Information and Inference: A Journal of the IMA (2019) **00**, 1–23
doi:10.1093/imaiai/iaz030

Stochastic modified equations for the asynchronous stochastic gradient descent

JING AN†
Institute for Computational and Mathematical Engineering, Stanford University, Stanford, CA 94305, USA
†Corresponding author. Email: jingan@stanford.edu

JIANFENG LU
Department of Mathematics, Department of Chemistry and Department of Physics, Duke University, Box 90320, USA
jianfeng@math.duke.edu

AND

LEXING YING
Department of Mathematics and Institute for Computational and Mathematical Engineering (ICME), Stanford University, Stanford, CA 94305, USA
lexing@stanford.edu

[Received on 16 October 2018; revised on 19 July 2019; accepted on 22 September 2019]

We propose stochastic modified equations (SMEs) for modelling 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, we propose an optimal mini-batching strategy for ASGD via solving the optimal control problem of the associated SME.

Keywords: stochastic modified equations; asynchronous stochastic gradient descent; optimal control.

1. Introduction

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

$$\min_{x\in\mathbb{R}^d} f(x) := \frac{1}{n}\sum_{i=1}^{n} f_i(x), \tag{1.1}$$

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 samples one random instance $f_i(\cdot)$ uniformly 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 [9,17]. 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 [1,21].

1.1 *Related work*

There has been a vast literature on the analysis of SGD; see, for example, Bottou *et al.* [3] for a comprehensive review of this subject. Some widely used methods include AdaGrad [5], which extends SGD by adapting step sizes for different features; RMSProp [24], which resolves AdaGrad's rapidly diminishing learning rates issue; and Adam [11], 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 [21] 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_is all Lipschitz, it showed that the convergence rate can be similar to the synchronous case. Duchi *et al.* [6] 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.* [16] observed that a standard queuing model of asynchrony correlates to the momentum, that is, asynchrony produces momentum in SGD updates. There are also several methods using asynchrony either in parallel or in a distributed way, such as asynchronous stochastic coordinate descent algorithms [14,15,18,22].

Recently, Li *et al.* [13] 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 analyses that give upper bounds for (strongly) convex objects, this new framework not only provides more precise analyses for the leading order dynamics of SGD, but also suggests adaptive hyper-parameter strategies using optimal control theory.

1.2 *Our contributions*

We give a novel derivation of SMEs for the ASGD algorithms by introducing auxiliary variables to treat an effective memory term. With the derived SME models, we are able 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 each loss function f_i is quadratic. The derivation results in a Langevin equation, which by assuming its ergodicity 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 behaviour as compared to the momentum-based methods [16]. 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 objective functions are quadratic. We remark that the presented results make use of a few simplifying approximations that are made in a non-rigorous and non-quantified manner, e.g., assuming the noise coefficients to be constant σ and the accumulation of i.i.d. noise.

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 or f_i, but only require their gradients to be (uniformly) Lipschitz. Numerical results including nonlinear forcing terms and non-convex objectives demonstrate that SME-ASGD provides much more accurate predictions for the behaviour of ASGD compared to SME-SGD derived in [13]. 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. SMEs

The ASGD carries out the following update at each step:

$$x_{k+1} = x_k - \eta \nabla f_{\gamma_k}(x_{k-\tau_k}), \tag{2.1}$$

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 the delayed read of the parameter x used to update x_{k+1} with a random *staleness* τ_k.

Assumption 2.1 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_is are all (uniformly) Lipschitz, that is, for each $1 \leqslant i \leqslant n$, there exists $L_i > 0$ such that for any $x, y \in \mathbb{R}^d$, we have $|\nabla f_i(x) - \nabla f_i(y)| \leqslant L_i|x - y|$. As a consequence, by taking $L = \frac{1}{n}\sum_{i=1}^{n} L_i$, ∇f is also (uniformly) Lipschitz: $|\nabla f(x) - \nabla f(y)| \leqslant L|x - y|$. In addition, the staleness process τ_k follows the geometric distribution: $\tau_k = l$ (i.e., $x_{k-\tau_k} = x_{k-l}$), $l \in \{0, 1, 2, \cdots\}$, with probability $(1-\mu)\mu^l$ for $\mu \in (0, 1)$.

Assumption 2.2 We assume that the equations SME-ASGD (2.5) and SME-MSGD (2.7) are ergodic.

The geometric distribution assumption here is not only made to simplify the computation, but also can be justified by considering the canonical queuing model [25]. For example, the computation at each processor may involve a randomized algorithm that requires each processor to do multiple independent trials until the result is accepted, thus resulting in a geometrically distributed computation time. The geometric staleness assumption has been used in the previous asynchrony analysis; for example, see [16]. Our derivation of SME models can be also easily generalized to other random staleness models if the memory kernel, i.e., the distribution of staleness in time, decays sufficiently fast for integrability and is completely monotone when we approximate the memory kernel by a $C^\infty(0,\infty)$ function $\kappa(r)$. $\kappa(r)$ is completely monotone if for all for $n \geqslant 0, r > 0, (-1)^n \frac{\mathrm{d}^n}{\mathrm{d}r^n}\kappa(r) \geqslant 0$. Under that circumstance, we can approximate the kernel accurately by $\sum_{k=1}^{n_k} c_k\, \mathrm{e}^{-\lambda_k r}$ using the Bernstein's theorem of monotone functions [2], and each term can be embedded into one auxiliary value to derive the SME formulation.

2.1 *Linear gradients*

We first show the derivation of Langevin dynamics with the linear forcing term. Suppose that, for each $1 \leqslant i \leqslant n$, ∇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 SME for the asynchronous algorithm.

A key quantity for our derivation is the *expected read* m_k defined as the expectation of x_k following Assumption 2.1:

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

Here m_k is a conditional expectation conditioned on the history of x, and m_k is random since x_{k-l}s are. 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 (2.1), we can rewrite ASGD as

$$\frac{m_{k+1} - 2m_k + m_{k-1}}{\eta(1-\mu)} = -\frac{m_k - m_{k-1}}{\eta} - \nabla f(m_k) + (\nabla f(m_k) - \nabla f_{\gamma_k}(x_{k-\tau_k})). \tag{2.2}$$

The left-hand side and the first term on the right-hand side of (2.2) 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, conditioned on the history of updates, can be decomposed as

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

The covariance matrix of the noise will be denoted as

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

conditioned on $\{x_{k-l}\}_{l\geqslant 0}$ 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 SME from (2.2), 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 SME, and it is chosen as $\Delta t = \sqrt{\eta(1-\mu)}$. By introducing the auxiliary variable $p_k = \frac{1}{\Delta t}(m_k - m_{k-1})$, we can reformulate (2.2) as a system of (m_k, p_k):

$$p_{k+1} = p_k - \Delta t\sqrt{(1-\mu)/\eta}\,p_k - \Delta t\nabla f(m_k) + \Delta t\big(\nabla f(m_k) - \nabla f_{\gamma_k}(x_{k-\tau_k})\big), \tag{2.3}$$

$$m_{k+1} = m_k + \Delta t\, p_{k+1}. \tag{2.4}$$

To obtain an SME, we first model the random term by a Gaussian random noise, that is, $\Delta t\big(\nabla f(m_k) - \nabla f_{\gamma_k}(x_{k-\tau_k})\big) \sim \sigma_k(\eta(1-\mu))^{1/4}\Delta B_t$, 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 $\mathbb{E}(\Delta B_t \Delta B_t^T) = \Delta t$) and the coefficient is chosen to match the variance. Such modelling is valid because the random variables γ_k and τ_k are independent to each other, and the choices are independent at each iteration, we can approximate the i.i.d. random random term by Gaussian noise in the weak sense. Assuming that Δt is small, we arrive at a Langevin type equation:

$$\begin{aligned}\mathrm{d}P_t &= -\nabla f(M_t)\mathrm{d}t - \sqrt{(1-\mu)/\eta}\,P_t\mathrm{d}t + \sigma(t)(\eta(1-\mu))^{1/4}\mathrm{d}B_t, \\ \mathrm{d}M_t &= P_t\mathrm{d}t,\end{aligned} \tag{2.5}$$

where $\Sigma(t) = \Sigma(\{M_s\}_{0\leqslant s<t}, \{P_s\}_{0\leqslant s<t})$ has the evolution equation (the derivation is deferred to Appendix A)

$$\mathrm{d}\Sigma_t = -\sqrt{\frac{1-\mu}{\eta}}\Sigma_t\,\mathrm{d}t - \mu(\nabla f(M_t)\nabla f(P_t)^T + \nabla f(P_t)\nabla f(M_t)^T)\,\mathrm{d}t - \sqrt{\frac{1-\mu}{\eta}}\nabla f(M_t)\nabla f(M_t)^T\mathrm{d}t$$
$$+\mu\sqrt{\eta(1-\mu)}\nabla f(P_t)\nabla f(P_t)^T\,\mathrm{d}t + \frac{1}{n}\sqrt{\frac{1-\mu}{\eta}}\sum_{i=1}^{n}\nabla f_i\left(M_t+\mu\sqrt{\frac{\eta}{1-\mu}}P_t\right)$$
$$\nabla f_i\left(M_t+\mu\sqrt{\frac{\eta}{1-\mu}}P_t\right)^T\,\mathrm{d}t.$$

When f is a smooth confining potential, that is, f satisfies $\lim_{|x|\to+\infty} f(x) = +\infty$ and $\mathrm{e}^{-\beta f(x)} \in L^1(\mathbb{R}^d)$ for all $\beta \in \mathbb{R}^+$ (an example for f is being a quadratic potential), the process approaches to the minimum of the potential function, and $\sigma(t)$ (as the damping term $-\sqrt{(1-\mu)/\eta}\,\Sigma_t\,\mathrm{d}t$ dominates in the evolution equation) can be approximated by a constant matrix σ up to a first order approximation for large time t. When this constant matrix σ is a multiple of the identity matrix, say $\sigma = \varsigma I$, (P_t, M_t) in the standardized model is an ergodic Markov process with stationary distribution [19]:

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

where Z is a normalization constant. In this case, the resulting friction is $\sqrt{(1-\mu)/\eta}$ and the temperature β^{-1} is $\frac{1}{2}\varsigma^2\eta$. When the constant matrix σ is not a multiple of identity (but still being constant), the stationary distribution takes a similar form in a transformed coordinate system. We remark that though in theory proving time-inhomogeneous process (2.5) has a unique stationary distribution is beyond the scope of this paper, the numerical observations suggest that such a constant approximation of the noise coefficient does not change the process's property fundamentally; in the numerical experiments, we observe that the trajectory of SME-ASGD does not change much when we replace the coefficient of noise by a constant matrix.

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 behaviour of the optimization process. With such a tool, we can better analyse the connection between different stochastic gradient algorithms. Let us illustrate it by showing one example here: Mitliagkas *et al.* [16] argue 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, thus offering a more detailed dynamical comparison.

Stochastic gradient descent with momentum (MSGD) introduced by [20] utilizes the velocity vector from the past updates to accelerate the gradient descent [23]:

$$\begin{aligned} v_{k+1} &= \mu' v_k - \eta'\nabla f_{\gamma_k}(x_k),\\ x_{k+1} &= x_k + v_{k+1}, \end{aligned} \tag{2.6}$$

with a momentum parameter $\mu' \in (0,1)$. (2.6) can be also viewed as a discretization of a second-order stochastic differential equation (SDE). Our derivation here is slightly different from [13], since we use a more natural time scale $\Delta t = \sqrt{\eta'}$ in order to obtain an SDE with bounded coefficients. By taking p to be $v/\sqrt{\eta'}$ (see Appendix A), we end up with the following SME for MSGD (denoted in short as SME-MSGD):

$$\begin{aligned} \mathrm{d}P_t &= -\nabla f(X_t)\,\mathrm{d}t - \frac{1-\mu'}{\sqrt{\eta'}} P_t\,\mathrm{d}t + \sigma(X_t)(\eta')^{\frac{1}{4}}\,\mathrm{d}B_t, \\ \mathrm{d}X_t &= P_t\,\mathrm{d}t, \end{aligned} \tag{2.7}$$

where the friction is $\frac{1-\mu'}{\sqrt{\eta'}}$. Note that (2.7) is time homogeneous with a multiplicative noise, such that the invariant measure usually does not have an explicit expression in general. We further postulate that when the noise is small, the coefficient $\sigma(X_t)$ can be approximated by a constant multiple of the identity matrix. In this case, the temperature $\beta'^{-1} = \frac{\varsigma^2\eta'}{2(1-\mu')}$ dictates the convergence rate to the stationary solution. If we further assume that the noise coefficients σ in SME-ASGD (2.5) and in SME-MSGD (2.7) are the same constant, comparing (2.5) with (2.7) results in the following interesting observation.

PROPOSITION 2.3 If we assume that the noise coefficients σ in SME-ASGD (2.5) and in SME-MSGD (2.7) are the same constant, if $\mu' = \mu$ and $\eta' = \eta(1-\mu)$, then (2.5) and (2.7) have the same stationary distribution.

In Theorems 3 and 5 in Mitliagkas *et al.*'s paper [16], 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 these assumptions, when looking at (2.5) and (2.7) under the same time scale with $\eta' = \eta(1-\mu)$, we can see that $\beta'^{-1} = \frac{\varsigma^2\eta'}{2(1-\mu')} = \frac{\varsigma^2\eta}{2} = \beta^{-1}$. Since the corresponding temperature for the asynchronous method and momentum method are equal, we conclude that the perspective of SME given above explains the observation in [16] that the 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 nonlinear. One can still write the ASGD into a stochastic modified equation. For this, let us define a new auxiliary variable y_k which is proportional to the expected gradient:

$$y_k = -\alpha \mathbb{E}_\tau(\nabla f(x_{k-\tau_k})) = -\alpha \sum\nolimits_{l=0}^{\infty} \nabla f(x_{k-l})(1-\mu)\mu^l, \tag{2.8}$$

where $\alpha > 0$ is to be determined. Again y_k is random and a conditional expectation conditioned on the history of x. Directly following the definition, y_k satisfies the difference equation

$$\frac{y_{k+1}-y_k}{\alpha(1-\mu)} = -\frac{y_k}{\alpha} - \nabla f(x_{k+1}). \tag{2.9}$$

Moreover, we can rewrite the ASGD (2.1) as

$$\frac{x_{k+1}-x_k}{\eta/\alpha} = y_k + \alpha\Big(-\frac{y_k}{\alpha} - \nabla f_{\gamma_k}(x_{k-\tau_k})\Big). \tag{2.10}$$

The reason for us arranging terms in this way is to formulate a Langevin-type equation, but with the noise term moved from the momentum side (Y) to the position side (X). Notice that on the right-hand side of (2.10), $-\frac{y_k}{\alpha} - \nabla f_{\gamma_k}(x_{k-\tau_k})$ can be viewed as a noise with mean 0

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

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

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

In order to view (2.9) and (2.10) as a time discretization of a coupled system with the same time step size, we match $\alpha(1-\mu)$ with η/α by choosing $\alpha = \sqrt{\eta/(1-\mu)}$. Setting the step size $\Delta t = \alpha(1-\mu) = \eta/\alpha = \sqrt{\eta(1-\mu)}$ 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

$$
\begin{aligned}
\mathrm{d}Y_t &= -\nabla f(X_t)\,\mathrm{d}t - \sqrt{\frac{1-\mu}{\eta}}Y_t\,\mathrm{d}t\\
\mathrm{d}X_t &= Y_t\,\mathrm{d}t + \sqrt{\Sigma(t)}\frac{\eta^{3/4}}{(1-\mu)^{1/4}}\,\mathrm{d}B_t.
\end{aligned}
\tag{2.11}
$$

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

$$
\begin{aligned}
\mathrm{d}\Sigma_t = -\sqrt{\frac{1-\mu}{\eta}}\Sigma_t\,\mathrm{d}t &+ \sqrt{\frac{1-\mu}{\eta}}\left(\frac{1}{n}\sum_{i=1}^{n}\nabla f_i(X_t)\nabla f_i(X_t)^T + \frac{1-\mu}{\mu}\nabla f(X_t)\nabla f(X_t)^T\right)\mathrm{d}t\\
&+ \frac{1-\mu}{\eta\mu}\left(\sqrt{\frac{1-\mu}{\eta}}Y_tY_t^T + \nabla f(X_t)Y_t^T + Y_t\nabla f(X_t)^T\right)\mathrm{d}t.
\end{aligned}
\tag{2.12}
$$

The derivation of (2.12) is shown in Appendix A. The combined system (2.11)–(2.12) will be referred as SME-ASGD (the SMEs for asynchronous SGD) for the general nonlinear-gradient case. We should point it out that, unlike the linear-gradient case (2.5), (2.11) has no known explicit formula for invariant measure even when $\Sigma(t)$ converging to a constant matrix. Nevertheless, the ergodicity of (2.11) and (2.12) will be an interesting future direction to explore.

We would like to point out that when the gradient ∇f is linear (2.9) and (2.10) can be easily transformed back to (2.3) and (2.4). As a consequence, (2.5) and (2.11) are equivalent. To see this,

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

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 (2.9) as

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

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 (2.4). To show (2.3), 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 (2.3).

3. Approximation error of the SME

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

THEOREM 3.1 Assume that Assumption 2.1 holds and that the variance from the asynchronous gradients is uniformly bounded (i.e., there exists $c > 0$ such that $||\sigma(t)|| \leqslant c$). Suppose also that all the iterates updated from the ASGD stay bounded and that the solutions for SME-ASGD and ASGD before time 0 agree (i.e., $X_{l\Delta t} = x_l, l \leqslant 0$, with $\Delta t = \sqrt{\eta(1-\mu)}$ as given previously). Then the SME-ASGD approximates the ASGD in the sense that there exists constant $K_T > 0$ depending only on T such that

$$\sup_{n\Delta t \leqslant T} \mathbb{E}\{|X_{n\Delta t} - x_n|\} \leqslant K_T \frac{\Delta t}{1-\mu} \tag{3.1}$$

for Δt sufficiently small. Here $X_{n\Delta t} \equiv X(n\Delta t)$ is the solution of (2.12) at time $n\Delta t$ and x_n is from ASGD (2.1).

The assumption $\sigma = \sqrt{\Sigma} = O(1)$ can be justified from (2.12) 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. Since we investigate the error approximation in finite time T and finite step size Δt, there are only a finite number of iterations. In each iteration, the iterate updated from the ASGD stays bounded by a sufficient large constant with high probability. Therefore, the assumption that all iterates stay bounded by a sufficient large constant holds with high probability.

The proof of the Theorem (3.1) follows from viewing the ASGD as a discretization of SME-ASGD and using the analysis of strong convergence for numerical schemes for SDEs.

Proof of the Theorem (3.1) We look at the one step approximation in the first step, and the global approximation can be done by induction. Using the variation of constant formula, we know that the solution of

$$\mathrm{d}Y_t = -\nabla f(X_t)\,\mathrm{d}t - \sqrt{\frac{1-\mu}{\eta}}Y_t\,\mathrm{d}t$$

is given by

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

where $Y_0 = -\sqrt{\frac{\eta}{1-\mu}}\sum_{l=0}^{\infty}\nabla f(x_{-l})(1-\mu)\mu^l$ as defined in (2.8). Plugging Y_t into the integral form of $X_{\Delta t}$ gives rise to

$$X_{\Delta t} = x_0 + \int_0^{\Delta t}\left(\mathrm{e}^{-\sqrt{\frac{1-\mu}{\eta}}s}Y_0 - \int_0^s \mathrm{e}^{-\sqrt{\frac{1-\mu}{\eta}}(s-u)}\nabla f(X_u)\,\mathrm{d}u\right)\mathrm{d}s + \frac{\eta^{3/4}}{(1-\mu)^{1/4}}\int_0^{\Delta t}\sigma(s)\,\mathrm{d}B_s. \quad (3.2)$$

Denote $v_k := x_{k-\tau_k}$ for notation convenience. By 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:

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

where *I*, *II* and *III* are the first three terms appearing in the right-hand side of the first inequality. In the above derivation, we have applied the Ito isometry to the fourth term and used

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

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

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

Let us now treat the first three terms

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

since the first two terms cancel. Because ∇f is Lipschitz and $\mathrm{e}^{-\sqrt{\frac{1-\mu}{\eta}}(s-u)}\leqslant 1$ for $u\leqslant s$, the second term can be estimated with

$$II\leqslant L\Delta t\int_0^{\Delta t}\mathbb{E}\left\{|X_u-x_1|\right\}\mathrm{d}u.$$

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

$$III\leqslant|\nabla f(x_1)|\int_0^{\Delta t}s\,\mathrm{d}s=|\nabla f(x_1)|\Delta t^2/2=O(\Delta t^2).$$

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

$$\mathbb{E}\left\{|X_{\Delta t}-x_1|\right\}\leqslant C\frac{\Delta t^2}{1-\mu}+L\Delta t\int_0^{\Delta t}\mathbb{E}\left\{|X_u-x_1|\right\}\mathrm{d}u.$$

An application of Gronwall's inequality shows that

$$\mathbb{E}\left\{|X_{\Delta t}-x_1|\right\}\leqslant C\frac{\Delta t^2}{1-\mu}\,\mathrm{e}^{L\Delta t^2}=C\frac{\Delta t^2}{1-\mu}+O(\Delta t^4)\leqslant C\frac{\Delta t^2}{1-\mu}.$$

This concludes the estimate for the first step at time 0.

The induction step is similar. We have

$$X_{(k+1)\Delta t} = X_{k\Delta t} + \int_{k\Delta t}^{(k+1)\Delta t} \left(\mathrm{e}^{-\sqrt{\frac{1-\mu}{\eta}}(s-k\Delta t)} Y_{k\Delta t} - \int_{k\Delta t}^{s} \mathrm{e}^{-\sqrt{\frac{1-\mu}{\eta}}(s-u)} \nabla f(X_u)\,\mathrm{d}u \right) \mathrm{d}s \\ + \frac{\eta^{3/4}}{(1-\mu)^{1/4}} \int_{k\Delta t}^{(k+1)\Delta t} \sigma(s)\,\mathrm{d}B_s.$$

For the discrete update step $x_{k+1} = x_k - \eta \nabla f_{\gamma_k}(v_k)$, we split $\eta \nabla f_{\gamma_k}(v_k)$ as before. With the assumption $\mathbb{E}\{|X_{k\Delta t} - x_k|\} \leqslant Ck\frac{\Delta t^2}{1-\mu}$, we have the following estimate:

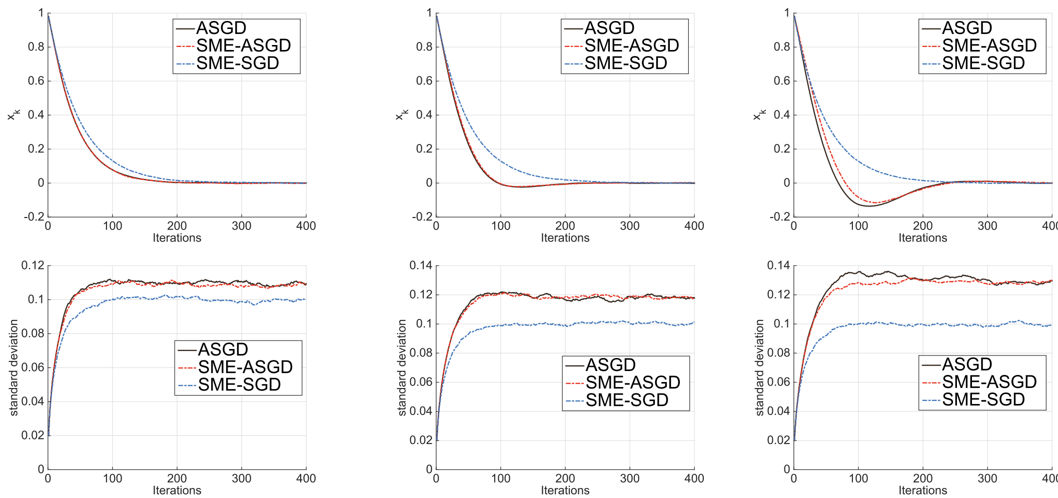
$$\begin{aligned}
\mathbb{E}\{|X_{(k+1)\Delta t} - x_{k+1}|\} \leqslant \mathbb{E}\{|X_{k\Delta t} - x_k|\} &+ \int_{k\Delta t}^{(k+1)\Delta t} \mathrm{e}^{-\sqrt{\frac{1-\mu}{\eta}}(s-k\Delta t)} \left|Y_{k\Delta t} - y_k\right| \mathrm{d}s \\
&+ \left| \int_{k\Delta t}^{(k+1)\Delta t} \mathrm{e}^{-\sqrt{\frac{1-\mu}{\eta}}(s-k\Delta t)} y_k\,\mathrm{d}s + \eta \sum_{l=0}^{\infty} \nabla f(x_{k-l})(1-\mu)\mu^l \right| \\
&+ \mathbb{E}\left\{ \int_{k\Delta t}^{(k+1)\Delta t} \left(\int_{k\Delta t}^{s} \mathrm{e}^{-\sqrt{\frac{1-\mu}{\eta}}(s-u)} \left|\nabla f(X_u) - \nabla f(x_{k+1})\right| \mathrm{d}u \right) \mathrm{d}s \right\} \\
&+ |\nabla f(x_{k+1})| \int_{k\Delta t}^{(k+1)\Delta t} \int_{k\Delta t}^{s} \mathrm{e}^{-\sqrt{\frac{1-\mu}{\eta}}(s-u)}\,\mathrm{d}u\,\mathrm{d}s \\
&+ \frac{\eta^{3/4}}{(1-\mu)^{1/4}} \left(\mathbb{E}\left\{ \left(\int_{k\Delta t}^{(k+1)\Delta t} \sigma(s)\,\mathrm{d}B_s \right)^2 \right\} \right)^{1/2} \\
&+ \mathbb{E}\left\{ \left| \eta \nabla f_{\gamma_k}(v_k) - \eta \sum_{l=0}^{\infty} \nabla f(x_{k-l})(1-\mu)\mu^l \right| \right\}.
\end{aligned}$$

Here the only difference compared to the first step is the term $Y_{k\Delta t}$, which is not given, but generated from SME. Note that

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

From (2.9), 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

$$\int_{k\Delta t}^{(k+1)\Delta t} \mathrm{e}^{-\sqrt{\frac{1-\mu}{\eta}}(s-k\Delta t)} \left|Y_{k\Delta t} - y_k\right| \mathrm{d}s = O(\Delta t^2).$$



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

FIG. 1. Apply the SME-ASGD to minimize the quadratic function $f(x) = x^2$ in different μs, with two components $f_1(x) = (x-1)^2 - 1$, 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 (3.3), especially when μ becomes large. However, one can also observe that when μ increases the error of the SME-ASGD approximation increases as well.

The third term has the estimate

$$\left|\int_{k\Delta t}^{(k+1)\Delta t} \mathrm{e}^{-\sqrt{\frac{1-\mu}{\eta}}(s-k\Delta t)} y_k \,\mathrm{d}s + \eta \sum_{l=0}^{\infty} \nabla f(x_{k-l})(1-\mu)\mu^l\right|$$

$$= \left|-\sqrt{\frac{\eta}{1-\mu}}\left(\mathrm{e}^{-\sqrt{\frac{\eta}{1-\mu}}\Delta t} - 1\right) y_k + \eta \sum_{l=0}^{\infty} \nabla f(x_{k-l})(1-\mu)\mu^l\right|$$

$$= \left|-\sqrt{\frac{\eta}{1-\mu}} \sum_{l=0}^{\infty} \nabla f(x_{k-l})(1-\mu)\mu^l \Delta t + \eta \sum_{l=0}^{\infty} \nabla f(x_{k-l})(1-\mu)\mu^l + O(\Delta t^2)\right| = O(\Delta t^2)$$

as before. All other terms have the same estimates as in the base case. Applying the Gronwall's inequality again and letting Δt be sufficiently small gives the estimate

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

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

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

□

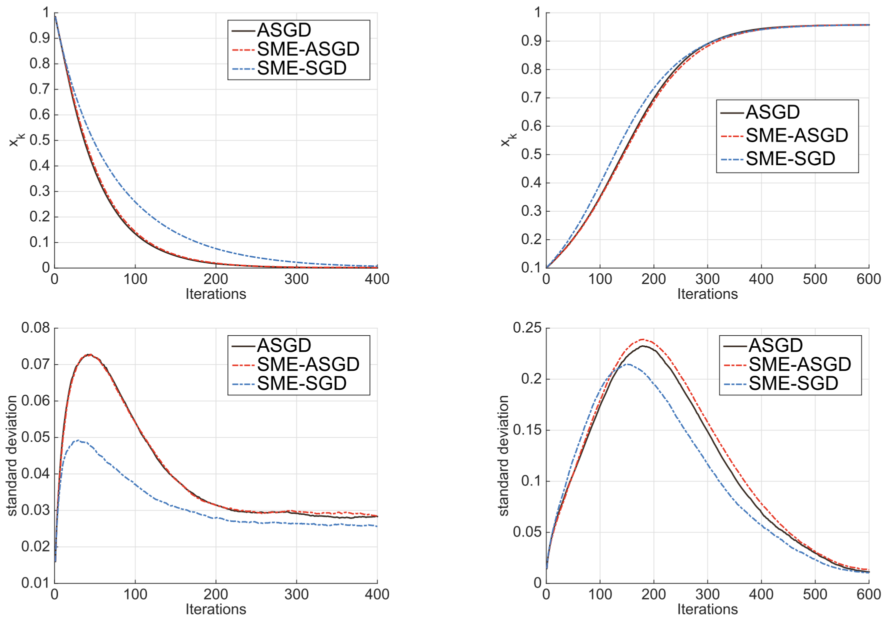


FIG. 2. (Left) Apply the SME-ASGD to minimize the convex function $f(x) = x^4 + 6x^2$ with two components $f_1(x) = (x-1)^4 - 1$, and $f_2(x) = (x+1)^4 - 1$. Notice that the gradients are Lipschitz locally. Here we choose $x_0 = 1$, and a smaller step size $\eta = 1e-3$. (Right) Apply the SME-ASGD to minimize the double well potential $f(x) = 1 - \mathrm{e}^{-(x-1)^2} - \mathrm{e}^{-(x+1)^2}$. Here $f_1 = 1 - 2\,\mathrm{e}^{-(x-1)^2}$, $f_2 = 1 - 2\,\mathrm{e}^{-(x+1)^2}$ 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 $\mu = 0.95$.

One interesting observation is that, contrary to the standard Euler–Maruyama method for SDEs having strong order of convergence $1/2$ [12], 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 [13] has the same feature: the coefficient of the noise term there is of order $\sqrt{\eta}$. When $\eta \approx 1-\mu$, the two orders are the same.

Here we provide some numerical evidences for Theorem 3.1 with various loss functions f. The results are shown in Figs 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 (2.11) and the second-order weak convergent SME-SGD proposed in Li *et al.*'s paper [13]

$$\mathrm{d}X_t = -\nabla\left(f(X_t) + \frac{\eta}{4}|\nabla f(X_t)|^2\right)\mathrm{d}t + (\eta\Sigma(X_t))^{1/2}\,\mathrm{d}B_t. \tag{3.3}$$

When μ is close to 0 (i.e., the expected delay is short), SME-SGD (3.3) serves as a good approximation to ASGD as expected. However, when μ is large, Figs 1 and 2 demonstrate that it is no longer the case: as μ gets closer to 1, the trajectories obtained from SME-SGD are way off, whereas our proposed SME-ASGD model demonstrate accurate path approximations for both the first and the second moments.

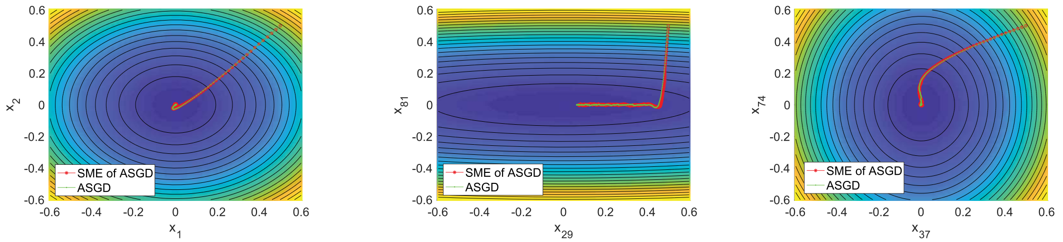


FIG. 3. Apply the SME-ASGD to minimize the quadratic function $f(x) = \sum_{i=1}^{100} c_i x_i^2/2, x \in \mathbb{R}^{100}$ with $\mu = 0.90$ and two components $f_1(x) = \sum_{i=1}^{100} c_i x_i^2/2 - x$, and $f_2(x) = \sum_{i=1}^{100} c_i x_i^2/2 + x$. The initial condition $x_0 = (0.5, 0.5, \cdots, 0.5) \in \mathbb{R}^{100}$ and the step size is $\eta = 1e-2$. The plots are done after 1000 iterations. The corresponding coefficients in the plots are $c_1 = 4.2593, c_2 = 4.9013, c_{29} = 0.1980, c_{81} = 4.3968, c_{37} = 3.9978, c_{74} = 1.9527$.

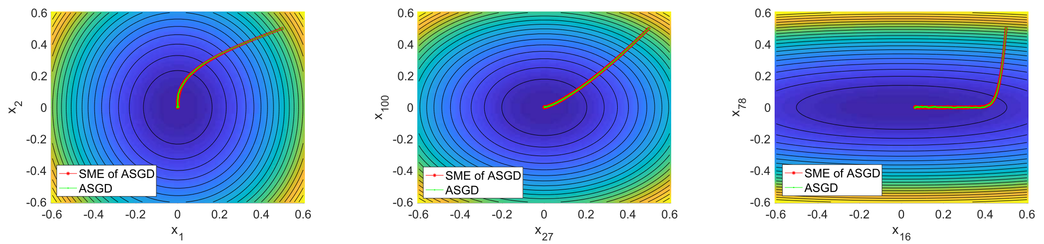


FIG. 4. Apply the SME-ASGD to minimize the convex function $f(x) = \sum_{i=1}^{100} c_i(x_i^4 + 6x_i^2)/2, x \in \mathbb{R}^{100}$ with $\mu = 0.90$ and two components $f_1(x) = \sum_{i=1}^{100} c_i(x_i-1)^4/2 - 1$ and $f_2(x) = \sum_{i=1}^{100} c_i(x_i+1)^4/2 + 1$. The initial condition $x_0 = (0.5, 0.5, \cdots, 0.5) \in \mathbb{R}^{100}$ and the step size is $\eta = 1e-3$. The plots are done after 1000 iterations. The corresponding coefficients in the plots are $c_1 = 3.9212, c_2 = 1.9370, c_{27} = 1.2093, c_{100} = 1.5661, c_{16} = 0.3353, c_{78} = 4.5502$.

A few remarks regarding the numerical results are in order here. (i) In Fig. 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 $\Sigma(X_t)$ by the covariance matrix formula 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 $f(x) = x^4 + 6x^2$ (with gradient $\nabla f(x) = 4x^3 + 12x$) in Fig. 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) For the non-convex example (the double-well function in Fig. 2), the SME-ASGD model gives a better prediction about which minimizer that a trajectory with given initial data will fall into: the percentage of path samples that converge to a local minimum in SME-ASGD is very close to that of the ASGD case. (iv) For all cases, SME-SGD underestimates the variance because the variance from the delayed reads is not taken into account by SME-SGD. (v) In higher dimensions, unlike the Monte Carlo sampling driven by Langevin dynamics that has the curse of dimensionality issue, our numerical simulations for both linear and nonlinear gradients have good approximation regardless of the dimensionality as the Figs 3 and 4 show. Here, we assign the coefficients c_i uniformly randomly in $[0, 5]$. We make plots by arbitrarily choosing any two dimensions as projected subspace. Although after 1000 time steps, some projected subspace have convergence and some (with significant coefficient differences) do not yet, we can see that the trajectories from the algorithm and modified equation are close.

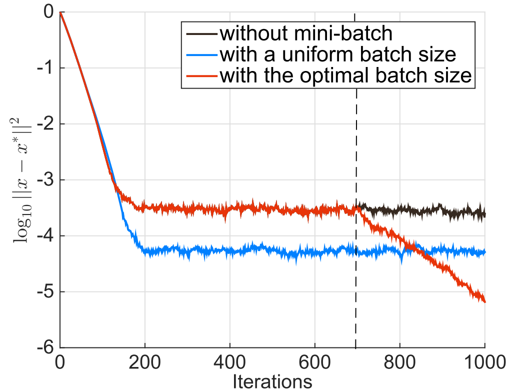


FIG. 5. A comparison of performance in terms of l^2 error. We apply mini-batching over $n = 100$ components $f_i(x) = \frac{1}{2}(x - c_i)^2, c_i = -1/2 + i/(2n)$. Here we choose the step size $\eta = 0.02$ and the initial data $x_0 = 1$. The batch size for the uniform mini-batching case is 5. For the 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.

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 $u_k \geqslant 0$, one can write the iteration as

$$x_{k+1} = x_k - \eta \frac{1}{1 + u_k} \sum_{j=1}^{1+u_k} \nabla f_{\gamma_j}(x_{k-\tau_k}). \tag{4.1}$$

We argue that it is reasonable to 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 can derive a corresponding SME

$$\begin{aligned} \mathrm{d}Y_t &= -\nabla f(X_t)\,\mathrm{d}t - \sqrt{\frac{1-\mu}{\eta}}\,Y_t\,\mathrm{d}t \\ \mathrm{d}X_t &= Y_t\,\mathrm{d}t + \frac{\sigma(t)\eta^{3/4}}{(1 + u(t))^{1/2}(1-\mu)^{1/4}}\,\mathrm{d}B_t. \end{aligned} \tag{4.2}$$

The derivation here is not much different from the one of SME-ASGD (2.11), except for identifying the right coefficient in front of the noise term $\mathrm{d}B_t$. The correct coefficient (denoted by c in the discussion below) is constrained by the following constraints on the variance:

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

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

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

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

as shown in (4.2).

We would like to explore the dynamics of SME to find the dominating eigenvalue for later use. 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, one obtains the following evaluation system for the second moments:

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix}\mathbb{E}(X_t^2)\\ \mathbb{E}(Y_t^2)\\ \mathbb{E}(X_tY_t)\end{bmatrix}=-\begin{bmatrix}0 & 0 & -2\\ 0 & 2\sqrt{(1-\mu)/\eta} & 4\\ 2 & -1 & \sqrt{(1-\mu)/\eta}\end{bmatrix}\begin{bmatrix}\mathbb{E}(X_t^2)\\ \mathbb{E}(Y_t^2)\\ \mathbb{E}(X_tY_t)\end{bmatrix}+\begin{bmatrix}\frac{\Sigma(t)\eta^{3/2}}{(1+u(t))(1-\mu)^{1/2}}\\ 0\\ 0\end{bmatrix}. \qquad (4.3)$$

A similar derivation is shown in Appendix B, and we just replace all Σ by $\Sigma/(1+u(t))$ in the mini-batching case. Here we make a simplifying but practical assumption that $u(t)$ varies slowly. Now by freezing $u(t)$ to a constant u, (4.3) is a linear system with constant coefficients, its asymptotic behaviour is determined by the eigenvalue of the coefficient matrix. An easy calculation shows that the eigenvalue with largest real part is given by $\lambda=-\sqrt{(1-\mu)/\eta}+\sqrt{(1-\mu-8\eta)/\eta}$ with a negative real part and therefore the second moment of X_t decays exponentially. Moreover, (4.3) provides us with the stationary solution for X^2:

$$z_\infty:=\mathbb{E}(X_\infty^2)=\frac{\Sigma\eta}{2(1+u(t))}\left(\frac{\eta}{1-\mu}+\frac{1}{2}\right). \qquad (4.4)$$

For a slowly varying $u(t)$, $z_\infty=z_\infty(u(t))$ is a function of $u(t)$. Based on this simplication, 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:

$$\begin{aligned} &\min_{u\in\mathcal{A}}\left\{z(T)+\frac{\gamma}{\eta}\int_0^T u(s)\,\mathrm{d}s\right\} \quad \text{subject to} \\ &\frac{\mathrm{d}}{\mathrm{d}t}z(t)=\mathrm{Re}(\lambda)(z(t)-z_\infty(u(t))) \qquad \text{with } z(0)=x_0^2, \end{aligned} \tag{4.5}$$

where $z(t)$ models $\mathbb{E}(X_t^2)$—the quantity to minimize, $\mathscr{A}=\{u(t)\geqslant 0\}$ 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. Below we show how to solve the optimal control problem (4.5). The value function can be defined as

$$V(z,t)=\min_{u\in\mathscr{A}}\left\{z(T)+\frac{\gamma}{\eta}\int_t^T u(s)\,\mathrm{d}s \;\middle|\; \frac{\mathrm{d}}{\mathrm{d}t}z(t)=F(u(t),z(t)),z(t)=z\right\}, \tag{4.6}$$

where $F(u(t),z(t))=\mathrm{Re}(\lambda)(z(t)-z_\infty(u(t)))=\mathrm{Re}(\lambda)\big(z(t)-\frac{\Sigma\eta}{2(1+u(t))}\big(\frac{\eta}{1-\mu}+\frac{1}{2}\big)\big)$. The corresponding Hamilton–Jacobi–Bellman equation is

$$\begin{aligned} &V_t+\min_{u\in\mathscr{A}}\left\{F(u,z)V_z+\frac{\gamma}{\eta}u\right\}=0 \\ &\text{with } V(0,t)=0, V(z,T)=z. \end{aligned} \tag{4.7}$$

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

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

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

$$u^*(V_z)=\begin{cases}\sqrt{\frac{-V_z\mathrm{Re}(\lambda)\Sigma\eta^2}{2\gamma}\big(\frac{\eta}{1-\mu}+\frac{1}{2}\big)}-1 & \text{if } \frac{-V_z\mathrm{Re}(\lambda)\Sigma\eta^2}{2\gamma}\big(\frac{\eta}{1-\mu}+\frac{1}{2}\big)>1,\\ 0 & \text{otherwise.}\end{cases} \tag{4.8}$$

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

$$\min_{u\in\mathscr{A}}\left\{F(u,z)V_z+\frac{\gamma}{\eta}u\right\}=\begin{cases}\mathrm{Re}(\lambda)zV_z-\frac{\gamma}{\eta} & \text{if } \frac{-V_z\mathrm{Re}(\lambda)\Sigma\eta^2}{2\gamma}\big(\frac{\eta}{1-\mu}+\frac{1}{2}\big)>1,\\ \mathrm{Re}(\lambda)\big(z-\frac{\Sigma\eta}{2}\big(\frac{\eta}{1-\mu}+\frac{1}{2}\big)\big)V_z & \text{otherwise.}\end{cases} \tag{4.9}$$

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

$$V(z,t) = \begin{cases} \frac{\Sigma\eta}{2}\left(\frac{\eta}{1-\mu}+\frac{1}{2}\right) + \left(z - \frac{\Sigma\eta}{2}\left(\frac{\eta}{1-\mu}+\frac{1}{2}\right)\right)\mathrm{e}^{\mathrm{Re}(\lambda)(T-t)} & \text{if } \gamma > \gamma^* \\ \left(z - \frac{\Sigma\eta}{2}\left(\frac{\eta}{1-\mu}+\frac{1}{2}\right)\right)\mathrm{e}^{\mathrm{Re}(\lambda)(T-t)} - \frac{\gamma}{\eta}\left(t^* + \frac{1}{\mathrm{Re}(\lambda)}\right) & \text{if } \gamma \leqslant \gamma^*, 0 \leqslant t \leqslant T - t^* \\ z\,\mathrm{e}^{\mathrm{Re}(\lambda)(T-t)} - \frac{\gamma}{\eta}(T-t) & \text{if } \gamma \leqslant \gamma^*, T - t^* < t \leqslant T, \end{cases} \tag{4.10}$$

where $t^* = \frac{1}{\mathrm{Re}(\lambda)}\log(\frac{\gamma}{\gamma^*})$. For all cases, $V_z = \mathrm{e}^{\mathrm{Re}(\lambda)(T-t)}$. With this inserted back into (4.8), we conclude that

$$u^*(t) = \begin{cases} 0 & \text{if } \gamma > \gamma^* \text{or } 0 \leqslant t \leqslant T - t^* \\ \sqrt{\frac{\gamma^*}{\gamma}}\,\mathrm{e}^{\mathrm{Re}(\lambda)(T-t)/2} - 1 & \text{if } \gamma \leqslant \gamma^*, T - t^* < t \leqslant T. \end{cases} \tag{4.11}$$

In particular, (4.11) tells that we should use a small mini-batch size (even size 1) during the early time (for $k \leqslant k^* = (T - t^*)/\eta$), since during this period the gradient flow dominates the dynamics. After the transition time k^* at which the noise starts to dominate, one shall apply mini-batch with size exponentially increasing in k to reduce the variance. Figure 5 demonstrates that our proposed mini-batching strategy outperforms the ASGD with a constant batch size (for example, applied in [**4**,**7**]). 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., [**8**,**10**].

5. Conclusion

In this paper, we have developed SMEs to model the ASGD algorithms in the continuous-time limit. For quadratic loss functions, the resulting SME can be put into a Langevin equation with a solution known to converge to the unique invariant measure with a 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 general case, though the resulting SME does not have an explicitly known invariant measure, it still provides rather 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.

Funding

Gene Golub Research Fellowship (to J.A.); National Science Foundation (DMS-1454939 to J.L.); U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, Scientific Discovery through Advanced Computing program and the National Science Foundation (DMS-1818449 to L.Y.).

References

1. Agarwal, A. & Duchi, J. C. (2011) Distributed delayed stochastic optimization. *Advances in Neural Information Processing Systems*, USA: Curran Associates, Inc., pp. 873–881.
2. Bernstein, S. (1929) Sur les fonctions absolument monotones. *Acta Math.*, **52**, 1–66.
3. Bottou, L., Curtis, F. E. & Nocedal, J. (2018) Optimization methods for large-scale machine learning. *SIAM Rev.*, **60**, 223–311.

4. Dekel, O., Gilad-Bachrach, R., Shamir, O. & Xiao, L. (2012) Optimal distributed online prediction using mini-batches. *J. Mach. Learn. Res.*, **13**, 165–202.

5. Duchi, J. C., Hazan, E. & Singer, Y. (2011) Adaptive subgradient methods for online learning and stochastic optimization. *J. Mach. Learn. Res.*, **12**, 2121–2159.

6. Duchi, J. C., Jordan, M. I. & McMahan, B. (2013) Estimation, optimization, and parallelism when data is sparse. *Advances in Neural Information Processing Systems*, USA: Curran Associates, Inc., pp. 2832–2840.

7. Gimpel, K., Das, D. & Smith, N. A. (2010) Distributed asynchronous online learning for natural language processing. *Proceedings of the Fourteenth Conference on Computational Natural Language Learning*, pp. 213–222. Association for Computational Linguistics.

8. Goyal, P., Dollár, P., Girshick, R., Noordhuis, P., Wesolowski, L., Kyrola, A., Tulloch, A., Jia, Y. & He, K. (2017) Accurate, large minibatch SGD: training ImageNet in 1 hour. arXiv preprint, arXiv:1706.02677.

9. Hardt, M., Recht, B. & Singer, Y. (2016) Train faster, generalize better: stability of stochastic gradient descent. *International Conference on Machine Learning*, vol. 48, New York, NY, USA: JMLR: W&CP pp. 1225–1234.

10. Keskar, N. S., Mudigere, D., Nocedal, J., Smelyanskiy, M. & Tang, P. T. P. (2017) On large-batch training for deep learning: generalization gap and sharp minima. *International Conference on Learning Representations*. https://openreview.net/forum?id=H1oyRlYgg.

11. Kingma, D. P. & Ba, J. (2015) Adam: a method for stochastic optimization. *International Conference on Learning Representations*, Amsterdam Machine Learning lab (IVI, FNWI).

12. Kloeden, P. E. & Platen, E. (1992) Stochastic differential equations. *Numerical Solution of Stochastic Differential Equations*. Verlag Berlin Heidelberg: Springer, pp. 103–160.

13. Li, Q., Tai, C. & Weinan, W. E. (2017) Stochastic modified equations and adaptive stochastic gradient algorithms. *International Conference on Machine Learning*, International Convention Centre, Sydney, Australia: Proceedings of Machine Learning Research, pp. 2101–2110.

14. Liu, J. & Wright, S. J. (2015) Asynchronous stochastic coordinate descent: parallelism and convergence properties. *SIAM J. Optim.*, **25**, 351–376.

15. Liu, J., Wright, S. J., Ré, C., Bittorf, V. & Sridhar, S. (2015) An asynchronous parallel stochastic coordinate descent algorithm. *J. Mach. Learn. Res.*, **16**, 285–322.

16. Mitliagkas, I., Zhang, C., Hadjis, S. & Ré, C. (2016) Asynchrony begets momentum, with an application to deep learning. *2016 54th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. IEEE, Allerton, USA, pp. 997–1004.

17. Needell, D., Ward, R. & Srebro, N. (2016) Stochastic gradient descent, weighted sampling, and the randomized kaczmarz algorithm. *Mathematical Programming*, **155**, 549–573, Springer.

18. Nesterov, Y. (2012) Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM J. Optim.*, **22**, 341–362.

19. Pavliotis, G. A. (2014) *Stochastic Processes and Applications: Diffusion Processes, the Fokker–Planck and Langevin Equations*, vol. **60**, New York, NY: Springer.

20. Polyak, B. T. (1964) Some methods of speeding up the convergence of iteration methods. *USSR Comput. Math. Math. Phys.*, **4**, 1–17.

21. Recht, B., Ré, C., Wright, S. & Niu, F. (2011) Hogwild: a lock-free approach to parallelizing stochastic gradient descent. *Advances in Neural Information Processing Systems*, pp. 693–701.

22. Richtárik, P. & Takáč, M. (2014) Iteration complexity of randomized block-coordinate descent methods for minimizing a composite function. *Math. Program.*, **144**, 1–38.

23. Sutskever, I., Martens, J., Dahl, G. & Hinton, G. (2013) *Proceedings of the 30th International Conference on Machine Learning, PMLR*, **28**: 1139–1147, 2013.

24. Tieleman, T. & Hinton, G. (2012) Lecture 6.5-rmsprop: divide the gradient by a running average of its recent magnitude. *COURSERA: Neural Networks for Machine Learning*, **4**, 26–31.

25. Younes, H. L. (2005) Verification and planning for stochastic processes with asynchronous events. *Ph.D Thesis*, Academy of Engineering Sciences.

Appendix A: miscellaneous computations in SMEs

In this section, we provide the missing computations in Section 2.

A.1 *Evolution equation of Σ for nonlinear gradients*

First, we have

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

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

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

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

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

Substituting it in (A.1), we obtain

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

Using this and $\Delta t = \alpha(1-\mu), \alpha = \sqrt{\frac{\eta}{1-\mu}}$, we obtain the evolution equation (2.12).

A.2 *Evolution equation of Σ for linear gradients*

Similar to Section 5.1, we have

$$\begin{aligned}\Sigma_k &= \mathbb{E}\big((\nabla f(m_k) - \nabla f_{\gamma_k}(x_{k-\tau_k}))(\nabla f(m_k) - \nabla f_{\gamma_k}(x_{k-\tau_k}))^T\big)\\ &= \mu(\Sigma_{k-1} + \nabla f(m_{k-1})\nabla f(m_{k-1})^T) + (1-\mu)\mathbb{E}(\nabla f_{\gamma_k}(x_k)\nabla f_{\gamma_k}(x_k)^T) - \nabla f(m_k)\nabla f(m_k)^T. \quad \text{(A.2)}\end{aligned}$$

We then subtract both sides by Σ_{k-1} and divide by $\Delta t = \sqrt{\eta(1-\mu)}$. Moreover, we use the relation

$$x_k = \frac{m_k - \mu m_{k-1}}{1-\mu} = \frac{m_k - \mu(m_k - p_k\Delta t)}{1-\mu} = m_k + \mu\sqrt{\frac{\eta}{1-\mu}}p_k$$

to replace x_k in (A.2), and replace m_{k-1} by $m_k - p_k\Delta t$. Then, since the gradient of f is linear, rearrange the terms we get

$$\begin{aligned}\frac{\Sigma_k - \Sigma_{k-1}}{\sqrt{\eta(1-\mu)}} = &-\sqrt{\frac{1-\mu}{\eta}}\Sigma_{k-1} - \sqrt{\frac{1-\mu}{\eta}}\nabla f(m_k)\nabla f(m_k)^T + \mu\sqrt{\eta(1-\mu)}\nabla f(p_k)\nabla f(p_k)^T\\ &-\mu(\nabla f(m_k)\nabla f(p_k)^T + \nabla f(p_k)\nabla f(m_k)^T)\\ &+\frac{1}{n}\sqrt{\frac{1-\mu}{\eta}}\sum_{i=1}^n \nabla f_i\left(m_k + \mu\sqrt{\frac{\eta}{1-\mu}}p_k\right)\nabla f_i\left(m_k + \mu\sqrt{\frac{\eta}{1-\mu}}p_k\right)^T.\end{aligned}$$

A.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 (2.6) as

$$\begin{aligned}\frac{v_{k+1}}{\sqrt{\eta'}} &= \frac{v_k}{\sqrt{\eta'}} + \sqrt{\eta'}\left(-\frac{1-\mu'}{\eta'}v_k - \nabla f(x_k)\right) + \sqrt{\eta'}(\nabla f(x_k) - \nabla f_{\gamma_k}(x_k))\\ x_{k+1} &= x_k + \frac{v_{k+1}}{\sqrt{\eta'}}\sqrt{\eta'}. \quad \text{(A.3)}\end{aligned}$$

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:

$$\begin{aligned}\mathrm{d}P_t &= -\nabla f(X_t)\,\mathrm{d}t - \frac{1-\mu'}{\sqrt{\eta'}}P_t\,\mathrm{d}t + \sigma(X_t)(\eta')^{\frac{1}{4}}\,\mathrm{d}B_t\\ \mathrm{d}X_t &= P_t\,\mathrm{d}t.\end{aligned}$$

Appendix B: dynamics of SME-ASGD (2.11)

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 (2.11). Taking expectation, we obtain

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

One observes that the eigenvalues of $A(\mu,\eta)$ are $\lambda_{1,2}(A) = \frac{1}{2}\left(-\sqrt{\frac{1-\mu}{\eta}} \pm \sqrt{\frac{1-\mu}{\eta} - 4a}\right)$, 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:

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

In order to study the behaviour of the second moments, we can rewrite (B.1) as

$$\mathrm{d}\begin{bmatrix}\mathbb{E}(X_t^2)\\ \mathbb{E}(Y_t^2)\\ \mathbb{E}(X_tY_t)\end{bmatrix} = \begin{bmatrix}0 & 0 & 2\\ 0 & -2\sqrt{\frac{1-\mu}{\eta}} & -2a\\ -a & 1 & -\sqrt{\frac{1-\mu}{\eta}}\end{bmatrix}\begin{bmatrix}\mathbb{E}(X_t^2)\\ \mathbb{E}(Y_t^2)\\ \mathbb{E}(X_tY_t)\end{bmatrix}\mathrm{d}t + \begin{bmatrix}\Sigma(t)\frac{\eta^{3/2}}{(1-\mu)^{1/2}}\\ 0\\ 0\end{bmatrix}\mathrm{d}t. \tag{B2}$$

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

$$\mu_{\text{opt}} = \max\{1 - 4a\eta, 0\}. \tag{B.3}$$

We note that (B.3) 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.