}{2})\big)$, and $\lambda=\lambda_+$, as what we computed in Appendix C. The corresponding Hamilton-Jacobi-Bellman equation is

$$\begin{aligned}&V_t+\min_{u\in\mathcal{A}}\left\{F(u,z)V_z+\frac{\gamma}{\eta}u\right\}=0\\&\text{with } V(0,t)=0, V(z,T)=z.\end{aligned}\tag{30}$$

Since $\min_{u\in\mathcal{A}}\left\{F(u,z)V_z+\frac{\gamma}{\eta}u\right\}=\min_{u\in\mathcal{A}}\left\{\frac{-V_z\text{Re}(\lambda)\Sigma\eta}{2(1+u)}\big(\frac{\eta}{1-\mu}+\frac{1}{2}\big)+\frac{\gamma}{\eta}u\right\}$, $V_z\geq 0$, and $\text{Re}(\lambda)<0$, the minimum could be obtained by solving the following equation

$$\frac{V_z\text{Re}(\lambda)\Sigma\eta}{2(1+u)^2}\big(\frac{\eta}{1-\mu}+\frac{1}{2}\big)+\frac{\gamma}{\eta}=0$$

with the derivative of the value function V_z to be determined later. Therefore the optimal batch size u^* as a function of V_z is

$$u^*(V_z)=\begin{cases}\sqrt{\frac{-V_z\text{Re}(\lambda)\Sigma\eta^2}{2\gamma}\big(\frac{\eta}{1-\mu}+\frac{1}{2}\big)}-1 & \text{if } \frac{-V_z\text{Re}(\lambda)\Sigma\eta^2}{2\gamma}\big(\frac{\eta}{1-\mu}+\frac{1}{2}\big)>1\\0 & \text{otherwise.}\end{cases}\tag{31}$$

The next step is to solve V to get an explicit formula for u^*. Placing $u^*(V_z)$ back into the minimization bracket, we obtain

$$\min_{u\in\mathcal{A}}\left\{F(u,z)V_z+\frac{\gamma}{\eta}u\right\}=\begin{cases}\text{Re}(\lambda)zV_z-\frac{\gamma}{\eta} & \text{if } \frac{-V_z\text{Re}(\lambda)\Sigma\eta^2}{2\gamma}\big(\frac{\eta}{1-\mu}+\frac{1}{2}\big)>1\\\text{Re}(\lambda)\big(z-\frac{\Sigma\eta}{2}\big(\frac{\eta}{1-\mu}+\frac{1}{2}\big)\big)V_z & \text{otherwise.}\end{cases}\tag{32}$$

This gives the Hamilton-Jacobi equation and we can solve it by using the method of characteristics. Let $\gamma^*=-\frac{\text{Re}(\lambda)\Sigma\eta^2}{2}\big(\frac{\eta}{1-\mu}+\frac{1}{2}\big)$ for notation convenience, we obtain the solution for V

$$V(z,t)=\begin{cases}\frac{\Sigma\eta}{2}\big(\frac{\eta}{1-\mu}+\frac{1}{2}\big)+\big(z-\frac{\Sigma\eta}{2}\big(\frac{\eta}{1-\mu}+\frac{1}{2}\big)\big)e^{\text{Re}(\lambda)(T-t)} & \text{if } \gamma>\gamma^*\\\big(z-\frac{\Sigma\eta}{2}\big(\frac{\eta}{1-\mu}+\frac{1}{2}\big)\big)e^{\text{Re}(\lambda)(T-t)}-\frac{\gamma}{\eta}(t^*+\frac{1}{\text{Re}(\lambda)}) & \text{if } \gamma\leq\gamma^*, 0\leq t\leq T-t^*\\ze^{\text{Re}(\lambda)(T-t)}-\frac{\gamma}{\eta}(T-t) & \text{if } \gamma\leq\gamma^*, T-t^*<t\leq T,\end{cases}\tag{33}$$

where

$$t^* = \frac{1}{\text{Re}(\lambda)} \log\left(\frac{\gamma}{\gamma^*}\right). \tag{34}$$

For all cases, $V_z = e^{\text{Re}(\lambda)(T-t)}$. With this inserted back into (31), we conclude that

$$u^*(t) = \begin{cases} 0 & \text{if } \gamma > \gamma^* \\ 0 & \text{if } \gamma \leq \gamma^*, 0 \leq t \leq T - t^* \\ \sqrt{\frac{-\text{Re}(\lambda)\Sigma\eta^2}{2\gamma}\left(\frac{\eta}{1-\mu} + \frac{1}{2}\right)} e^{\text{Re}(\lambda)(T-t)/2} - 1 & \text{if } \gamma \leq \gamma^*, T - t^* < t \leq T. \end{cases} \tag{35}$$