Journal of Computational Dynamics
Vol. 11, No. 4, October 2024, pp. 453-467
doi:10.3934/jcd.2024008

BACKWARD ERROR ANALYSIS AND THE QUALITATIVE BEHAVIOUR OF STOCHASTIC OPTIMIZATION ALGORITHMS: APPLICATION TO STOCHASTIC COORDINATE DESCENT

Stefano Di Giovacchino[1], Desmond J. Higham[*2,3] and Konstantinos C. Zygalakis[2,3]

[1]Department of Information Engineering and Computer Science and Mathematics, University of L'Aquila, Via Vetoio, Loc. Coppito - 67100 L'Aquila, Italy

[2]School of Mathematics, University of Edinburgh, Edinburgh, Scotland, UK

[3]Maxwell Institute for Mathematical Sciences, Bayes Centre, 47 Potterrow, Edinburgh, Scotland, UK

Abstract. Stochastic optimization methods have been hugely successful in making large-scale optimization problems feasible when computing the full gradient is computationally prohibitive. Using the theory of modified equations for numerical integrators, we propose a class of stochastic differential equations that approximate the dynamics of general stochastic optimization methods more closely than the original gradient flow. Analyzing a modified stochastic differential equation can reveal qualitative insights about the associated optimization method. Here, we study mean-square stability of the modified equation in the case of stochastic coordinate descent.

1. **Introduction.** The connection between optimization and numerical timestepping has been exploited by several researchers, leading to new insights in both fields; see, for example, [11, 12, 16, 26]. In this work, we focus on stochastic optimization algorithms and their connection with numerical methods for stochastic ordinary differential equations (SDEs) by using a backward error, or modified equation, approach. Backward error analysis is a technique that was initially developed to explain the behaviour of deterministic algorithms in polynomial root finding and linear algebra [14, 30] as well as the long-time integration of Hamiltonian dynamics [10, 25]. In the last few years there has been a revised interest in applications of backward error analysis for studying stochastic algorithms, starting with the works in [27, 32] for analysing integrators for SDEs with subsequent applications to the analysis of stochastic optimization methods [19], as well as a range of phenomena in machine learning [1, 21, 28]. Here, the aim is to derive a new SDE that accurately

2020 *Mathematics Subject Classification.* 60H35, 65K10.

Key words and phrases. Gradient descent, mean square stability, modified equation, strong convexity, weak approximation.

The first author is member of the INDAM Research group GNCS and has been supported by PRIN2017-MIUR project. KCZ acknowledges support from the EPSRC grant EP/V006177/1.

* Corresponding author: Desmond J. Higham.

describes the dynamics of the timestepping method, and hence reveals qualitative properties of the optimization method.

The main contributions of this work are

- Proposition 4.2: a modified SDE for a general class of stochastic optimization iterations derived by following the framework introduced in [32] and generalizing the results of [19].
- Theorem 4.6: conditions that guarantee mean-square stable convergence to the minimizer for the stochastic coordinate descent case.

The rest of the paper is organized as follows. In Section 2 we introduce the idea of modified equations for numerical integrators, in both the ODE and SDE settings. We include examples of their application to deterministic Hamiltonian dynamics, and to the Ornstein-Uhlenbeck process. In Section 3 we discuss the main idea behind stochastic optimization methods and focus on two cases; stochastic gradient descent and stochastic coordinate descent. Section 4 presents our main results: a modified SDE for a general stochastic optimization iteration and mean-square stability conditions in the case of stochastic coordinate descent. Finally, we conclude in Section 5 with a brief discussion of possible avenues for future research.

2. Preliminaries.

2.1. Backward Error analysis and Ordinary Differential equations.

Consider a smooth ODE

$$\frac{dX}{dt} = f(X), \quad X(0) = x, \tag{1}$$

where $f : \mathbb{R}^d \mapsto \mathbb{R}^d$. Suppose we apply a numerical method with stepsize h to produce an approximation $x_n \approx X(nh)$ at time $T = nh$. The objective of backward error analysis [3, 6, 8, 9, 10] in this setting is to find an ODE

$$\frac{d\tilde{X}}{dt} = f_h(\tilde{X}), \quad \tilde{X}(0) = x, \tag{2}$$

that is closer than (1) to the numerical approximation. Typically $f_h(\tilde{X})$ has an expansion

$$f_h(\tilde{X}) = f(\tilde{X}) + hf_1(\tilde{X}) + h^2 f_2(\tilde{X}) + \cdots, \tag{3}$$

for some appropriately chosen functions f_i that are determined by the numerical method. We note that the ODE (2) depends on h and is often referred to as a modified equation. Backward error analysis has proved to be a useful technique since in many contexts the modified equation (2) is more amenable to analysis than the underlying numerical method.

The main tool for finding the terms f_i in (3) is the Taylor expansion. We denote by $\phi_h(x)$ the true solution of (1) at time h starting from x and by $\psi_h(x)$ the corresponding numerical solution starting from x after one step of length h. We know [15] that the local error for a numerical method of order p satisfies

$$\phi_h(x) - \psi_h(x) = \mathcal{O}(h^{p+1}).$$

Hence in order to find an ODE that approximates the numerical method more closely we ask for

$$\tilde{\phi}_h(x) - \psi_h(x) = \mathcal{O}(h^{p+s}),$$

where $\tilde{\phi}_h(x)$ denotes the true solution of our new ODE after time h starting from x, and $s > 0$. Increasing the value of s gives an ODE that approximates more closely the dynamics of our numerical method. It may be argued that $s = 1$ is the most

important case as it characterises the first non-zero term in (3), and hence, for small h and finite times, captures the main deviation from the original ODE.

2.1.1. *An application to Hamiltonian dynamics.* We now discuss an example of backward error analysis in the case of a Hamiltonian problem. What we describe applies also to non-quadratic Hamiltonians; however, for the purpose of illustration, we present everything in the quadratic setting as this allows us to calculate the exact solution of both the original and modified equations. We consider the ODE

$$\frac{dp}{dt} = q, \quad \frac{dq}{dt} = -p, \tag{4}$$

where p and q are scalar. Here, the Hamiltonian function $H(p,q) = \frac{1}{2}p^2 + \frac{1}{2}q^2$ remains constant for all time. We will study the Euler method

$$p_{n+1} = p_n + hq_n, \quad q_{n+1} = q_n - hp_n, \tag{5}$$

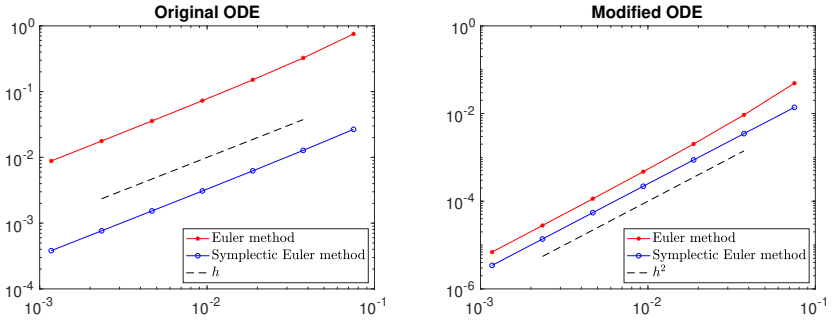
and the symplectic Euler method

$$q_{n+1} = q_n - hp_n \quad p_{n+1} = p_n + hq_{n+1}. \tag{6}$$

We will look at the finite time global error

$$e_h(T) := \left\| \begin{bmatrix} p(T) \\ q(T) \end{bmatrix} - \begin{bmatrix} p_n \\ q_n \end{bmatrix} \right\| \tag{7}$$

for $T = nh$. Here, and throughout, $\|\cdot\|$ denotes the Euclidean norm. As we can see in Figure 1a, by plotting $e_h(T)$ versus h, both numerical methods are first order[1]. Furthermore, we observe that the error of the symplectic Euler method is much smaller than that of the standard Euler method. This is further illustrated



(A) $e_h(T)$ vs h (B) $e_h(T)$ vs h

FIGURE 1. (a) Finite time global error $e_h(T)$ versus h where $(p(T), q(T))$ is the exact solution of (4). (b) Finite time global error $e_h(T)$ versus h where $(p(T), q(T))$ is the exact solution of the modified equation (8) in the case of Euler method and (9) in the case of the symplectic Euler method.

in Figure 2 where we plot numerical trajectories given by the Euler and symplectic Euler methods using a stepsize $h = 0.0375$ up to time $T = 15$. We see that the Euler method fails to reflect the conservative nature of the true solution, unlike the symplectic Euler method.

[1]Here we have chosen $T = 15, p(0) = 1, q(0) = 0$.

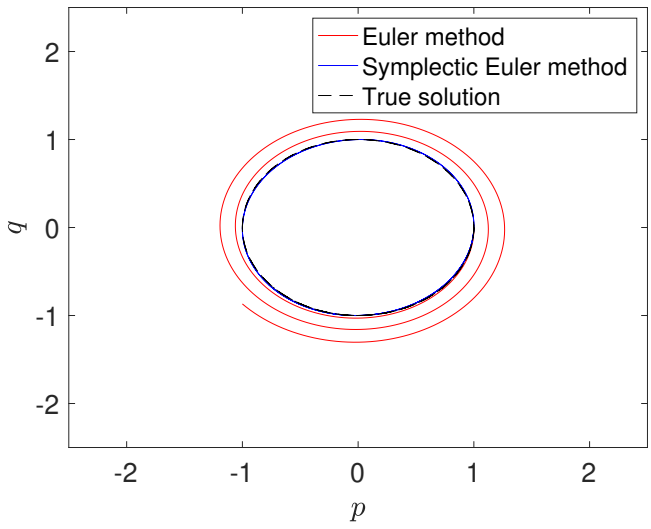


FIGURE 2. Comparison of the numerical solution using Euler and symplectic Euler method for $h = 0.0375$ against the exact solution of (4). Here we have taken $T = 15$.

A modified equation associated with the Euler method is

$$\frac{d\tilde{p}}{dt} = \tilde{q} + \frac{h}{2}\tilde{p}, \quad \frac{d\tilde{q}}{dt} = -\tilde{p} + \frac{h}{2}\tilde{q}, \tag{8}$$

while a modified equation associated with the symplectic Euler method is

$$\frac{d\tilde{p}}{dt} = \tilde{q} - \frac{h}{2}\tilde{p}, \quad \frac{d\tilde{q}}{dt} = -\tilde{p} + \frac{h}{2}\tilde{q}; \tag{9}$$

see [8, 10, 25] for derivations. In Figure 1b we plot the global error $e_h(T)$ as a function of the stepsize h, with the true solution based on (8) for the Euler method and (9) for the symplectic Euler method. The results are consistent with what is predicted by the theory since the numerical methods approximate the original equation (4) to order 1 and their corresponding modified equation to order 2.

A closer look at (8) and (9) can shed some light on the discrepancy between the two numerical solutions observed in Figure 2. Recall that in the original problem (4) the Hamiltonian function $H(p, q) = \frac{1}{2}p^2 + \frac{1}{2}q^2$ is preserved. The rate of change of $H(\tilde{p}(t), \tilde{q}(t))$ for the modified equation of the Euler method satisfies

$$\frac{dH}{dt} = \frac{1}{2}hH,$$

implying that no matter how small the stepsize h is taken, over long periods of integration time there will be exponential growth in the values of the Hamiltonian; thus explaining the outward spiraling observed for the Euler method in Figure 2. Repeating this calculation for the modified equation associated with the symplectic Euler method we find a rate of change of H equal to zero; implying that up to order h^2 symplectic Euler conserves the true Hamiltonian and explaining the very small differences between the true solution and the symplectic Euler method observed in Figures 1a and 2.

This example gives an indication of the effectiveness of backward error analysis. We note that a textbook Gronwall-based numerical analysis would not differentiate between Euler and symplectic Euler, viewing them both as first-order methods. However, the corresponding modified equations help us to understand their qualitative differences. We refer to [3, 6, 9, 10] for much more detail on the insights that backward error analysis can provide.

2.2. **Backward error analysis and Stochastic Differential equations.** Consider the SDE

$$dX = f(X)dt + g(X)dW, \quad X(0) = x, \tag{10}$$

where $f : \mathbb{R}^d \mapsto \mathbb{R}^d, g : \mathbb{R}^{d\times m} \mapsto \mathbb{R}^d$, $x \in \mathbb{R}^d$ is the initial condition, and $W(t)$ is the standard m-dimensional Brownian motion. In addition consider the numerical approximation of (10) by a one-step numerical integrator at time $t = nh$ of the form

$$X_{n+1} = \Psi(X_n, h, \xi_n), \tag{11}$$

where h denotes the stepsize and ξ_n are independent random vectors. In choosing an appropriate method (11) to approximate (10), we must first decide how error is to be measured. For the approximation of individual trajectories one is interested in strong convergence, while for the approximation of the expectation of functionals of the solution, one is interested in weak convergence [13, 17, 20].

Following on from the ODE case, the objective of backward error analysis is to find an SDE of the form

$$d\tilde{X} = f_h(\tilde{X})dt + g_h(\tilde{X})dW \tag{12}$$

that the numerical method (11) approximates more closely in a weak or strong sense. While there has been some work on trying to specify f_h and g_h in the case of strong convergence, the majority of the backward error analysis literature for SDEs concentrates on weak convergence, and this is what we focus on here. Denote the expectation of functionals $\phi(\cdot)$ of the solution of (10) after one step by

$$u(x, h) = \mathbb{E}(\phi(X(h))|X(0) = x) \tag{13}$$

and the corresponding numerical one-step expectation by

$$U(x, h) = \mathbb{E}(\phi(X_1)|X_0 = x). \tag{14}$$

If the numerical method is of weak order p [17, 27] then the local error satisfies

$$u(x, h) - U(x, h) = \mathcal{O}(h^{p+1}).$$

Hence in order to find an SDE of the form (12) that approximates the numerical method more closely, we require

$$\tilde{u}(x, h) - U(x, h) = \mathcal{O}(h^{p+s}),$$

where $\tilde{u}(x, h)$ denotes the one-step expectation of the modified SDE and $s > 0$. As with ODEs, the most important value of s is $s = 1$, but, unlike the ODE setting, there is not always a solution even for this value of s [32].

2.2.1. *First modified equation (the case $s = 1$).* As discussed above we may derive modified equations for a stochastic numerical integrator by analyzing the one-step weak local error. We will do this by using an expansion in terms of the stepsize h. In particular, the following expansion holds for $u(x, h)$

$$u(x,h) = \phi(x) + h\mathcal{L}\phi + \frac{h^2}{2}\mathcal{L}^2\phi + \cdots \tag{15}$$

where

$$\mathcal{L} := f(x)\cdot\nabla\phi + \frac{1}{2}gg^T : \nabla\nabla\phi$$

is the generator of the Markov process associated with the solution of (10). Here $A : B$ denotes the Frobenius inner product between two matrices. For simplicity of presentation, we have not specified conditions on f and g that allow (15) to be made rigorous; these can be found in [29, 4]. We will assume in addition that $U(x, h)$ admits the expansion

$$U(x,h) = \phi(x) + h\mathcal{A}_0\phi + h^2\mathcal{A}_1\phi + \cdots,$$

where $\mathcal{A}_0, \mathcal{A}_1$ are partial differential operators that depend on higher order derivatives of f and g. A necessary condition for the numerical integrator to be of weak order 1 is for the weak local error to be of order 2, which implies $\mathcal{A}_0 = \mathcal{L}$. We now want to find an SDE that our numerical integrator approximates with weak order 2. Writing (12) as

$$d\tilde{X} = (f(\tilde{X}) + hf_1(\tilde{X}))dt + (g(\tilde{X}) + hg_1(\tilde{X}))dW \tag{16}$$

and expanding $\tilde{u}(x, h)$, it may be shown [32] that the condition

$$\mathcal{L}_1\phi = \mathcal{A}_1\phi - \frac{1}{2}\mathcal{L}^2\phi, \quad \text{where} \quad \mathcal{L}_1 := f_1\cdot\nabla + \frac{1}{2}\left(gg_1^T + g_1g^T\right) : \nabla\nabla,$$

ensures that $\tilde{u}(x, h) - U(x, h) = \mathcal{O}(h^3)$, which in turn implies that the weak error between (16) and the numerical integrator is of order 2.

2.2.2. *Application to Ornstein–Uhlenbeck process.* We now discuss a specific application of backward error analysis in the case of the one dimensional Ornstein–Uhlenbeck process

$$dX = -\gamma X dt + \sigma dW. \tag{17}$$

We consider two different numerical methods: Euler–Maruyama

$$X_{n+1} = X_n - \gamma h X_n + \sigma\sqrt{h}\,\xi_n, \tag{18}$$

and implicit Euler

$$X_{n+1} = X_n - \gamma h X_{n+1} + \sigma\sqrt{h}\,\xi_n. \tag{19}$$

In both cases the ξ_n are i.i.d. standard Gaussians. In Figure 3a we plot

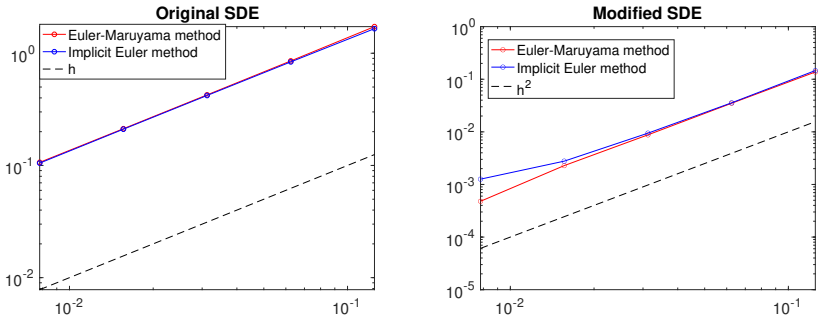
$$e_h(T,\phi) = |\mathbb{E}(\phi(X(T))) - \mathbb{E}(\phi(X_n))|,$$

for $T = nh$ and $\phi(x) = x^2$. As we can see[2]. both numerical methods are first order in the weak sense.

The associated first modified equations are

$$d\tilde{X} = -\left(\gamma + \frac{h}{2}\gamma^2\right)\tilde{X} + \sigma\left(1 + \frac{\gamma h}{2}\right)dW, \tag{20}$$

[2]Here we have used $T = 1, \gamma = 1, \sigma = 10^{-1}$ and $X(0) = 10$



(A) $e_h(T,\phi)$ vs h (B) $e_h(T,\phi)$ vs h

FIGURE 3. (a) Error $e_h(T,\phi)$ versus h, where $X(t)$ is the exact solution of (17) (b) Error $e_h(T,\phi)$ versus h, where $(p(T),q(T))$ is the exact solution of (20) (Euler method) or (21) (symplectic Euler method).

for Euler–Maruyama and

$$d\tilde{X} = -\left(\gamma - \frac{h}{2}\gamma^2\right)\tilde{X} + \sigma\left(1 + \frac{\gamma h}{2}\right)dW, \tag{21}$$

for implicit Euler, [32]. In Figure 3b we plot again $e_h(T,\phi)$ as a function of the stepsize h, but to form the true solution we use (20) for the Euler-Maruyama method and (21) for the implicit Euler method. The results are consistent with weak order equal to 2.

3. **Stochastic optimization algorithms.** Given an objective function $F:\mathbb{R}^d \mapsto \mathbb{R}$, we now consider the unconstrained optimization problem

$$\min_{x\in\mathbb{R}^d} F(x).$$

A natural iteration in the case where F is differentiable is gradient descent

$$x_{n+1} = x_n - h\nabla F(x_n), \tag{22}$$

which can be seen as an Euler discretization of the ODE

$$\frac{dX}{dt} = -\nabla F(X). \tag{23}$$

In many applications of interest calculating the full gradient of F can be computationally prohibitive, so in (22) one replaces $\nabla F(x)$ with an unbiased estimator $\widehat{\nabla} F(x,w)$, where w is an appropriately chosen random variable such that $\mathbb{E}(\widehat{\nabla} F(x,w)) = \nabla F(x)$. In this case (22) becomes

$$x_{n+1} = x_n - h\,\widehat{\nabla} F(x_n,w). \tag{24}$$

This framework covers a wide class of stochastic optimization algorithms [31]. We will focus here on stochastic gradient descent and stochastic coordinate descent.

3.1. **Stochastic gradient descent.** In most machine learning applications, the function F is of the additive form

$$F(X) := \frac{1}{N}\sum_{i=1}^{N} F_i(X), \tag{25}$$

where $F, F_i : \mathbb{R}^d \mapsto \mathbb{R},\ i = 1, \dots, N$, with N representing, for example, the number of training points in a supervised learning task. Hence calculating the gradient of F involves a summation of N terms, which can be expensive when N is large. A typical estimator of ∇F in this case is

$$\widehat{\nabla} F(x, w) = \frac{m}{N}\sum_{i=1}^{m} \nabla F_{w_i}(x), \tag{26}$$

where $w = (w_1, \cdots, w_m)$ is a random subset of $[N] = \{1, \cdots, N\}$ of size m, generated for example by sampling with or without replacement from $[N]$. Intuitively, this approach replaces the full training set by a randomly chosen subset, using a different subset on each iteration. In this case it is not difficult to show that $\widehat{\nabla} F(x, w)$ is indeed an unbiased estimator of ∇F. Using (26) we have the iteration

$$x_{n+1} = x_n - \frac{h\,m}{N}\sum_{i=1}^{m} \nabla F_{w_i}(x_n), \tag{27}$$

which is known as stochastic gradient descent [24]. To analyze the behaviour of this algorithm, assumptions must be placed on the functions F_i. In certain cases, it is possible to show that with a suitable iteration-dependent diminishing step size sequence, $h = h_n$, the algorithm will converge to a minimizer of F [24, 2, 7, 31]. Nevertheless, it is also of interest to understand how the algorithm behaves for fixed step size h, and there is indeed a wide range of literature investigating this question; see, for example, [5, 18, 19].

3.2. **Stochastic coordinate descent and variants.** Another case of interest in large-scale optimization arises when the dimension d is very large. Since

$$\nabla F = \sum_{i=1}^{d} \frac{\partial F}{\partial x_i} e_i := \sum_{i=1}^{d} (\nabla_i F) e_i, \tag{28}$$

where e_i are the standard orthonormal vectors in $\mathbb{R}^d$, calculating the gradient in all directions when d is large can be expensive. A widely used estimator of ∇F in this case is

$$\widehat{\nabla} F(x, w) = \frac{d}{m}\sum_{i=1}^{m} (\nabla_{w_i} F(x)) e_{w_i}, \tag{29}$$

where $w = (w_1, \cdots, w_m)$ is a random subset of $[d] = \{1, \cdots, d\}$ of size m generated for example by sampling with or without replacement from $[d]$. Intuitively, we are using gradient information from a randomly chosen subset of the coordinate directions. Using (29) we have the following iteration

$$x_{n+1} = x_n - \frac{d}{m} h \sum_{i=1}^{m} (\nabla_{w_i} F(x_n)) e_{w_i}$$

which is known as stochastic coordinate descent [22].

In practice, one might have more information about the properties of the gradients in different directions. In particular, one can try to exploit the fact that the

Lipschitz constant L_i of $\nabla_i F$ might be different for each i. In this scenario, we may take $m = 1$ and use the stepsize $h = 1/(dL_{w_1})$, giving rise to the following iteration [23]:

$$x_{n+1} = x_n - \frac{1}{L_{w_1}} \nabla_{w_1} F(x_n) e_{w_1},$$

where w_1 follows the uniform distribution on the set of integers $[d]$. It is also worth mentioning that one can group different directions together in (28) and use a similar estimator to (29) to give rise to what is known as block stochastic coordinate descent [23].

4. **Backward error analysis for stochastic optimization algorithms.** Our starting point in this section is the general form of stochastic optimization algorithms (24). In particular, it is not difficult to show that if $\widehat{\nabla} F(x, w)$ is an unbiased estimator of ∇F then (24) approximates weakly to first order the solution of the ODE (23). More precisely, expectations of functionals of (24) would be $\mathcal{O}(h)$ away from the corresponding functions of the deterministic solution of (23). Regarding (24) as a time-stepping method, our objective is to find a modified equation that it approximates weakly to second-order. We will do this by following the approach first introduced in [32]. In particular, we will look for an SDE of the form

$$d\tilde{X} = \left(-\nabla F(\tilde{X}) + h F_1(\tilde{X}) \right) dt + \sqrt{h}\, G_1(\tilde{X}) dW \tag{30}$$

and seek F_1 and G_1 such that the one-step expansion $u(x, h) = \mathbb{E}(\phi(x_1)|x_0 = x)$ is $O(h^3)$ from $\tilde{u}(x, h) = \mathbb{E}(\phi(\tilde{X}(h))|x_0 = x)$. Since the numerical solution (24) is a first-order weak approximation to (23) it admits the following expansion[3]

$$U(x, h) = \phi(x) + h\mathcal{L}\phi(x) + h^2 \mathcal{A}_1 \phi(x) + \cdots$$

where

$$\mathcal{L}\phi = -\nabla F \cdot \nabla \phi, \tag{31a}$$

$$\mathcal{A}_1 \phi = \frac{1}{2} \mathbb{E}(\widehat{\nabla} F(x, w)(\widehat{\nabla} F(x, w))^T) : \nabla\nabla\phi. \tag{31b}$$

Furthermore, similar to the case of backward error analysis for SDEs, we have that

$$\mathcal{L}_1 \phi = \mathcal{A}_1 \phi - \frac{1}{2}\mathcal{L}^2 \phi, \quad \text{where} \quad \mathcal{L}_1 = F_1 \cdot \nabla + \frac{1}{2} G_1 G_1^T : \nabla\nabla,$$

in order for the local weak error between the stochastic optimization scheme (24) and the true solution of the modified equation to be $\mathcal{O}(h^3)$. By equating $\nabla\phi$ terms we find that

$$F_1 = -\frac{1}{2}(\nabla\nabla F)\nabla F = -\frac{1}{4}\nabla \left\| \nabla F \right\|^2.$$

By equating $\nabla\nabla\phi$ terms we find that

$$G_1 = \sqrt{\mathbb{E}[(\widehat{\nabla} F - \nabla F)(\widehat{\nabla} F - \nabla F)^T]},$$

where $\sqrt{\cdot}$ denotes the principal matrix square root. We hence have the following proposition.

[3] details of this calculation can be found in the Appendix.

Proposition 4.1. *Let x_n be the iterates from a stochastic optimization scheme of the form (24), where $\mathbb{E}(\widehat{\nabla}F(x,w)) = \nabla F(x)$. In addition, consider the solution to the equation*

$$d\tilde{X} = -\nabla\left(F(\tilde{X}) + \frac{h}{4}\left\|\nabla F(\tilde{X})\right\|^2\right)dt + \sqrt{h}\sqrt{\Sigma(\tilde{X})}dW, \tag{32}$$

where

$$\Sigma(\tilde{X}) = \mathbb{E}[(\widehat{\nabla}F(\tilde{X},w) - \nabla F(\tilde{X}))(\widehat{\nabla}F(\tilde{X},w) - \nabla F(\tilde{X}))^T]. \tag{33}$$

Then

$$\mathbb{E}(\phi(\tilde{X}(T))) - \mathbb{E}(\phi(x_n)) = \mathcal{O}(h^2).$$

Remark 1. Equation (32) coincides with the modified equation that appeared in [19] in the case of stochastic gradient descent, i.e., when (26) holds.

4.1. **Application to stochastic coordinate descent.** We now study the properties of stochastic coordinate descent in the case where $m = 1$ in (29). In this case, the iteration becomes

$$x_{n+1} = x_n - d\,h\,(\nabla_{w_1}F(x_n))e_{w_1},$$

which can also be rewritten using the full gradient of F in the following way

$$x_{n+1} = x_n - d\,h\,U_{w_1}\nabla F(x_n), \tag{34}$$

where w_1 is a random variable following the uniform distribution on the set of integers $[d]$ and U_{w_1} is a matrix that has w_1th column given by e_{w_1} and is zero elsewhere. We may then state a proposition about the form of the matrix $\Sigma(\tilde{X})$ in (33).

Proposition 4.2. *For the stochastic estimator (29) with $m = 1$, the matrix $\Sigma(\tilde{X})$ in (33) may be written*

$$\Sigma(\tilde{X}) = d\sum_{i=1}^{d} U_i(\nabla F(\tilde{X}))(\nabla F(\tilde{X}))^T U_i^T - (\nabla F(\tilde{X}))(\nabla F(\tilde{X}))^T.$$

Proof. We have

$$\begin{aligned}
\Sigma(\tilde{X}) &= \mathbb{E}[(\widehat{\nabla}F(\tilde{X},w) - \nabla F(\tilde{X}))(\widehat{\nabla}F(\tilde{X},w) - \nabla F(\tilde{X}))^T] \\
&= \mathbb{E}[(\widehat{\nabla}F(\tilde{X},w))(\widehat{\nabla}F(\tilde{X},w))^T] - (\nabla F(\tilde{X}))(\nabla F(\tilde{X}))^T \\
&= d^2\mathbb{E}[U_{w_1}(\nabla F(\tilde{X},w)))(\nabla F(\tilde{X},w))^T U_{w_1}^T] - (\nabla F(\tilde{X}))(\nabla F(\tilde{X}))^T \\
&= d\sum_{i=1}^{d} U_i(\nabla F(\tilde{X}))(\nabla F(\tilde{X}))^T U_i^T - (\nabla F(\tilde{X}))(\nabla F(\tilde{X}))^T,
\end{aligned}$$

where in the last line we have used the fact the w_1 is random variable with a uniform distribution on $[d]$ (since the random index i is chosen uniformly at random from the set $[d]$). □

Having calculated an expression for $\Sigma(\tilde{X})$ we may now study the mean square stability properties of the modified equation (32). To proceed, we make the following assumptions.

Assumption 4.3. *There exists a constant $L > 0$ such that*

$$\|\nabla F(x) - \nabla F(y)\| \le L\|x - y\|, \quad \forall x, y \in \mathbb{R}^d.$$

Assumption 4.4. *There exists a constant $\mu > 0$ such that*

$$\langle x - y, \nabla F(x) - \nabla F(y)\rangle \geq \mu\|x - y\|^2, \quad \forall x, y \in \mathbb{R}^d.$$

Assumption 4.5. *There exists a constant $K > 0$ such that*

$$\langle x - y, \nabla\nabla F(x)\nabla F(x) - \nabla\nabla F(y)\nabla F(y)\rangle \geq K\|x - y\|^2, \quad \forall x, y \in \mathbb{R}^d.$$

Remark 2. Assumption 4.4 is equivalent to strong convexity and hence implies that the function $F(x)$ has a unique minimizer. In addition, in the quadratic case $F(x) = \frac{1}{2}x^T Ax + b^T x$, where A is a positive definite matrix, Assumptions 4.3–4.5 are satisfied; we may take $L = \lambda_{max}$, $\mu = \lambda_{\min}$ and $K = \lambda_{\min}^2$, where $\lambda_{\min}, \lambda_{\max}$ are the smallest and largest eigenvalues of the matrix A. Furthermore, the first two assumptions are automatically satisfied if $f \in \mathcal{F}_{\mu,L}$ the set of L-smooth and μ-strongly convex functions.

We then have the following theorem.

Theorem 4.6. *Consider the modified SDE (32) for the stochastic estimator (29), hence with $\Sigma(\tilde{X})$ given in Proposition 4.2. Under Assumptions 4.3–4.5, and assuming that $X(0)$ is deterministic, the solution $X(t)$ of (32) satisfies*

$$\mathbb{E}\left[\|X(t) - X_\star\|^2\right] \leq e^{-\alpha t}\|X(0) - X_\star\|^2 \tag{35}$$

with $\alpha = 2\mu + hK - h(d-1)L^2$ and $X_\star$ is the unique minimizer of F. Hence, if the stepsize satisfies $h < 2\mu/((d-1)L^2 - K)$ we have

$$\lim_{t\to\infty} \mathbb{E}\left[\|X(t) - X_\star\|^2\right] = 0. \tag{36}$$

Proof. Applying the Ito formula to the function $\phi(X(t)) = \|X(t) - X_\star\|^2$, where $X(t)$ solves (32) and $X_\star$ is the minimizer of F, we get

$$d\|X(t) - X_\star\|^2 = \Big[-2\left\langle X(t) - X_\star, \nabla F(X(t))\right\rangle - \frac{h}{2}\left\langle X(t) - X_*, \nabla\|\nabla F(X(t))\|^2\right\rangle . \\ + h\mathrm{Tr}\left(\Sigma(X(t))\right)\Big]dt + M_g,$$

where M_g is a Martingale. Since

$$\begin{aligned} \nabla\|\nabla F(X)\|^2 &= 2\nabla\nabla F(X)\nabla F(X), \\ \mathrm{Tr}\left(\Sigma(X)\right) &= (d-1)\|\nabla F(X)\|^2, \end{aligned}$$

we have

$$d\|X(t) - X_\star\|^2 = \Big[-2\left\langle X(t) - X_\star, \nabla F(X(t)) + \frac{h}{2}\nabla\nabla F(X(t))\nabla F(X(t))\right\rangle \\ + h(d-1)\left\|\nabla F(x(t))\right\|^2\Big]dt + M_g.$$

Now using the fact that $\nabla F(X_\star) = 0$ along with Assumptions 4.3–4.5, we obtain

$$d\|X(t) - X_\star\|^2 \leq -\alpha\|X(t) - X_\star\|^2 + M_g,$$

where $\alpha := 2\mu - h\big((d-1)L^2 - K\big)$. Taking an expectation and using the Gronwall inequality then gives us the desired result. □

Remark 3. For $d = 1$, the iteration (34) becomes deterministic and corresponds to gradient descent for a univariate function. In this case, Theorem 4.6 implies deterministic convergence to the minimizer $X_\star$ with rate $-2\mu - hK$. When $h = 0$ this rate coincides with the rate of convergence that we expect for the gradient flow (23) under Assumption 4.4. Furthermore, when $h > 0$ we see that the modified equation bound for gradient descent convergences more rapidly than the original gradient flow bound.

Remark 4. In Theorem 4.6, if one were to use a different stochastic gradient method, the only change in (37) involves the term associated with the trace of $\Sigma(X)$, which in turn directly relates to the covariance of the stochastic estimator $\widehat{\nabla} F(X, w)$. In the case of the stochastic gradient method in particular, it would no longer be true that $\mathrm{Tr}\left(\Sigma(X_\star)\right) = 0$ or that this term could be directly bounded by a function of $\nabla F(X)$, and hence one cannot conclude that the solution to the modified equation (32) is mean-square stable. Nevertheless, under Assumptions 4.4–4.5 there would be a period of time for which the dynamics of the modified equation would be contractive before driven by the properties of $\Sigma(X)$. This coincides precisely with the behaviour of the stochastic gradient descent method discussed in [19].

Remark 5. In [19] a first-order modified equation was also proposed which does not have the h-dependent term in the drift in (32). In this case the constant α in Theorem 4.6 weakens to $\alpha = 2\mu - h(d-1)L^2$, failing to capture the potential benefits of the extra gradient term.

5. Discussion.

Modified equations provide a useful approach for creating continuous-time models that can be more straightforward to analyze than an underlying discrete iteration. In this work, following on from [19], the modified equations take the form of stepsize-dependent diffusion equations that accurately describe the dynamics of a stochastic optimization algorithm.

There are many avenues for future work in this area. The general setting analyzed in Proposition 4.2 raises the possibility of comparing the modified equations associated with a range of stochastic optimization algorithms in order to gain new insights into their relative strengths and weaknesses. It would also be of interest to derive and study modified equations that approximate stochastic optimization algorithms to higher order. Further, incorporating a variable stepsize, or "adaptive learning rate," regime into the modified equation framework would greatly expand the scope of these results.

Appendix A. Useful calculations.

We now report on the calculations used to achieve (31a)–(31b). Given $x_0 = x$, the single-step iteration of the stochastic method (24) reads

$$x_1 = x - h\widehat{\nabla} F(x, w).$$

For any test function $\phi : \mathbb{R}^d \mapsto \mathbb{R}$, Taylor expansion of $\phi(x_1)$ around x gives

$$\phi(x_1) = \phi(x) - h\nabla\phi(x) \cdot \widehat{\nabla} F(x, w) + \frac{1}{2} h^2 \widehat{\nabla} F(x, w) \cdot \nabla\nabla\phi(x) \widehat{\nabla} F(x, w) + \mathcal{O}\left(h^3\right).$$

Using Einstein's repeated index notation, we have

$$\begin{aligned}\widehat{\nabla}F(x,w)\cdot\nabla\nabla\phi(x)\widehat{\nabla}F(x,w) &= \widehat{\nabla}F(x,w)_i\Big(\nabla\nabla\phi(x)\widehat{\nabla}F(x,w)\Big)_i\\ &= \widehat{\nabla}F(x,w)_i\nabla\nabla\phi(x)_{ij}\widehat{\nabla}F(x,w)_j\\ &= \widehat{\nabla}F(x,w)_i\widehat{\nabla}F(x,w)_j\nabla\nabla\phi(x)_{ji}\\ &= \widehat{\nabla}F(x,w)\widehat{\nabla}F(x,w)^T:\nabla\nabla\phi,\end{aligned}$$

where we have used the symmetry of the matrix $\nabla\nabla\phi$. Hence, we obtain

$$\phi(x_1)=\phi(x)-h\nabla\phi(x)\cdot\widehat{\nabla}F(x,w)+\frac{1}{2}h^2\widehat{\nabla}F(x,w)\widehat{\nabla}F(x,w)^T:\nabla\nabla\phi+\mathcal{O}\left(h^3\right). \quad (37)$$

By taking the expectation in (37) and considering that $\mathbb{E}\big(\widehat{\nabla}F(x,w)\big)=\nabla F(x)$, we obtain the desired expansion for $\mathbb{E}\big(\phi(x_1)|x_0=x\big)=u(x,h)$, with the differential operators $\mathcal{L}$ and $\mathcal{A}_1$ given in (31a) and (31b), respectively. We next derive an expansion for $\tilde{u}(x,h)=\mathbb{E}\big(\phi(\tilde{X}(h))|\tilde{X}(0)=x\big)$, where $\tilde{X}(h)$ is the solution to the modified SDE (38) at time $t=h$. For such a process, the generator is given by

$$\tilde{\mathcal{L}}_h=-\nabla F\cdot\nabla+hF_1\cdot\nabla+\frac{1}{2}h\ G_1G_1^T:\nabla\nabla=\mathcal{L}+h\mathcal{L}_1 \quad (38)$$

and a stochastic Taylor expansion gives

$$\tilde{u}(x,h)=\phi(x)+h\tilde{\mathcal{L}}_h\phi(x)+\frac{1}{2}h^2\tilde{\mathcal{L}}_h^2\phi(x)+\cdots. \quad (39)$$

Substituting (38) into (39) and neglecting higher order terms, we obtain

$$\begin{aligned}\tilde{u}(x,h) &= \phi(x)+h\left(\mathcal{L}+h\mathcal{L}_1\right)\phi(x)+\frac{1}{2}h^2\left(\mathcal{L}+h\mathcal{L}_1\right)^2\phi(x)+\cdots\\ &= \phi(x)+h\mathcal{L}\phi(x)+h^2\mathcal{L}_1\phi(x)+\frac{1}{2}h^2\mathcal{L}^2\phi(x)+\cdots \quad (40)\\ &= \phi(x)+h\mathcal{L}\phi(x)+h^2\big(\mathcal{L}_1\phi(x)+\frac{1}{2}\mathcal{L}^2\phi(x)\big)+\cdots.\end{aligned}$$

From (40), we see that $\tilde{u}(x,h)-U(x,h)=\mathcal{O}(h^3)$ if

$$\mathcal{L}_1\phi(x)=\mathcal{A}_1\phi(x)-\frac{1}{2}\mathcal{L}^2\phi(x).$$

REFERENCES

[1] D. Barrett and B. Dherin, *Implicit Gradient Regularization*, In *International Conference on Learning Representations*, 2021.

[2] L. Bottou, F. E. Curtis and J. Nocedal, Optimization methods for large-scale machine learning, *SIAM Review*, **60** (2018), 223-311.

[3] M. P. Calvo, A. Murua and J. M. Sanz-Serna, Modified equations for ODEs, In *Chaotic Numerics (Geelong, 1993)*, *Contemp. Math.*, **172** (1994), 63-74.

[4] A. Debussche and E. Faou, Weak backward error analysis for SDEs, *SIAM J. Numer. Anal.*, **50** (2012), 1735-1752.

[5] A. Durmus, P. Jiménez, E. Moulines and S. Said, On Riemannian stochastic approximation schemes with fixed step-size, *Proceedings of Machine Learning Research*, **130** (2021), 1018-1026.

[6] O. Gonzalez, A. M. Stuart and D. J. Higham, Qualitative properties of modified equations, *IMA J. Numer. Anal.*, **19** (1999), 169-190.

[7] R. M. Gower, N. Loizou, X. Qian, A. Sailanbayev, E. Shulgin and P. Richtárik, SGD: General analysis and improved rates, *Proceedings of Machine Learning Research*, (2019), 5200-5209.
[8] D. F. Griffiths and D. J. Higham, *Numerical Methods for Ordinary Differential Equations: Initial Value Problems*, Springer Undergrad. Math. Ser., Springer-Verlag London, Ltd., London, 2010.
[9] D. F. Griffiths and J. M. Sanz-Serna, On the scope of the method of modified equations, *SIAM J. Scientific and Statistical Computing*, **7** (1986), 994-1008.
[10] E. Hairer, C. Lubich and G. Wanner, *Geometric Numerical Integration. Structure-Preserving Algorithms for Ordinary Differential Equations*, Springer Ser. Comput. Math., 31, Springer-Verlag, Berlin, 2006.
[11] U. Helmke and J. B. Moore, *Optimization and Dynamical Systems*, Springer-Verlag London, Ltd., London, 1994.
[12] D. J. Higham, Trust region algorithms and timestep selection, *SIAM Journal on Numerical Analysis*, **37** (1999), 194-210.
[13] D. J. Higham and P. E. Kloeden, *An Introduction to the Numerical Simulation of Stochastic Differential Equations*, SIAM, Philadelphia, 2021.
[14] N. J. Higham, *Accuracy and Stability of Numerical Algorithms*, Second edition, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2002.
[15] A. Iserles, *A First Course in the Numerical Analysis of Differential Equations*, Cambridge Texts Appl. Math., Cambridge University Press, Cambridge, 2009.
[16] C. T. Kelley and D. E. Keyes, Convergence analysis of pseudo-transient continuation, *SIAM J. Numer. Anal.*, **35** (1998), 508-523.
[17] P. E. Kloeden and E. Platen, *Numerical Solution of Stochastic Differential Equations*, Appl. Math. (N. Y.), 23, Springer-Verlag, Berlin, 1992.
[18] J. Latz, Analysis of stochastic gradient descent in continuous time, *Stat. Comput.*, **31** (2021), Paper No. 39, 25 pp.
[19] Q. Li, C. Tai and W. E, Stochastic modified equations and dynamics of stochastic gradient algorithms I: Mathematical foundations, *J. Mach. Learn. Res.*, **20** (2019), Paper No. 40, 47 pp.
[20] G. N. Milstein and M. V. Tretyakov, *Stochastic Numerics for Mathematical Physics*, Springer-Verlag, Berlin, 2004.
[21] T. Miyagawa, Toward equation of motion for deep neural networks: Continuous-time gradient descent and discretization error analysis, In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, Curran Associates, Inc., **35** (2022), 37778-37791.
[22] Y. Nesterov, Efficiency of coordinate descent methods on huge-scale optimization problems, *SIAM J. Optim.*, **22** (2012), 341-362.
[23] P. Richtarik and M. Takáč, Iteration complexity of randomized block-coordinate descent methods for minimizing a composite function, *Math. Program.*, **144** (2014), 1-38.
[24] H. Robbins and S. Monro, A stochastic approximation method, *Ann. Math. Statist.*, **22** (1951), 400-407.
[25] J. M. Sanz-Serna and M. P. Calvo, *Numerical Hamiltonian Problems*, Appl. Math. Math. Comput., 7, Chapman & Hall, London, 1994.
[26] J. Schropp, Using dynamical systems methods to solve minimization problems, *Appl. Numer. Math.*, **18** (1995), 321-335.
[27] T. Shardlow, Modified equations for stochastic differential equations, *BIT Numer. Math.*, **46** (2006), 111-125.
[28] S. L. Smith, B. Dherin, D. Barrett and S. De, On the origin of implicit regularization in stochastic gradient descent, In *International Conference on Learning Representations*, 2021.
[29] D. Talay and L. Tubaro, Expansion of the global error for numerical schemes solving stochastic differential equations, *Stochastic Process. Appl.*, **8** (1990), 483-509.
[30] J. H. Wilkinson, *Rounding Errors in Algebraic Processes*, Prentice-Hall, Englewood Cliffs, NJ, 1964.

[31] S. J. Wright and B. Recht, *Optimization for Data Analysis*, Cambridge University Press, Cambridge, 2022.
[32] K. C. Zygalakis, On the existence and applications of modified equations for stochastic differential equations, *SIAM J. Sci. Comput.*, **33** (2011), 102-130.

Received September 2023; revised January 2024; early access February 2024.