Supplementary materials for Stochastic modified equations and the dynamics of stochastic gradient algorithms

A Modified equations in the numerical analysis of PDEs

The method of modified equations is widely applied in finite difference methods in numerical solution of PDEs Hirt (1968); Noh & Protter (1960); Daly (1963); Warming & Hyett (1974). In this section, we briefly demonstrate this classical method. Consider the one dimensional transport equation

$$\frac{\partial u}{\partial t} = c\frac{\partial u}{\partial x} \tag{1}$$

where $u : [0,T] \times [0,L] \to \mathbb{R}$ represents a density of some material in $[0,L]$ and $c > 0$ is the transport velocity. It is well-known that the simple forward-time-central-space differencing leads to instability for all discretization step-sizes (LeVeque, 2002). Instead, more sophisticated differencing schemes must be used.

We set time and space discretization steps to Δt and Δx and denote $u(n\Delta t, j\Delta x) = U_{n,j}$ for $1 \leq n \leq N$ and $1 \leq j \leq J$. The simplest scheme that can exhibit stability is the *upwind scheme* (Courant et al., 1952), where we approximate (1) by the difference equation

$$U_{n+1,j} = U_{n,j} + \Delta t \left(c^{+}\frac{U_{n,j+1} - U_{n,j}}{\Delta x} + c^{-}\frac{U_{n,j} - U_{n,j-1}}{\Delta x} \right), \tag{2}$$

where $c^{+} = \max(c,0)$ + $c^{-} = \min(c,0)$. The idea is to now approximate this difference scheme by another continuous PDE, that is not equal to the original equation (1) for non-zero $\Delta x, \Delta t$. This can be done by Taylor expanding each term in (2) around $u(t,x) = U_{n,j}$. Simplifying and truncating to leading term in $\Delta t, \Delta x$, we obtain the modified equation

$$\frac{\partial u}{\partial t} - c\frac{\partial u}{\partial x} = \frac{1}{2}c\Delta x(1-r)\frac{\partial^2 u}{\partial x^2}, \tag{3}$$

where $r = c\Delta t/\Delta x$ is the Courant-Friedrichs-Lewy (CFL) number (Courant et al., 1952). Notice that in the limit $\Delta t, \Delta x \to 0$ with r fixed, one recovers the original transport equation, but for finite step sizes, the upwind scheme is really described by the modified equation (3). In other words, this truncated equation describes the leading order, non-trivial behavior of the finite difference scheme.

From the modified equation (3), one can immediately deduce a number of interesting properties. First, the error from the upwind scheme is diffusive in nature, due to the presence of the second order spatial derivative on the right hand side. Second, we observe that if the CFL number r is greater than 1, then the coefficient for the diffusive term becomes negative and this results in instability. This is the well-known *CFL condition*. This places a fundamental limit on the spatial resolution for fixed temporal resolution with regards to the stability of the algorithm. Lastly, the error term is proportional to Δx for fixed r, thus it may be considered a first order method.

Now, another possible proposal for discretizing (1) is the Lax-Wendroff (LW) scheme (Lax & Wendroff, 1960):

$$U_{n+1,j} = U_{n,j} + \Delta t \left(c\frac{U_{n,j+1} - U_{n,j-1}}{2\Delta x} - c\Delta t \frac{U_{n,j+1} - 2U_{n,j} + U_{n,j-1}}{2\Delta x^2} \right), \quad (4)$$

whose modified equation is

$$\frac{\partial u}{\partial t} - c\frac{\partial u}{\partial x} = \frac{1}{6}c\Delta x^2(r^2 - 1)\frac{\partial^3 u}{\partial x^3}. \quad (5)$$

Comparing with (3), we observe that the LW scheme error is of higher order (Δx^2), but at the cost of introducing dispersive, instead of diffusive errors due to the presence of the third derivative. These findings are in excellent agreement with the actual behavior of their respective discrete numerical schemes (Warming & Hyett, 1974).

We stress here that if we simply took the trivial leading order, the right hand sides of (3) and (5) disappear and vital information, including stability, accuracy and the nature of the error term will be lost. The ability to capture the effective dynamical behavior of finite difference schemes is the key strength of the modified equations approach, which has become the primary tool in analyzing and improving finite difference algorithms. The goal of our work is to extend this approach to analyze stochastic algorithms.

B Summary of SDE terminologies and results

Here, we summarize various SDE terminologies and results we have used throughout the main paper and also subsequent derivations. A particular important result is the Itô formula (Sec. B.2), which is used throughout this work for deriving moment equations. For a thorough reference on the subject of stochastic calculus and SDEs, we suggest Oksendal (2013).

B.1 Stochastic differential equations

Let $T > 0$. An Itô stochastic differential equation on the interval $[0, T]$ is an equation of the form

$$dX_t = b(X_t, t)dt + \sigma(X_t, t)dW_t, \qquad X_0 = x_0, \quad (6)$$

where $X_t \in \mathbb{R}^d$, $b : \mathbb{R}^d \times [0,T] \to \mathbb{R}^d$, $\sigma : \mathbb{R}^d \times [0,T] \to \mathbb{R}^{d\times l}$ and W_t is a l-dimensional Wiener process, or Brownian motion. This is a mathematically sound way of expressing the intuitive notion of SDEs being ODEs plus noise:

$$\dot{X}_t = b(X_t, t) + \text{“noise”} \tag{7}$$

The equation (6) is really a “shorthand” for the integral equation

$$X_t - x_0 = \int_0^t b(X_s, s)ds + \int_0^t \sigma(X_s, s)dW_s. \tag{8}$$

The last integral is defined in the Itô sense, i.e.

$$\int_0^t F_s dW_s := \lim_{n\to\infty} \sum_{[s_{i-1}, s_i]\in\pi_n} F_{s_{i-1}}(W_{s_i} - W_{s_{i-1}}), \tag{9}$$

where π_n is a sequence of n-partitions of $[0,t]$ and the limit represents convergence in probability. In (6), b is known as the *drift*, and σ is known as the *diffusion matrix*. When they satisfy Lipschitz conditions, one can show that (6) (or (8)) has a unique strong solution (Oksendal (2013), Chapter 5). For our purposes in this paper, we consider the special case where b, σ do not depend on time and we set $d = l$ so that σ is a square matrix.

To perform calculus, we need an important result that generalizes the notion of chain rule to the stochastic setting.

B.2 Itô formula

Itô formula, also known as Itô’s lemma, is the extension of the chain rule of ordinary calculus to the stochastic setting. Let $\phi \in C^{2,1}(\mathbb{R}^d \times [0,T])$ and let X_t be a stochastic process satisfying the SDE (6), and thus (8). Then, the stochastic process $\phi(X_t, t)$ is again an Itô process satisfying

$$\begin{aligned} d\phi(X_t, t) = &\left[\partial_t\phi(X_t, t) + \left(\nabla\phi(X_t, t)^T b(X_t, t) + \frac{1}{2}\mathrm{Tr}[\sigma(t, X_t)^T H\phi(t, X_t)\sigma(t, X_t)]\right)\right] dt \\ &+ \left[\nabla\phi(X_t, t)^T\sigma(X_t, t)\right] dW_t, \end{aligned} \tag{10}$$

where ∇ denotes gradient with respect to the first argument and $H\phi$ denotes the Hessian, i.e. $H\phi_{(ij)} = \partial^2\phi/\partial x_{(i)}\partial x_{(j)}$. The formula (10) is the Itô formula. If ϕ is not a scalar but a vector, then each of its component satisfy (10). Note that if $\sigma = 0$, this reduces to the chain rule of ordinary calculus.

B.3 The Ornstein-Uhlenbeck process

An important solvable SDE is the Ornstein-Uhlenbeck (OU) process Uhlenbeck & Ornstein (1930). Consider $d = 1$, $b(x,t) = \theta(\xi - x)$ and $\sigma(x,t) = \sigma > 0$, with $\theta > 0$, $\sigma > 0$ and $\xi \in \mathbb{R}$. Then we have the SDE

$$dX_t = \theta(\xi - X_t)dt + \sigma dW_t, \qquad X_0 = x_0. \tag{11}$$

To solve this equation, we change variables $x \mapsto \phi(x,t) = xe^{\theta t}$. Applying Itô formula, we have

$$d\phi(X_t,t) = \theta\xi e^{\theta t}dt + \sigma e^{\theta t}dW_t, \tag{12}$$

which we can integrate from 0 to T to get

$$X_t = x_0 e^{-\theta t} + \xi(1 - e^{-\theta t}) + \sigma \int_0^t e^{-\theta(t-s)}dW_s. \tag{13}$$

This is a path-wise solution to the SDE (11). To infer distributional properties, we do not require such precise solutions. In fact, we only need the distribution of the random variable X_t at any fixed time $t \in [0,T]$. Observe that X_t is really a Gaussian process, since the integrand in the Wiener integral is deterministic. Hence, we need only calculate its moments. Taking expectation on (13), we get

$$\mathbb{E}X_t = x_0 e^{-\theta t} + \xi(1 - e^{-\theta t}). \tag{14}$$

To obtain the covariance function, we see that

$$\mathbb{E}(X_t - \mathbb{E}X_t)(X_s - \mathbb{E}X_s) = \sigma^2 \mathbb{E}\left[\int_0^t e^{\theta(u-s)}dW_u \int_0^t e^{\theta(v-s)}dW_v\right]. \tag{15}$$

This can be evaluated by using *Itô's isometry*, which says that for any W_t adapted process ϕ_t, ψ_t, we have

$$\mathbb{E}\left[\int_0^t \phi_u dW_u \int_0^t \psi_v dW_v\right] = \mathbb{E}\left[\int_0^t \phi_s \psi_s ds\right]. \tag{16}$$

We get, for $s \leq t$

$$\text{cov}(X_s, X_t) = \frac{\sigma^2}{2\theta}\left(e^{-\theta|t-s|} + e^{-\theta|t+s|}\right), \tag{17}$$

and in particular, for fixed $t \in [0,T]$, we have

$$\text{Var}(X_t) = \frac{\sigma^2}{2\theta}(1 - e^{-2\theta t}). \tag{18}$$

Hence, we have

$$X_t \sim \mathcal{N}\left(x_0 e^{-\theta t} + \xi(1 - e^{-\theta t}), \frac{\sigma^2}{2\theta}(1 - e^{-2\theta t})\right). \tag{19}$$

In Sec. 3.1 in the main paper, the solution of the SME is the OU process with $\theta = 2(1+\eta), \xi = 0, \sigma = 2\sqrt{\eta}$. Making these substitutions, we obtain

$$X_t \sim \mathcal{N}\left(x_0 e^{-2(1+\eta)t}, \frac{\eta}{1+\eta}\left(1 - e^{-4(1+\eta)t}\right)\right).$$

B.4 Numerical solution of SDEs

Unfortunately, most SDEs are not amenable to exact solutions. Often, we resort to numerical methods. The simplest method is the *Euler-Maruyama method*. This extends the Euler method for ODEs to SDEs. Fix a time discretization size $\delta > 0$ and define $\tilde{X}_k = X_{k\delta}$, then we can iterate the finite difference equation

$$\tilde{X}_{k+1} = \tilde{X}_k + \delta b(\tilde{X}_k, k\delta) + \sigma(\tilde{X}_k, k\delta)(W_{(k+1)\delta} - W_{k\delta}). \tag{20}$$

By definition, $W_{(k+1)\delta} - W_{k\delta} \sim \mathcal{N}(0, \delta I)$, and are independent for each k. Here, I is the identity matrix. Hence, we have the Euler-Maruyama scheme

$$\tilde{X}_{k+1} = \tilde{X}_k + \delta b(\tilde{X}_k, k\delta) + \sqrt{\delta}\sigma(\tilde{X}_k, k\delta)Z_k, \tag{21}$$

where $Z_k \overset{i.i.d.}{\sim} \mathcal{N}(0, I)$.

One can show that the Euler-Maruyama method (21) is a first order weak approximation (c.f. Def. 1 in main paper) to the SDE (6). However, it is only a order $1/2$ scheme in the strong sense (Kloeden & Platen, 2011), i.e.

$$\mathbb{E}|X_{k\delta} - \tilde{X}_{k\delta}| < C\delta^{\frac{1}{2}}. \tag{22}$$

With more sophisticated methods, one can design higher order schemes (both in the strong and weak sense), see Milstein (1986).

B.5 Stochastic asymptotic expansion

Besides numerics, if there exists small parameters in the SDE, we can proceed with stochastic asymptotic expansions Freidlin et al. (2012). This is the case for the SME, which has a small $\eta^{1/2}$ multiplied to the noise term. Let us consider a time-homogeneous SDE of the form

$$dX^\epsilon_t = b(X^\epsilon_t)dt + \epsilon\sigma(X^\epsilon_t)dW_t \tag{23}$$

where $\epsilon \ll 1$. The idea is to follow standard asymptotic analysis and write X^ϵ_t as an asymptotic series

$$X^\epsilon_t = X_{0,t} + \epsilon X_{1,t} + \epsilon^2 X_{2,t} + \dots. \tag{24}$$

We substitute (24) into (23) and assuming smoothness of b and σ, we expand

$$\begin{aligned} b_\epsilon(X^\epsilon_t) &= b(X_{0,t}) + \epsilon\nabla b(X_{0,t})X_{1,t} + \mathcal{O}(\epsilon^2) \\ \sigma(X^\epsilon_t) &= \sigma(X_{0,t}) + \epsilon\nabla\sigma(X_{0,t})X_{1,t} + \mathcal{O}(\epsilon^2) \end{aligned} \tag{25}$$

to get

$$\begin{aligned} dX_{0,t} &= b(X_{0,t})dt, \\ dX_{1,t} &= \nabla b(X_{0,t})X_{1,t}dt + \sigma(X_{0,t})dW_t, \\ &\vdots \end{aligned} \tag{26}$$

and $X_{0,0} = x_0, X_{1,0} = 0$. In general, the equation for $X_{i,t}$ are linear stochastic differential equations with time-dependent coefficients depending on $\{X_{0,t}, X_{1,t}, \dots, X_{i-1,t}\}$ and the initial conditions are $X_{0,0} = x_0$, $X_{i,0} = 0$ for all $i \geq 1$. Hence, the asymptotic equations can be solved sequentially to obtain an estimate of X_t to arbitrary order in ϵ. The equations for higher order terms become messy quickly, but they are always linear in the unknown, as long as all the previous equations are solved. For more details on stochastic asymptotic expansions, the reader is referred to Freidlin et al. (2012).

B.6 Asymptotics of the SME

We now derive the first two asymptotic equations of the SME. we take $\epsilon = \sqrt{\eta}$, $b = -\nabla f$ ($\mathcal{O}(\eta)$ term can be ignored for first two terms) and $\sigma = \Sigma^{1/2}$. Then, (26) becomes

$$dX_{0,t} = -\nabla f(X_{0,t})dt, \tag{27}$$

$$dX_{1,t} = -Hf(X_{0,t})X_{1,t}dt + \Sigma(X_{0,t})^{\frac{1}{2}}dW_t, \tag{28}$$

where $Hf_{(ij)} = \partial_{(i)}\partial_{(j)}f$ is the Hessian of f.

In the following analysis, we shall assume that the truncated series approximation

$$\hat{X}_t = X_{0,t} + \sqrt{\eta}X_{1,t}, \tag{29}$$

where $X_{0,t}, X_{1,t}$ satisfy (27) and (28), describes the leading order stochastic dynamics of the SGD. Now, let us analyze the asymptotic equations in detail. First, we assume that the ODE (27) has a unique solution $X_{0,t}, t \geq 0$ with $X_{0,0} = x_0$. This is true if for example, ∇f is locally Lipschitz. Next, let us define the non-random functions

$$\begin{aligned} H_t &= Hf(X_{0,t}), \\ \Sigma_t &= \Sigma(X_{0,t}). \end{aligned} \tag{30}$$

Both H and σ are $d \times d$ matrices for each t. Then, (28) becomes the time-inhomogeneous linear SDE

$$dX_{1,t} = -H_t X_{1,t} + \Sigma_t^{\frac{1}{2}}dW_t, \tag{31}$$

with $X_{1,0} = 0$. Since the drift is linear and the diffusion matrix is constant (i.e. independent of $X_{1,t}$), $X_{1,t}$ is a Gaussian process. Hence we need only calculate its mean and covariance using Itô formula (see B.2). We have

$$\mathbb{E}X_{1,t} = 0, \tag{32}$$

and the covariance matrix $S_t = \text{Cov}(X_{1,t})$ satisfies the differential equation

$$\frac{d}{dt}S_t = -S_t H_t - H_t S_t + \Sigma_t, \tag{33}$$

with $S_0 = 0$. This equation is a linearized version of the *Riccati equation* and there are simple closed-form solutions under special conditions, e.g. $d = 1$ or H_t is constant.

Hence, we conclude that the asymptotic approximation $\hat{X}_t$ is a Gaussian process with distribution

$$\hat{X}_t \sim \mathcal{N}(X_{0,t}, \eta S_t), \tag{34}$$

where $X_{0,t}$ solves the ODE (27) and S_t solves the ODE (33), with H_t, Σ_t given by (30).

Remark 1. *At this point, it is important to discuss the validity of the asymptotic approximation* (34)*, and the SME approximation* (35) *in general. What we prove in Sec. C and is shown in Freidlin et al. (2012) is that for fixed T, we can take $\eta = \eta(T)$ small enough so that the SME and its asymptotic expansion is a good approximation of the distribution of the SGD iterates. What we did not prove is that for fixed η, the approximations hold for arbitrary T. In particular, it is not hard to construct systems where for fixed η, both the SME and the asymptotic expansion fails when T is large enough. To prove the second general statement requires further assumptions, particularly on the distribution of f_i's. This is out of the scope of the current work.*

C Formal Statement and proof of Thm. 1

Theorem 1 (Stochastic modified equations)**.** *Let $\alpha \in \{1, 2\}$, $0 < \eta < 1$, $T > 0$ and set $N = \lfloor T/\eta \rfloor$. Let $x_k \in \mathbb{R}$, $0 \leq k \leq N$ denote a sequence of SGD iterations defined by (2). Define $X_t \in \mathbb{R}^d$ as the stochastic process satisfying the SDE*

$$dX_t = -\nabla(f(X_t) + \frac{1}{4}(\alpha - 1)\eta|\nabla f(X_t)|^2)dt + (\eta\Sigma(X_t))^{\frac{1}{2}}dW_t \tag{35}$$

$X_0 = x_0$ and $\Sigma(x) = \frac{1}{n}\sum_{i=1}^{n}(\nabla f(x) - \nabla f_i(x))(\nabla f(x) - \nabla f_i(x))^T$.

Fix some test function $g \in G$ (c.f. Def. 1 in main paper). Suppose further that the following conditions are met:

(i) *$\nabla f, \nabla f_i$ satisfy a Lipschitz condition: there exists $L > 0$ such that*

$$|\nabla f(x) - \nabla f(y)| + \sum_{i=1}^{n} |\nabla f_i(x) - \nabla f_i(y)| \leq L|x - y|.$$

(ii) *f, f_i and its partial derivatives up to order 7 belong to G.*

(iii) *$\nabla f, \nabla f_i$ satisfy a growth condition: there exists $M > 0$ such that*

$$|\nabla f(x)| + \sum_{i=1}^{n} |\nabla f_i(x)| \leq M(1 + |x|).$$

(iv) *g and its partial derivatives up to order 6 belong to G.*

Then, there exists a constant $C > 0$ independent of η such that for all $k = 0, 1, \ldots, N$, we have

$$|\mathbb{E}g(X_{k\eta}) - \mathbb{E}g(x_k)| \leq C\eta^{\alpha}.$$

That is, the equation (35) *is an order α weak approximation of the SGD iterations.*

The basic idea of the proof is similar to the classical approach in proving weak convergence of discretization schemes of SDEs outlined in the seminal papers by Milstein (Milstein (1975, 1979, 1986, 1995)). The main difference is that we wish to establish that the continuous SME is an approximation of the discrete SGD, instead of the other way round, which is the case dealt by classical approximation theorems of SDEs with finite difference schemes. In the following, we first show that a one-step approximation has order $\eta^{\alpha+1}$ error, and then deduce, using the general result in Milstein (1986), that the overall global error is of order η^{α}.

It is well known that a second order weak convergence discretization scheme for a SDE is not trivial. The classical Euler-Maruyama scheme, as well as the Milstein scheme are both first order weak approximations. However, in our case the problem simplifies significantly. This is because the noise we are trying to model is small, so that from the outset, we may assume that $b(x) = \mathcal{O}(1)$ but $\sigma(x) = \mathcal{O}(\eta^{1/2})$, i.e. we set $\sigma(x) = \eta^{1/2}\tilde{\sigma}(x)$ where $\tilde{\sigma} = \mathcal{O}(1)$ and deduce the appropriate expansions. For brevity, in the following we will drop the tilde and simply denote the noise term of the SDE by $\eta^{1/2}\sigma$.

In the subsequent proofs we will make repeated use of Taylor expansions in powers of η. To simplify presentation, we introduce the shorthand that whenever we write $\mathcal{O}(\eta^{\alpha})$, we mean that there exists a function $K(x) \in G$ (c.f. Def. 1 in main text) such that the error terms are bounded by $K(x)\eta^{\alpha}$. For example, we write

$$b(x+\eta) = b_0(x) + \eta b_1(x) + \mathcal{O}(\eta^2) \tag{36}$$

to mean: there exists $K \in G$ such that

$$|b(x+\eta) - b_0(x) - \eta b_1(x)| \leq K(x)\eta^2. \tag{37}$$

These results can be deduced easily using Taylor's theorem with a variety of forms of the remainder, e.g. Lagrange form. We omit such routine calculations. We also denote the partial derivative with respect to $x_{(i)}$ by $\partial_{(i)}$.

First, let us prove a lemma regarding moments of SDEs with small noise.

Lemma 1. *Let $0 < \eta < 1$. Consider a stochastic process X_t, $t \geq 0$ satisfying the SDE*

$$dX_t = b(X_t) + \eta^{\frac{1}{2}}\sigma(X_t)dW_t \tag{38}$$

with $X_0 = x \in \mathbb{R}^d$ and b, σ together with their derivatives belong to G. Define the one-step difference $\Delta = X_\eta - x$, then we have

(i) $\mathbb{E}\Delta_{(i)} = b_{(i)}\eta + \frac{1}{2}[\sum_{j=1}^{d} b_{(j)}\partial_{(j)}b_{(i)}]\eta^2 + \mathcal{O}(\eta^3)$.

(ii) $\mathbb{E}\Delta_{(i)}\Delta_{(j)} = [b_{(i)}b_{(j)} + \sigma\sigma^T_{(ij)}]\eta^2 + \mathcal{O}(\eta^3)$.

(iii) $\mathbb{E}\prod_{j=1}^{s}\Delta_{(i_j)} = \mathcal{O}(\eta^3)$ *for all $s \geq 3$, $i_j = 1, \ldots, d$.*

All functions above are evaluated at x.

Proof. One way to establish (i)-(iii) is to employ the Ito-Taylor expansion (see Kloeden & Platen (2011), Chapter 5) on the random variable X_η around x and calculating the moments. Here, we will employ instead the method of semigroup expansions (see Hille & Phillips (1996), Chapter XI), which works directly on expectation functions. The generator of the stochastic process (38) is the operator L acting on sufficiently smooth functions $\phi : \mathbb{R}^d \to \mathbb{R}$, and is defined by

$$L\phi = \sum_{i=1}^{d} b_{(i)}\partial_{(i)}\phi + \frac{1}{2}\eta^2 \sum_{i,j=1}^{d} \sigma\sigma^T_{(ij)}\partial_{(i)}\partial_{(j)}\phi, \tag{39}$$

A classical result on semigroup expansions (Hille & Phillips (1996), Chapter XI) states that if ϕ and its derivatives up to order 6 belong to G, then

$$\mathbb{E}\phi(X_\eta) = \phi(x) + L\phi(x)\eta + \frac{1}{2}L^2\phi(x)\eta^2 + \mathcal{O}(\eta^3). \tag{40}$$

Now, let $t \in \mathbb{R}^d$ and consider the moment-generating function (MGF)

$$M(t) = \mathbb{E}e^{t\cdot\Delta}. \tag{41}$$

To ensure its existence we may instead set t to be purely imaginary, i.e. $t = is$ where s is real. Then, (41) is known as the characteristic function (CF). The important property we make use of is that the moments of Δ are found by differentiating the MGF (or CF) with respect to t. In fact, we have

$$\mathbb{E}\prod_{j=1}^{s}\Delta_{(i_j)} = \frac{\partial^s M(t)}{\prod_{j=1}^{s}\partial t_{(i_j)}}\bigg|_{t=0}, \tag{42}$$

where $i_j = 1, \ldots, d$. We now expand $M(t)$ in powers of η using formula (40). We get,

$$\begin{aligned} M(t) =& 1 + \left[\sum_{i=1}^{d} b_{(i)}t_{(i)}\eta + \frac{1}{2}\sum_{i,j=1}^{d} b_{(i)}t_{(j)}\partial_{(i)}b_{(j)}\eta^2\right] \\ &+ \left[\frac{1}{2}\eta^2(\sum_{i=1}^{d} b_{(i)}t_{(i)})^2 + \frac{1}{2}\sum_{i,j=1}^{d}\sigma\sigma^T_{(ij)}t_{(i)}t_{(j)}\right] + \mathcal{O}(\eta^3). \end{aligned} \tag{43}$$

All functions are again evaluated at x. Finally, we apply formula (42) to deduce (i)-(iii). □

Next, we have an equivalent result for one SGD iteration.

Lemma 2. *Let $0 < \eta < 1$. Consider x_k, $k \geq 0$ satisfying the SGD iterations*

$$x_{k+1} = x_k - \eta\nabla f_{\gamma_k}(x_k) \tag{44}$$

with $x_0 = x \in \mathbb{R}^d$. Define the one-step difference $\bar{\Delta} = x_1 - x$, then we have

(i) $\mathbb{E}\bar{\Delta}_{(i)} = -\partial_{(i)}f\eta$

(ii) $\mathbb{E}\bar{\Delta}_{(i)}\bar{\Delta}_{(j)} = \partial_{(i)}f\partial_{(j)}f\eta^2 + \Sigma_{(ij)}\eta^2$.

(iii) $\mathbb{E}\prod_{j=1}^{s}\bar{\Delta}a_{(i_j)} = \mathcal{O}(\eta^3)$ *for all* $s \geq 3$, $i_j = 1, \ldots, d$.

where $\Sigma = \frac{1}{n}\sum_{i=1}^{n}(\nabla f - \nabla f_i)(\nabla f - \nabla f_i)^T$*. All functions above are evaluated at* x*.*

Proof. From definition (44) and the definition of Σ, the results are immediate. □

Now, we will need a key result linking one step approximations to global approximations due to Milstein. We reproduce the theorem, tailored to our problem, below. The more general statement can be found in Milstein (1986).

Theorem 2 (Milstein, 1986)**.** *Let* α *be a positive integer and let the assumptions in Theorem 1 hold. If in addition there exists* $K_1, K_2 \in G$ *so that*

$$|\mathbb{E}\prod_{j=1}^{s}\Delta_{(i_j)} - \mathbb{E}\prod_{j=1}^{s}\bar{\Delta}_{(i_j)}| \leq K_1(x)\eta^{\alpha+1},$$

for $s = 1, 2, \ldots, 2\alpha+1$ *and*

$$\mathbb{E}\prod_{j=1}^{2\alpha+2}|\bar{\Delta}_{(i_j)}| \leq K_2(x)\eta^{\alpha+1}.$$

Then, there exists a constant C *so that for all* $k = 0, 1, \ldots, N$ *we have*

$$|\mathbb{E}g(X_{k\eta}) - \mathbb{E}g(x_k)| \leq C\eta^{\alpha}$$

Proof. See Milstein (1986), Theorem 2 and Lemma 5. □

Proof of Theorem 1

We are now ready to prove theorem 1 by checking the conditions in theorem 2 with $\alpha = 1, 2$. The second condition is implied by Lemma 2. The first condition is implied by Lemma 1 and Lemma 2 with the choice

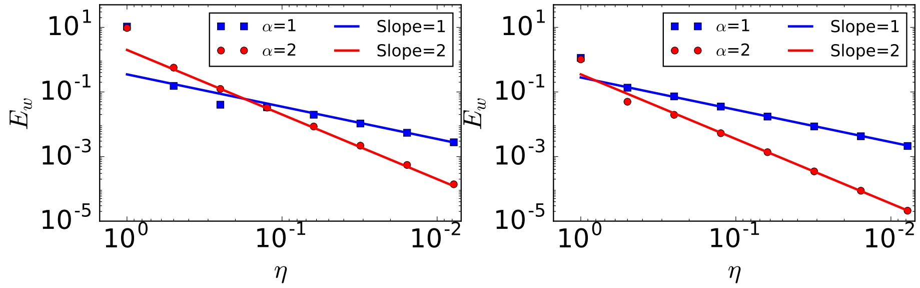
$$\begin{aligned} b(x) &= -\nabla(f(x) + \frac{1}{4}\eta(\alpha-1)|\nabla f(x))|^2, \\ \sigma(x) &= \Sigma(x)^{\frac{1}{2}}. \end{aligned}$$

□

To illustrate our approximation result, let us calculate, using Monte-Carlo simulations, the weak error of the SME approximation

$$E_w = |\mathbb{E}g(X_{N\eta}) - \mathbb{E}g(x_N)|, \tag{45}$$

for $\alpha = 1, 2$ v.s. η for different f, f_i and generic polynomial test functions g. The results are shown in Fig. 1. We see that we have order α weak convergence, even when some conditions of the above theorem are not satisfied (Fig. 1(b)).



(a)

(b)

Figure 1: Weak error E_w, as defined in (45) with $\alpha = 1, 2$, v.s. learning rate η for two different choices of f, f_i. All errors are averaged over 10^{12} samples of SGD trajectories up to $T = 1.0$. The initial condition is $x_0 = 1$. The SMEs moments are solved exactly since they involve linear drifts. (a) Quadratic objective with $n = 2$, $f_i = (x - \gamma_i)^2$ where $\gamma_i \in \{\pm 1\}$. The total objective is $f(x) = x^2 + 1$. The test function is $g(x) = x + x^2 + x^3$. (b) Non-convex f_i's with $n = 2$, $f_i(x) = (x - \gamma_i)^2 + \gamma_i x^3$ where $\gamma_i \in \{\pm 1\}$. The total objective is the same $f(x) = x^2 + 1$. We chose $g(x) = x$ so that $\mathbb{E} g(X_T)$ has closed form solution. Note that for this choice of f_i, the condition (iii) of Theorem 1 is not satisfied. Nevertheless, in both cases, we observe that the weak error decreases with η like $E_w \sim \eta^{\alpha}$.

Remark 2. *From above, we also observe that if we pick $b(x) = -\nabla f(x)$ and $\sigma(x)$ to be any function in G (and its sufficiently high derivatives are also in G), then we have matching moments up to order η^2 and hence we can conclude that for this choice, the resulting SDE is a first order weak approximation of the SGD, with $|\mathbb{E} g(X_{k\eta}) - \mathbb{E} g(x_k)| \leq C\eta$, $k = 0, 1, \ldots, N$. In particular, the deterministic gradient flow is a first order weak approximation of the SGD. Hence, just like traditional modified equations, our SME* (35) *($\alpha = 2$) is the next order approximation of the underlying algorithm.*

However, we stress that for our first order SME with $\alpha = 1$, i.e. the choice $b = -\nabla f$ and $\sigma = \Sigma^{\frac{1}{2}}$, the fact that we did not the improve the order of weak convergence from the deterministic gradient flow does not mean that this is a equally bad approximation. The constant C in the weak error depends on the choice of Σ and in fact, it can be shown empirically that with this choice, we do have lower weak error $|\mathbb{E} g(X_{k\eta}) - \mathbb{E} g(x_k)|$, but the order of convergence of the weak error as $\eta \to 0$ is the same. An analytical justification must then rely on using the Itô-Taylor expansion to obtain precise estimates for the factor C (see e.g. Talay & Tubaro (1990)). This is beyond the scope of the current paper.

Remark 3. *The Lipschitz condition (i) is to ensure that the SME has a unique strong solution with uniformly bounded moments Milstein (1986). If we allow weak solutions and establish uniform boundedness of moments by other means (more assumptions on the growth and direction of ∇f for large x), then condition (i) is expected to be relaxed although the technical details will be tedious.*

Condition (iii) in Theorem 1 appears to be the most stringent one and in fact it may limit applications to problems with objectives that have more than quadratic growth.

However, closer inspection tells us it can also be relaxed. For example, if there exists an invariant distribution that concentrates on a compact subset of $\mathbb{R}^d$ then as $\eta \to 0$, x_k's would be bounded with high probability, and hence for large x we may replace f, f_i with a version that satisfies the growth condition in (iii). Further work is needed to make this precise but we can already see in Fig. 1(b) that we have quadratic weak convergence even when (iii) is not satisfied.

Remark 4. *The regularity conditions on f and g in Theorem 1 are inherited from Theorem 2 in Milstein (1986). For smooth objectives, polynomial growth conditions are usually not restrictive. Still, with care, these should be relaxed since in our case the small noise helps to reduce the number of terms containing higher derivatives in various Taylor and Itô-Taylor expansions. Proving a more general version of Theorem 1 will be left as future work.*

D Derivation of SMEs

In this section, we include more detailed derivations of the SMEs used in the main paper. For brevity, we do not include rigorous proofs of approximation statements for SGD variants in Sec. D.2 and D.3, but only heuristic justifications. Proving rigorous statements for these approximations can be done by modifying the proof of Thm. 1.

D.1 SME for the simple quadratic example

We start with the example in Sec. 3.1 of the main paper. Let $n = 2$, $d = 1$ and set $f(x) = x^2 + 1$ with $f_1(x) = (x-1)^2$ and $f_2(x) = (x+1)^2$. The SGD iterations picks at random between f_1 and f_2 and performs descent with their respective gradients. Recall that the (second order) SME is given by

$$dX_t = -(f'(X_t) + \frac{1}{2}\eta f'(X_t)f''(X_t))dt + (\eta\Sigma(X_t))^{\frac{1}{2}}dW_t. \tag{46}$$

Now, $f'(x) = 2x$, $f''(x) = 2$ and

$$\Sigma(x) = \frac{1}{2}\sum_{i=1}^{2}(f_i'(x) - f_i(x))^2 = 4 \tag{47}$$

and hence the SME is

$$dX_t = -2(1+\eta)X_t dt + 2\sqrt{\eta}dW_t. \tag{48}$$

D.2 SME for learning rate adjustment

The SGD iterations with learning rate adjustment is

$$x_{k+1} = x_k - \eta u_k f'(x_k), \tag{49}$$

where $u_k \in [0, 1]$ is the learning rate adjustment factor. η is the maximum allowed learning rate. There are two reasons we introduce this hyper-parameter. First, gradients cannot be arbitrarily large since that will cause instabilities. Second, the SME is only an approximation of the SGD for small learning rates, and so it is hard to justify the approximation for large η.

In this case, deriving the corresponding SME is extremely simple. Notice that we can define $g_{i,k}(x_k) = u_k f_i(x_k)$, $g_k = u_k f(x_k)$. Then, the iterations above is simply

$$x_{k+1} = x_k - \eta g'_{\gamma_k,k}(x_k), \tag{50}$$

whose (first order) SME is by Thm. 1

$$dX_t = -g'(X_t)dt + (\eta u_t^2 \Sigma(X_t))^{\frac{1}{2}} dW_t. \tag{51}$$

And hence the SME for SGD with learning rate adjustments is

$$dX_t = -u_t f'(X_t)dt + u_t(\eta\Sigma(X_t))^{\frac{1}{2}} dW_t. \tag{52}$$

D.3 SME for SGD with momentum

First let us consider the constant momentum parameter case. The SGD with momentum is the paired update

$$\begin{aligned} v_{k+1} &= \mu v_k - \eta f'_{\gamma_k}(x_k), \\ x_{k+1} &= x_k + v_{k+1}. \end{aligned} \tag{53}$$

To derive and SME, notice that we can write the above as

$$\begin{aligned} v_{k+1} &= v_k + \eta\left(-\frac{1-\mu}{\eta} v_k - f'(x_k)\right) + \eta\left(f'(x_k) - f'_{\gamma_k}(x_k)\right), \\ x_{k+1} &= x_k + \eta\left(\frac{v_{k+1}}{\eta}\right). \end{aligned} \tag{54}$$

Recall that since we are looking at first order weak approximations, it is sufficient to compare to the Euler-Maruyama discretization (Sec. B.4). We observe that the above can be seen as an Euler-Maruyama discretization of the coupled SDE

$$\begin{aligned} dV_t &= (-\frac{1-\mu}{\eta} V_t - f'(X_t))dt + (\eta\Sigma(X_t))^{\frac{1}{2}} dW_t, \\ dX_t &= \frac{1}{\eta} V_t dt, \end{aligned} \tag{55}$$

with the usual choice of $\Sigma(x)$. Hence, this is the first order SME for the SGD with momentum having a constant momentum parameter μ. For time-varying momentum parameter, we just replace μ by μ_t to get

$$\begin{aligned} dV_t &= (-\frac{1-\mu_t}{\eta} V_t - f'(X_t))dt + (\eta\Sigma(X_t))^{\frac{1}{2}} dW_t, \\ dX_t &= \frac{1}{\eta} V_t dt. \end{aligned} \tag{56}$$

E Solution of optimal control problems

E.1 Brief introduction to optimal control

We first introduce some basic terminologies and results on optimal control theory to pave way for our solutions to optimal control problems for the learning rate and momentum parameter. For simplicity, we restrict to one state dimension ($d = 1$), but similar equations hold for multiple dimensions. For a more thorough introduction to optimal control theory and calculus of variations, we refer the reader to Liberzon (2012).

Let $t \in [0, T]$ and consider the ODE

$$\frac{d}{dt} z_t = \Phi(z_t, u_t), \tag{57}$$

where $z_t, u_t \in \mathbb{R}$ and $\Phi : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$. The variable z_t describes the evolution of some state and u_t is the control variable, which can affect the state dynamics. Consider the control problem of minimizing the cost functional

$$C(u) = \int_0^T L(z_s, u_s) ds + G(z_T) \tag{58}$$

with respect to u_t, subject to z_t satisfying the ODE (57) with prescribed initial condition $z_0 \in \mathbb{R}$. The function L is known as the running cost and G is the terminal cost. Usually, we also specify some control set $U \subset \mathbb{R}$ so that we only consider $u : [0, T] \to U$. The full control problem reads

$$\min_{u:[0,T]\to U} C(u) \text{ subject to (57).} \tag{59}$$

Note that additional path constraints can also be added and (57) can also be made time-inhomogeneous, but for our purposes it is sufficient to consider the above form.

There are two principal ways of solving optimal control problems: either dynamic programming through the Hamilton-Jacobi-Bellman (HJB) equation (Bellman, 1956), or using the Pontryagin's maximum principle (PMP) (Pontryagin, 1987). In this section, we will only discuss the HJB method as this is the one we employ to solve the relevant control problems in this paper.

E.2 Dynamic programming and the HJB equation

The first way to solve (59) is through the dynamic programming principle. For $t \in [0, T]$ and $z \in \mathbb{R}$, define the value function

$$\begin{aligned} V(z,t) = \min_{u:[t,T]\to U} &\int_t^T L(z_s, u_s) ds + G(z_T), \\ &\text{subject to} \\ &\frac{d}{ds} z_s = \Phi(z_s, u_s), \quad s \in [t, T], \\ &z(t) = z. \end{aligned} \tag{60}$$

Notice that if there exists a solution to (59), then the value of the minimum cost is $V(z_0, 0)$. The dynamic programming principle allows us to derive a recursion on the function V, in the form of a partial differential equation (PDE)

$$\begin{aligned} \partial_t V(z,t) + \min_{u \in U}\{\partial_z V(z,t)\Phi(z,u) + L(z,u)\} = 0, \\ V(T,z) = G(z). \end{aligned} \tag{61}$$

This is known as the *Hamilton-Jacobi-Bellman equation* (HJB). Note that this PDE is solved backwards in time. The derivation of this PDE can be found in most references on optimal control, e.g. in Liberzon (2012). The main idea is the dynamic programming principle: for any t the $[t, T]$-portion of the optimal trajectory must again be optimal.

After solving the HJB (61), we can then obtain the optimal control u_t^* as function of the state process z_t and t, given by

$$u_t^* = \arg\min_{u \in U}\{\partial_z V(z_t, t)\Phi(z_t, u) + L(z_t, u)\}. \tag{62}$$

In some cases, we find that the optimal control is independent of time and is strictly of a *feed-back control law*, i.e.

$$u_t^* = u^*(z_t) \tag{63}$$

for some function $u^* : \mathbb{R} \to U$. This is the case for the problems considered in this paper. With the optimal control found in (62), we can then substitute $u_t = u_t^*$ in (57) to obtain the optimally controlled process z_t^*.

In summary, to solve the optimal control problem (59), we first solve the HJB PDE (61), and then solve for the optimal control (62), and lastly (if necessary) solve the optimally controlled state process by substituting the solution of (62) into (57). Sometimes, the optimal control (62) can be solved without fully solving the HJB for V, e.g. when $L = 0$ and one can infer the sign of ∂V. This is the case for the two control problems we encounter in this paper. The solution to (62) is the most important for all practical purposes since it gives a way to adjust the control parameters on-the-fly, especially when we have a feed back control law.

E.3 Solution of the learning rate control problem

Now, let us apply the HJB equations (Sec. E.2) to solve the learning rate control problem. Recall from Sec. 4.1.2 that we wish to solve

$$\begin{aligned} \min_{u:[0,T]\to[0,1]} m_T \text{ subject to} \\ \frac{d}{dt}m_t = -2au_t m_t + \frac{1}{2}a\eta\Sigma u_t^2, \\ m_0 = \frac{1}{2}a(x_0 - b)^2. \end{aligned} \tag{64}$$

This is of the form (59) with $\Phi(m,u) = -2aum + a\eta\Sigma u^2/2$, $L(m,u) = 0$ and $G(m) = m$. Thus, we write the HJB equation

$$\partial_t V(m,t) + \min_{u\in[0,1]}\{\partial_m V(m,t)[-2aum + \frac{1}{2}a\eta\Sigma u^2]\} = 0,$$
$$V(m,T) = m. \tag{65}$$

First, it's not hard to see that for $a > 0$, $\partial_m V \geq 0$ for all m, t. This is because, the lower the m, the closer we are to the optimum and hence the minimum cost achievable in the same time interval $[t,T]$ should be less. Similarly,$\partial_m V \geq 0$ holds for $a < 0$ if one reverses all previous statements (in this case m is negative). Hence, we can calculate the minimum

$$\begin{aligned} u^* &= \arg\min_{u\in[0,1]}\{-2aum + \frac{1}{2}a\eta\Sigma u^2\}, \\ &= \begin{cases} 1 & a \leq 0, \\ \min(1, \frac{2m}{\eta\Sigma}) & a > 0. \end{cases} \end{aligned} \tag{66}$$

Notice that this solution is a feed-back control policy. We can now substitute $u_t = u_t^*$ where

$$u_t^* = \begin{cases} 1 & a \leq 0, \\ \min(1, \frac{2m_t}{\eta\Sigma}) & a > 0. \end{cases} \tag{67}$$

into the ODE in (64) to obtain

$$m_t^* = \begin{cases} m_0 e^{-2at} + \frac{1}{4}\eta\Sigma(1 - e^{-2at}) & a \leq 0 \text{ or } t < t^*, \\ \frac{\eta\Sigma}{2+2a(t-t^*)} & a > 0 \text{ and } t \geq t^*. \end{cases} \tag{68}$$

where

$$t^* = \frac{1}{2a}\log\left(\frac{4m_0}{\eta\Sigma} - 1\right) \tag{69}$$

And therefore, we get from (67) the effective annealing schedule

$$u_t^* = \begin{cases} 1 & a \leq 0 \text{ or } t \geq t^*, \\ \frac{1}{1+a(t-t^*)} & a > 0 \text{ and } t > t^*, \end{cases} \tag{70}$$

E.4 Solution of the momentum parameter control problem

We shall consider the case $a > 0$, since for $a \leq 0$ the optimal control is trivially $\mu_t = 1$. The momentum parameter control problem is

$$\begin{aligned} &\min_{\mu:[0,T]\to[0,1]} m_T \text{ subject to} \\ &\frac{d}{dt}m_t = \mathcal{R}\lambda(\mu_t)(m_t - m_\infty(\mu_t)), \\ &m_0 = \frac{1}{2}a(x_0 - b)^2, \end{aligned} \tag{71}$$

where

$$\lambda(\mu) = -\frac{(1-\mu) - \sqrt{(1-\mu)^2 - 4a\eta}}{\eta}, \qquad m_\infty(\mu) = \frac{\eta\Sigma}{4(1-\mu)}. \tag{72}$$

This is of the form (59) with $\Phi(m,\mu) = \mathcal{R}\lambda(\mu)(m - m_\infty(\mu))$, $L(m,u) = 0$ and $G(m) = m$. The HJB equation is

$$\partial_t V(m,t) + \min_{\mu\in[0,1]} \{\partial_m V(m,t)[\mathcal{R}\lambda(\mu)(m - m_\infty(\mu))]\} = 0,$$

$$V(m,T) = m. \tag{73}$$

Again, it is easy to see that $\partial_m V(m,t) \geq 0$ for all m, t and so

$$\mu^* = \underset{\mu\in[0,1]}{\text{arg min}} \{\mathcal{R}\lambda(\mu)(m - m_\infty(\mu))\} \tag{74}$$

This minimization problem has no closed form solution. However, observe that $\mathcal{R}\lambda(\mu) \leq 0$ and is minimized at $\mu = \mu_{\text{opt}} = \max(0, 1 - 2\sqrt{a\eta})$. Now, if $\mu > \mu_{\text{opt}}$, we have $\mathcal{R}\lambda(\mu) = -(1-\mu)/\eta$ and so $\mathcal{R}\lambda(\mu)(m - m_\infty(\mu))$ is increasing in μ for $\mu > \mu_{\text{opt}}$ (one can check this by differentiation and showing that the derivative is always positive). Hence, $\mu^* \leq \mu_{\text{opt}}$ and it is sufficient to consider $\mu \in [0, \mu_{\text{opt}}]$ in the minimization problem (74).

Next, observe that $m - m_\infty(\mu)$ is decreasing in μ and negative if

$$m < \frac{\eta\Sigma}{4(1-\mu)}. \tag{75}$$

or

$$\mu > 1 - \frac{\eta\Sigma}{4m}. \tag{76}$$

At the same time, $\mathcal{R}\lambda(\mu)$ is negative and decreasing for $\mu \in [0, \mu_{\text{opt}}]$. Thus, the product $\mathcal{R}\lambda(\mu)(m - m_\infty(\mu))$ is positive and increasing for $1 - \frac{\eta\Sigma}{4m} < \mu < \mu_{\text{opt}}$ and hence we must have

$$\mu^* \leq 1 - \frac{\eta\Sigma}{4m}. \tag{77}$$

Note that this is only a bound, but for small η, we can take this as an approximation of μ^*, so long as it is less than μ_{opt}. Hence, we arrive at

$$\mu_t^* = \begin{cases} 1 & a \leq 0, \\ \min(\mu_{\text{opt}}, \max(0, 1 - \frac{\eta\Sigma}{4m_t})) & a > 0. \end{cases} \tag{78}$$

One can of course follow the steps in Sec. E.3 to calculate m_t^* and hence μ_t^* in the form of an annealing schedule. We omit these calculations since they are not relevant to applications.

F Numerical experiments

In this section, we provide model and algorithmic details for the various numerical experiments considered in the main paper, as well as a brief description of the commonly applied adaptive learning rate methods that we compare the cSGD algorithm with.

F.1 Model details

In Sec. 4 from the main paper, we consider three separate models for two datasets.

M0: fully connected NN on MNIST

The first dataset we consider the MNIST dataset (LeCun et al., 1998), which involves computer recognition of 60000 28×28 pixel pictures of handwritten digits. We split it into 55000 training samples and 5000 test samples. Our inference model is a fully connected neural network with one hidden layer. For a input batch K of pixel data (flattened into a vector) $z \in \mathbb{R}^{784 \times K}$, we define the model

$$y = \text{softmax}(W_2 h_R(W_1 z + b_1) + b2), \tag{79}$$

where the activation function h_R is the commonly used Rectified Linear Unit (ReLU)

$$h_R(z)_{(ij)} = \max(z_{(ij)}, 0). \tag{80}$$

The first layer weights and biases are $W_1 \in \mathbb{R}^{784 \times 10}$ and $b_1 \in \mathbb{R}^{10}$ and the second layer weights and biases are $W_2 \in \mathbb{R}^{10 \times 10}$ and $b_2 \in \mathbb{R}^{10}$. These constitute the trainable parameters. The softmax function is defined as

$$\text{softmax}(z)_{(ij)} = \frac{\exp(-z_{(ij)})}{\sum_k \exp(-z_{(kj)})}. \tag{81}$$

The output tensors $y \in \mathbb{R}^{10 \times K}$ is compared to a batch of one-hot target labels $\hat{y}$ with the cross-entropy loss

$$C(y, \hat{y}) = -\frac{1}{10K} \sum_{i,j} \hat{y}_{(ij)} \log y_{(ij)}. \tag{82}$$

Lastly, we use ℓ_2 regularization so that the minimization problem is

$$\min_{W_1, b_1, W_2, b_2} C(y, \hat{y}) + \sum_{i=1}^{2} \lambda_{W,i} \|W_i\|_2^2 + \sum_{i=1}^{2} \lambda_{b,i} \|b_i\|_2^2, \tag{83}$$

Each regularization strength λ is set to be 1 divided by the dimension of the trainable parameter.

C0: fully connected NN on CIFAR-10

The CIFAR-10 dataset (Krizhevsky & Hinton, 2009) consists of 60000 small 32×32 pixels of RGB natural images belonging to ten separate classes. We split the dataset into 50000 training samples and 10000 test samples. Our first model for this dataset is a deeper fully connected neural network

$$y = \text{softmax}(W_3 h_T(W_2 h_T(W_1 z + b_1) + b_2) + b_3), \tag{84}$$

where we use a tanh activation function between the hidden layers

$$h_T(z)_{(ij)} = \text{tanh}(z_{(ij)}). \tag{85}$$

The layers have width 3071,500,300,10. That is, the trainable parameters have dimensions $W_1 \in \mathbb{R}^{3071\times 500}, b_1 \in \mathbb{R}^{500}$, $W_2 \in \mathbb{R}^{500\times 300}, b_2 \in \mathbb{R}^{300}$, $W_3 \in \mathbb{R}^{300\times 10}, b_3 \in \mathbb{R}^{10}$. We use the same soft-max output, cross-entropy loss and ℓ_2 regularization as as before.

C1: convolutional NN on CIFAR-10

Our last experiment is a convolutional neural network on the same CIFAR-10 dataset. We use four convolution layers consisting of convolution,batch-normalization,ReLU,max-pooling. Convolution filter size is 5×5, with uniform stride 1 and padding 2. Output channels of convolution layers are {96,128,256,64}. The pooling size is 2×2 with stride 2. The output layers consist of two fully connected layers of width {1024,256} and drop-out rate 0.5. ℓ_2 regularization is introduced as a weight decay with decay parameter 5e-3.

F.2 Adagrad and Adam

Here, we write down for completeness the iteration rules of Adagrad (Duchi et al., 2011), and Adam (Kingma & Ba, 2015) optimizers, which are commonly applied tools to tune the learning rate. For more details and background, the reader should consult the respective references.

Adagrad. The Adagrad modification to the SGD reads

$$x_{(i),k+1} = x_{k,(i)} - \frac{\eta}{\sqrt{G_{k,(i)}}}\partial_{(i)} f_{\gamma_k}(x_k), \tag{86}$$

where $G_{k,(i)}$ is the running sum of gradients $\partial_{(i)} f_{\gamma_l}(x_l)$ for $l = 0, \dots, k-1$. The tunable hyper-parameters are the learning rate η and the initial accumulator value G_0. In this paper we consider only the learning rate hyper-parameter as this is equivalent to setting the initial accumulator to a common constant across all dimensions.

Adam. The Adam method has similar ideas to momentum. It keeps the exponential moving averages

$$\begin{aligned} m_{(i),k+1} &= \beta_1 m_{k,(i)} + (1-\beta_1)\partial_{(i)} f_{\gamma_k}(x_k), \\ v_{(i),k+1} &= \beta_2 v_{k,(i)} + (1-\beta_2)[\partial_{(i)} f_{\gamma_k}(x_k)]^2. \end{aligned} \tag{87}$$

Next, set,

$$\begin{aligned} \hat{m}_{k,(i)} &= \frac{m_{k,(i)}}{1-\beta_1^k}, \\ \hat{v}_{k,(i)} &= \frac{v_{k,(i)}}{1-\beta_2^k}. \end{aligned} \tag{88}$$

Finally, the Adam update is

$$x_{(i),k+1} = x_{k,(i)} - \frac{\eta}{\sqrt{\hat{v}_{k,(i)}}} \hat{m}_{k,(i)}. \tag{89}$$

The hyper-parameters are the learning rate η and the EMA decay parameters β_1, β_2.

Note that for both methods above, one can also introduce a regularization term ϵ to the denominator to prevent numerical instabilities.

F.3 Implementation of cSGD

Recall from Sec. 4.1 that the optimal control solution for learning rate control of the quadratic objective $f(x) = \frac{1}{2}a(x-b)^2$ is given by

$$u_t^* = \begin{cases} 1 & a \leq 0, \\ \min(1, \frac{2m_t}{\eta\Sigma}) & a > 0. \end{cases} \tag{90}$$

The idea is to perform a local quadratic approximation

$$f(x) \approx \frac{1}{2}\sum_{i=1}^{d} a_{(i)}(x_{(i)} - b_{(i)})^2. \tag{91}$$

This is equivalent to a local linear approximation of the gradient, i.e. for $i = 1, 2, \ldots, d$

$$\partial_{(i)} f(x) \approx a_{(i)}(x_{(i)} - b_{(i)}). \tag{92}$$

This effectively decouples the control problems of d identical one-dimensional control problems, so that we may apply (90) element-wise. We note that this approximation is only assumed to hold locally and the parameters must be updated. There are many ways to do this. Our approach uses linear regression on-the-fly via exponential moving averages (EMA). For each trainable dimension i, we maintain the following exponential averages

$$\begin{aligned}
\overline{g}_{k+1,(i)} &= \beta_{k,(i)}\overline{g}_{k,(i)} + (1-\beta_{k,(i)})f'_{\gamma_k}(x_{k,(i)}), \\
\overline{g^2}_{k+1,(i)} &= \beta_{k,(i)}\overline{g^2}_{k,(i)} + (1-\beta_{k,(i)})f'_{\gamma_k}(x_{k,(i)})^2, \\
\overline{x}_{k+1,(i)} &= \beta_{k,(i)}\overline{x}_{k,(i)} + (1-\beta_{k,(i)})x_{k,(i)}, \\
\overline{x^2}_{k+1,(i)} &= \beta_{k,(i)}\overline{x^2}_{k,(i)} + (1-\beta_{k,(i)})x^2_{k,(i)}, \\
\overline{gx}_{k+1,(i)} &= \beta_{k,(i)}\overline{gx}_{k,(i)} + (1-\beta_{k,(i)})x_{k,(i)}f'_{\gamma_k}(x_{k,(i)}).
\end{aligned} \tag{93}$$

The decay parameter $\beta_{k,(i)}$ controls the effective averaging window size. In practice, we should adjust $\beta_{k,(i)}$ so that it is small when variations are large, and vice versa. This ensures that our local approximations adapts to the changing landscapes. Since local variations is related to the gradient, we use the following heuristic

$$\beta_{k+1,(i)} = \beta_{\text{min}} + (\beta_{\text{max}} - \beta_{\text{min}})\frac{\overline{g^2}_{k,(i)} - \overline{g}^2_{k,(i)}}{\overline{g^2}_{k,(i)}}. \tag{94}$$

Algorithm 1 controlled SGD (cSGD)

```
**Hyper-parameters:** $\eta$, $u_0$
Initialize $x_0$; $\beta_{0,(i)} = 0.9 \,\forall i$
**for** $k = 0$ **to** $(\#\text{iterations} - 1)$ **do**
    Compute sample gradient $\nabla f_{\gamma_k}(x_k)$
    **for** $i = 1$ **to** $d$ **do**
        Update EMA using (93)
        Compute $a_{k,(i)}$, $b_{k,(i)}$, $\Sigma_{k,(i)}$ using (95)
        Compute $u^*_{k,(i)}$ using (96)
        $\beta_{k+1,(i)} = (\overline{g^2}_{k,(i)} - \overline{g}^2_{k,(i)})/\overline{g^2}_{k,(i)}$ and clip
        $u_{k+1,(i)} = \beta_{k,(i)} u_{k,(i)} + (1 - \beta_{k,(i)}) u^*_{k,(i)}$
        $x_{k+1,(i)} = x_{k,(i)} - \eta u_{k,(i)} \nabla f_{\gamma_k}(x_k)_{(i)}$
    **end for**
**end for**
```

which is similar to the one employed in Schaul et al. (2013) for maintaining EMAs. The additional clipping to the range $[\beta_{\text{min}}, \beta_{\text{max}}]$ is to make sure that there are enough samples to calculate meaningful regressions, and at the same time prevent too large decay values where the contribution of new samples vanish. In the applications presented in this paper, we usually set $\beta_{\text{min}} = 0.9$ and $\beta_{\text{max}} = 0.999$, but results are generally insensitive to these values.

With the EMAs (93), we compute $a_{k,(i)}$ by the ordinary-least-squares formula and $\Sigma_{k,(i)}$ as the variance of the gradients:

$$\begin{aligned} a_{k,(i)} &= \frac{\overline{gx}_{k,(i)} - \overline{g}_{k,(i)}\overline{x}_{k,(i)}}{\overline{x^2}_{k,(i)} - \overline{x}^2_{k,(i)}}, \\ b_{k,(i)} &= \overline{x}_{k,(i)} - \frac{\overline{g}_{k,(i)}}{a_{k,(i)}}, \\ \Sigma_{k,(i)} &= \overline{g^2}_{k,(i)} - \overline{g}^2_{k,(i)}, \end{aligned} \tag{95}$$

This allows us to estimate the policy (90) as

$$u^*_{k,(i)} = \begin{cases} 1 & a_{k,(i)} \leq 0, \\ \min(1, \frac{a_{k,(i)}(\overline{x}_{k,(i)} - b_{k,(i)})^2}{\eta \Sigma_{k,(i)}}) & a_{k,(i)} > 0. \end{cases} \tag{96}$$

for $i = 1, 2, \ldots, d$. Since our averages are from exponentially averaged sources, we should also update our learning rate policy in the same way:

$$u_{k+1,(i)} = \beta_{k,(i)} u_{k,(i)} + (1 - \beta_{k,(i)}) u^*_{k,(i)} \tag{97}$$

The algorithm is summarized in Alg. 1

Algorithm 2 controlled momentum SGD (cMSGD)

```
**Hyper-parameters:** $\eta$, $\mu_0$
Initialize $x_0$, $v_0$; $\beta_{0,(i)} = 0.9 \; \forall i$
**for** $k = 0$ **to** $(\#\text{iterations} - 1)$ **do**
    Compute sample gradient $\nabla f_{\gamma_k}(x_k)$
    **for** $i = 1$ **to** $d$ **do**
        Update EMA using (93)
        Compute $a_{k,(i)}$, $b_{k,(i)}$, $\Sigma_{k,(i)}$ using (95)
        Compute $\mu^*_{k,(i)}$ using (99)
        $\beta_{k+1,(i)} = (\overline{g^2}_{k,(i)} - \overline{g}^2_{k,(i)})/\overline{g^2}_{k,(i)}$ and clip
        $\mu_{k+1,(i)} = \beta_{k,(i)}\mu_{k,(i)} + (1 - \beta_{k,(i)})\mu^*_{k,(i)}$
        $v_{k+1,(i)} = \mu_{k,(i)} v_{k,(i)} - \eta \nabla f_{\gamma_k}(x_k)_{(i)}$
        $x_{k+1,(i)} = x_{k,(i)} + v_{k+1,(i)}$
    **end for**
**end for**
```

F.4 Implementation of cMSGD

We wish to apply the momentum parameter control

$$\mu^*_t = \begin{cases} 1 & a \leq 0, \\ \min(\mu_{\text{opt}}, \max(0, 1 - \frac{\eta\Sigma}{4m_t})) & a > 0, \end{cases} \tag{98}$$

where $\mu_{\text{opt}} = \max\{0, 1 - 2\sqrt{a\eta}\}$. We proceed in the same way as in Sec. F.3 by keeping the relevant EMA averages and performing linear regression on the fly. The only difference is the application of the momentum parameter adjustment, which is

$$\mu^*_{k,(i)} = \begin{cases} 1 & a_{k,(i)} \leq 0, \\ \min[\max(0, 1 - 2\sqrt{a_{k,(i)}\eta}), & \\ \max(0, 1 - \frac{\eta\Sigma_{k,(i)}}{2a_{k,(i)}(x_{k,(i)} - b_{k,(i)})^2})] & a_{k,(i)} > 0, \end{cases} \tag{99}$$

The algorithm is summarized in Alg. 2.

F.5 Training accuracy for C1

For completeness we also provide in Fig. 2 the training accuracies of C1 with various hyper-parameter choices and methods tested in this work. These complements the plots of test accuracies in Fig. 3,5,6 in the main paper. We see that cSGD and cMSGD display the same robustness in terms of test and training accuracies.

References

Bellman, Richard. Dynamic programming and Lagrange multipliers. *Proceedings of the National Academy of Sciences*, 42(10):767–769, 1956.

Courant, Richard, Isaacson, Eugene, and Rees, Mina. On the solution of nonlinear hyperbolic differential equations by finite differences. *Communications on Pure and Applied Mathematics*, 5(3):243–255, 1952.

Daly, Bart J. The stability properties of a coupled pair of non-linear partial difference equations. *Mathematics of Computation*, 17(84):346–360, 1963.

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

Freidlin, Mark I, Szücs, Joseph, and Wentzell, Alexander D. *Random perturbations of dynamical systems*, volume 260. Springer Science & Business Media, 2012.

Hille, Einar and Phillips, Ralph Saul. *Functional analysis and semi-groups*, volume 31. American Mathematical Soc., 1996.

Hirt, CW. Heuristic stability theory for finite-difference equations. *Journal of Computational Physics*, 2(4):339–355, 1968.

Kingma, Diederik and Ba, Jimmy. Adam: A method for stochastic optimization. *ICLR*, 2015.

Kloeden, P. E. and Platen, E. *Numerical Solution of Stochastic Differential Equations*. Springer, New York, corrected edition, June 2011.

Krizhevsky, Alex and Hinton, Geoffrey. Learning multiple layers of features from tiny images. 2009.

Lax, Peter and Wendroff, Burton. Systems of conservation laws. *Communications on Pure and Applied mathematics*, 13(2):217–237, 1960.

LeCun, Yann, Cortes, Corinna, and Burges, Christopher JC. The mnist dataset of handwritten digits. *URL http://yann. lecun. com/exdb/mnist*, 1998.

LeVeque, Randall J. *Finite volume methods for hyperbolic problems*, volume 31. Cambridge university press, 2002.

Liberzon, Daniel. *Calculus of variations and optimal control theory: a concise introduction*. Princeton University Press, 2012.

Milstein, GN. Approximate integration of stochastic differential equations. *Theory of Probability & Its Applications*, 19(3):557–562, 1975.

Milstein, GN. A method of second-order accuracy integration of stochastic differential equations. *Theory of Probability & Its Applications*, 23(2):396–401, 1979.

Milstein, GN. Weak approximation of solutions of systems of stochastic differential equations. *Theory of Probability & Its Applications*, 30(4):750–766, 1986.

Milstein, GN. *Numerical integration of stochastic differential equations*, volume 313. Springer Science & Business Media, 1995.

Noh, WF and Protter, MH. Difference methods and the equations of hydrodynamics. Technical report, California. Univ., Livermore. Lawrence Radiation Lab., 1960.

Oksendal, Bernt. *Stochastic differential equations: an introduction with applications*. Springer Science & Business Media, 2013.

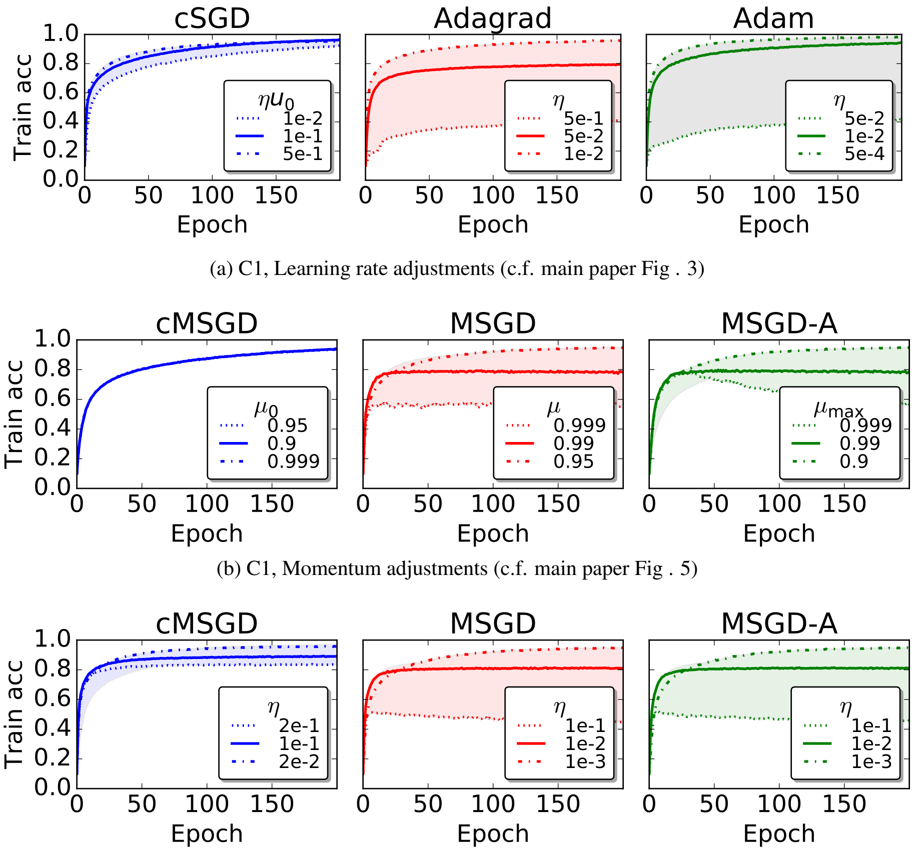
Pontryagin, Lev Semenovich. *Mathematical theory of optimal processes*. CRC Press, 1987.

Schaul, Tom, Zhang, Sixin, and LeCun, Yann. No more pesky learning rates. In *ICML (3)*, volume 28, pp. 343–351, 2013.

Talay, Denis and Tubaro, Luciano. Expansion of the global error for numerical schemes solving stochastic differential equations. *Stochastic analysis and applications*, 8(4): 483–509, 1990.

Uhlenbeck, George E and Ornstein, Leonard S. On the theory of the Brownian motion. *Physical review*, 36(5):823, 1930.

Warming, RF and Hyett, BJ. The modified equation approach to the stability and accuracy analysis of finite-difference methods. *Journal of computational physics*, 14(2):159–179, 1974.



(a) C1, Learning rate adjustments (c.f. main paper Fig . 3)

(b) C1, Momentum adjustments (c.f. main paper Fig . 5)

(c) C1, Learning rate sensitivity (c.f. main paper Fig . 6)

Figure 2: Training accuracies for various methods and hyper-parameter choices. The set-up is the same as in the main paper, Fig. 3,5,6 except that we plot training accuracy instead of test accuracy. The qualitative observation is the same: cSGD and cMSGD are generally robust to changing parameters and models.