Continuous Time Analysis of Momentum Methods

Nikola B. Kovachki   NKOVACHKI@CALTECH.EDU
Computing and Mathematical Sciences
California Institute of Technology
Pasadena, CA 91125, USA

Andrew M. Stuart   ASTUART@CALTECH.EDU
Computing and Mathematical Sciences
California Institute of Technology
Pasadena, CA 91125, USA

Editor: Suvrit Sra

Abstract

Gradient descent-based optimization methods underpin the parameter training of neural networks, and hence comprise a significant component in the impressive test results found in a number of applications. Introducing stochasticity is key to their success in practical problems, and there is some understanding of the role of stochastic gradient descent in this context. Momentum modifications of gradient descent such as Polyak's Heavy Ball method (HB) and Nesterov's method of accelerated gradients (NAG), are also widely adopted. In this work our focus is on understanding the role of momentum in the training of neural networks, concentrating on the common situation in which the momentum contribution is fixed at each step of the algorithm. To expose the ideas simply we work in the deterministic setting.

Our approach is to derive continuous time approximations of the discrete algorithms; these continuous time approximations provide insights into the mechanisms at play within the discrete algorithms. We prove three such approximations. Firstly we show that standard implementations of fixed momentum methods approximate a time-rescaled gradient descent flow, asymptotically as the learning rate shrinks to zero; this result does not distinguish momentum methods from pure gradient descent, in the limit of vanishing learning rate. We then proceed to prove two results aimed at understanding the observed practical advantages of fixed momentum methods over gradient descent, when implemented in the non-asymptotic regime with fixed small, but non-zero, learning rate. We achieve this by proving approximations to continuous time limits in which the small but fixed learning rate appears as a parameter; this is known as the method of *modified equations* in the numerical analysis literature, recently rediscovered as the *high resolution ODE* approximation in the machine learning context. In our second result we show that the momentum method is approximated by a continuous time gradient flow, with an additional momentum-dependent second order time-derivative correction, proportional to the learning rate; this may be used to explain the stabilizing effect of momentum algorithms in their transient phase. Furthermore in a third result we show that the momentum methods admit an exponentially attractive invariant manifold on which the dynamics reduces, approximately, to a gradient flow with respect to a modified loss function, equal to the original loss function plus a small perturbation proportional to the learning rate; this small correction provides convexification of the loss function and encodes additional robustness present in momentum methods, beyond the transient phase.

Keywords: Optimization, Machine Learning, Deep Learning, Gradient Flows, Momentum Methods, Modified Equation, Invariant Manifold

1. Introduction

1.1 Background and Literature Review

At the core of many machine learning tasks is solution of the optimization problem

$$\arg\min_{u \in \mathbb{R}^d} \; \Phi(u) \tag{1}$$

where $\Phi : \mathbb{R}^d \to \mathbb{R}$ is an objective (or loss) function that is, in general, non-convex and differentiable. Finding global minima of such objective functions is an important and challenging task with a long history, one in which the use of stochasticity has played a prominent role for many decades, with papers in the early development of machine learning Geman and Geman (1987); Styblinski and Tang (1990), together with concomitant theoretical analyses for both discrete Bertsimas et al. (1993) and continuous problems Kushner (1987); Kushner and Clark (2012). Recent successes in the training of deep neural networks have built on this older work, leveraging the enormous computer power now available, together with empirical experience about good design choices for the architecture of the networks; reviews may be found in Goodfellow et al. (2016); LeCun et al. (2015). Gradient descent plays a prominent conceptual role in many algorithms, following from the observation that the equation

$$\frac{du}{dt} = -\nabla\Phi(u) \tag{2}$$

will decrease Φ along trajectories. The most widely adopted methods use stochastic gradient decent (SGD), a concept introduced in Robbins and Monro (1951); the basic idea is to use gradient decent steps based on a noisy approximation to the gradient of Φ. Building on deep work in the convex optimization literature, momentum-based modifications to stochastic gradient decent have also become widely used in optimization. Most notable amongst these momentum-based methods are the Heavy Ball Method (HB), due to Polyak (1964), and Nesterov's method of accelerated gradients (NAG) Nesterov (1983). To the best of our knowledge, the first application of HB to neural network training appears in Rumelhart et al. (1986). More recent work, such as Sutskever et al. (2013), has even argued for the indispensability of such momentum based methods for the field of deep learning.

From these two basic variants on gradient decent, there have come a plethora of adaptive methods, incorporating momentum-like ideas, such as Adam Kingma and Ba (2014), Adagrad Duchi et al. (2011), and RMSProp Tieleman and Hinton (2012). There is no consensus on which method performs best and results vary based on application. The recent work of Wilson et al. (2017) argues that the rudimentary, non-adaptive schemes SGD, HB, and NAG result in solutions with the greatest generalization performance for supervised learning applications with deep neural network models.

There is a natural physical analogy for momentum methods, namely that they relate to a damped second order Hamiltonian dynamic with potential Φ:

$$m\frac{d^2u}{dt^2} + \gamma(t)\frac{du}{dt} + \nabla\Phi(u) = 0. \tag{3}$$

2

This perspective goes back to Polyak's original work Polyak (1964, 1987) and was further expanded on in Qian (1999), although no proof was given. For NAG, the work of Su et al. (2014) proves that the method approximates a damped Hamiltonian system of precisely this form, with a time-dependent damping coefficient. The analysis in Su et al. (2014) holds when the momentum factor is chosen according to the rule

$$\lambda = \lambda_n = \frac{n}{n+3}, \tag{4}$$

where n is the iteration count; this choice was proposed in the original work of Nesterov (1983) and results in a choice of λ which is asymptotic to 1. In the setting where Φ is μ-strongly convex, it is proposed in Nesterov (2014) that the momentum factor is fixed and chosen close to 1; specifically it is proposed that

$$\lambda = \frac{1 - \sqrt{\mu h}}{1 + \sqrt{\mu h}} \tag{5}$$

where $h > 0$ is the time-step (learning rate). In Wilson et al. (2016), a limiting equation for both HB and NAG of the form

$$\ddot{u} + 2\sqrt{\mu}\dot{u} + \nabla\Phi(u) = 0$$

is derived under the assumption that λ is fixed with respect to iteration number n, and dependent on the time-step h as specified in (5); convergence is obtained to order $\mathcal{O}(h^{1/2})$. Using insight from this limiting equation it is possible to choose the optimal value of μ to maximize the convergence rate in the neighborhood of a locally strongly convex objective function. Further related work is developed in Shi et al. (2018) where separate limiting equations for HB and NAG are derived both in the cases of λ given by (4) and (5), obtaining convergence to order $\mathcal{O}(h^{3/2})$. Much work has also gone into analyzing these methods in the discrete setting, without appeal to the continuous time limits, see Hu and Lessard (2017); Lessard et al. (2016), as well as in the stochastic setting, establishing how the effect on the generalization error, for example, Gadat et al. (2018); Loizou and Richtárik (2017); Yang et al. (2016). In this paper, however, our focus is on the use of continuous time limits as a methodology to explain optimization algorithms.

In many machine learning applications, especially for deep learning, NAG and HB are often used with a constant momentum factor λ that is chosen independently of the iteration count n (contrary to (4)) and independently of the learning rate h (contrary to (5)). In fact, popular books on the subject such as Goodfellow et al. (2016) introduce the methods in this way, and popular articles, such as He et al. (2016) to name one of many, simply state the value of the constant momentum factor used in their experiments. Widely used deep learning libraries such as Tensorflow Abadi et al. (2015) and PyTorch Paszke et al. (2017) implement the methods with a fixed choice of momentum factor. Momentum based methods used in this way, with fixed momentum, have not been carefully analyzed. We will undertake such an analysis, using ideas from numerical analysis, and in particular the concept of *modified equations* Griffiths and Sanz-Serna (1986); Chartier et al. (2007) and from the theory of *attractive invariant manifolds* Hirsch et al. (2006); Wiggins (2013); both ideas are explained in the text Stuart and Humphries (1998). It is noteworthy that

the *high resolution ODE approximation* described in Shi et al. (2018) may be viewed as a rediscovery of the method of modified equations. We emphasize the fact that our work is not at odds with any previous analyses of these methods, rather, we consider a setting which is widely adopted in deep learning applications and has not been subjected to continuous time analysis to date.

Remark 1 *Since publication of this article in Kovachki and Stuart (2021), we became aware of related, and earlier, work by Farazmand (2018). Farazmand starts from the Bregman Lagrangian introduced in Wibisono et al. (2016) and uses ideas from geometric singular perturbation theory to derive an invariant manifold. The work leads to a more general description of the invariant manifold than the one given by our equation* (20)*. Farazmand's work was published in Farazmand (2020).*

1.2 Our Contribution

We study momentum-based optimization algorithms for the minimization task (1), with learning rate independent momentum, fixed at every iteration step, focusing on deterministic methods for clarity of exposition. Our approach is to derive continuous time approximations of the discrete algorithms; these continuous time approximations provide insights into the mechanisms at play within the discrete algorithms. We prove three such approximations. The first shows that the asymptotic limit of the momentum methods, as learning rate approaches zero, is simply a rescaled gradient flow (2). The second two approximations include small perturbations to the rescaled gradient flow, on the order of the learning rate, and give insight into the behavior of momentum methods when implemented with momentum and fixed learning rate. Through these approximation theorems, and accompanying numerical experiments, we make the following contributions to the understanding of momentum methods as often implemented within machine learning:

- We show that momentum-based methods with a fixed momentum factor, satisfy, in the continuous-time limit obtained by sending the learning rate to zero, a rescaled version of the gradient flow equation (2).

- We show that such methods also approximate a damped Hamiltonian system of the form (3), with small mass m (on the order of the learning rate) and constant damping $\gamma(t) = \gamma$; this approximation has the same order of accuracy as the approximation of the rescaled equation (2) but provides a better qualitative understanding of the fixed learning rate momentum algorithm in its transient phase.

- We also show that, for the approximate Hamiltonian system, the dynamics admit an exponentially attractive invariant manifold, locally representable as a graph mapping co-ordinates to their velocities. The map generating this graph describes a gradient flow in a potential which is a small (on the order of the learning rate) perturbation of Φ – see (21); the correction to the potential is convexifying, does not change the global minimum, and provides insight into the fixed learning rate momentum algorithm beyond its initial transient phase.

- We provide numerical experiments which illustrate the foregoing considerations, for simple linear test problems, and for the MNIST digit classification problem; in the

latter case we consider SGD and thereby demonstrate that the conclusions of our theory have relevance for understanding the stochastic setting as well.

Taken together our results are interesting because they demonstrate that the popular belief that (fixed) momentum methods resemble the dynamics induced by (3) is misleading. Whilst it is true, the mass in the approximating equation is small and as a consequence understanding the dynamics as gradient flows (2), with modified potential, is more instructive. In fact, in the first application of HB to neural networks described in Rumelhart et al. (1986), the authors state that "[their] experience has been that [one] get[s] the same solutions by setting [the momentum factor to zero] and reducing the size of [the learning rate]." However our theorems should not be understood to imply that there is no practical difference between momentum methods (with fixed learning rate) and SGD. There is indeed a practical difference as has been demonstrated in numerous papers throughout the machine learning literature, and our experiments in Section 5 further confirm this. We show that while these methods have the same transient dynamics, they are approximated differently. Our results demonstrate that, although momentum methods behave like a gradient descent algorithm, asymptotically, this algorithm has a modified potential. Furthermore, although this modified potential (20) is on the order of the learning rate, the fact that the learning rate is often chosen as large as possible, constrained by numerical stability, means that the correction to the potential may be significant. Our results may be interpreted as indicating that the practical success of momentum methods stems from the fact that they provide a more stable discretization to (2) than the forward Euler method employed in SGD. The damped Hamiltonian dynamic (11), as well the modified potential, give insight into how this manifests. Our work gives further theoretical justification for the exploration of the use of different numerical integrators for the purposes of optimization such as those performed in Scieur et al. (2017); Betancourt et al. (2018); Zhang et al. (2018).

While our analysis is confined to the non-stochastic case to simplify the exposition, the results will, with some care, extend to the stochastic setting using ideas from averaging and homogenization Pavliotis and Stuart (2008) as well as continuum analyses of SGD as in Li et al. (2017); Feng et al. (2018); indeed, in the stochastic setting, sharp uniform in time error estimates are to be expected for empirical averages Mattingly et al. (2010); Dieuleveut et al. (2017). To demonstrate that our analysis is indeed relevant in the stochastic setting, we train a deep autoencoder with mini-batching (stochastic) and verify that our convergence results still hold. The details of this experiment are given in section 5. Furthermore we also confine our analysis to fixed learning rate, and impose global bounds on the relevant derivatives of Φ; this further simplifies the exposition of the key ideas, but is not essential to them; with considerably more analysis the ideas exposed in this paper will transfer to adaptive time-stepping methods and much less restrictive classes of Φ.

The paper is organized as follows. Section 2 introduces the optimization procedures and states the convergence result to a rescaled gradient flow. In section 3 we derive the modified, second-order equation and state convergence of the schemes to this equation. Section 4 asserts the existence of an attractive invariant manifold, demonstrating that it results in a gradient flow with respect to a small perturbation of Φ. In section 5, we train a deep autoencoder, showing that our results hold in a stochastic setting with Assumption

2 violated. We conclude in section 6. All proofs of theorems are given in the appendices so that the ideas of the theorems can be presented clearly within the main body of the text.

1.3 Notation

We use $|\cdot|$ to denote the Euclidean norm on $\mathbb{R}^d$. We define $f : \mathbb{R}^d \to \mathbb{R}^d$ by $f(u) := -\nabla\Phi(u)$ for any $u \in \mathbb{R}^d$. Given parameter $\lambda \in [0, 1)$ we define $\bar{\lambda} := (1 - \lambda)^{-1}$.

For two Banach spaces A, B, and A_0 a subset in A, we denote by $C^k(A_0; B)$ the set of k-times continuously differentiable functions with domain A_0 and range B. For a function $u \in C^k(A_0; B)$, we let $D^j u$ denote its j-th (total) Fréchet derivative for $j = 1, \ldots, k$. For a function $u \in C^k([0, \infty), \mathbb{R}^d)$, we denote its derivatives by $\frac{du}{dt}, \frac{d^2u}{dt^2}$, etc. or equivalently by $\dot{u}, \ddot{u}$, etc.

To simplify our proofs, we make the following assumption about the objective function.

Assumption 2 *Suppose* $\Phi \in C^3(\mathbb{R}^d; \mathbb{R})$ *with uniformly bounded derivatives. Namely, there exist constants* $B_0, B_1, B_2 > 0$ *such that*

$$\|D^{j-1}f\| = \|D^j\Phi\| \le B_{j-1}$$

for $j = 1, 2, 3$ *where* $\|\cdot\|$ *denotes any appropriate operator norm.*

We again stress that this assumption is not key to developing the ideas in this work, but is rather a simplification used to make our results global. Without Assumption 2, and no further assumption on Φ such as convexity, one could only hope to give local results i.e. in the neighborhood of a critical point of Φ. Such analysis could indeed be carried out (see for example Carr (2012)), but we choose not to do so here for the sake of clarity of exposition. In section 5, we give a practical example where this assumption is violated and yet the behavior is as predicted by our theory.

Finally we observe that the nomenclature "learning rate" is now prevalent in machine learning, and so we use it in this paper; it refers to the object commonly referred to as "time-step" in the field of numerical analysis.

2. Momentum Methods and Convergence to Gradient Flow

In subsection 2.1 we state Theorem 3 concerning the convergence of a class of momentum methods to a rescaled gradient flow. Subsection 2.2 demonstrates that the HB and NAG methods are special cases of our general class of momentum methods, and gives intuition for proof of Theorem 3; the proof itself is given in Appendix A. Subsection 2.3 contains a numerical illustration of Theorem 3.

2.1 Main Result

The standard Euler discretization of (2) gives the discrete time optimization scheme

$$\mathsf{u}_{n+1} = \mathsf{u}_n + hf(\mathsf{u}_n), \quad n = 0, 1, 2, \ldots . \tag{6}$$

Implementation of this scheme requires an initial guess $\mathsf{u}_0 \in \mathbb{R}^d$. For simplicity we consider a fixed learning rate $h > 0$. Equation (2) has a unique solution $u \in C^3([0, \infty); \mathbb{R}^d)$ under

Assumption 2 and for $u_n = u(nh)$

$$\sup_{0 \leq nh \leq T} |\mathsf{u}_n - u_n| \leq C(T)h;$$

see Stuart and Humphries (1998), for example.

In this section we consider a general class of momentum methods for the minimization task (1) which can be written in the form, for some $a \geq 0$ and $\lambda \in (0,1)$,

$$\mathsf{u}_{n+1} = \mathsf{u}_n + \lambda(\mathsf{u}_n - \mathsf{u}_{n-1}) + hf(\mathsf{u}_n + a(\mathsf{u}_n - \mathsf{u}_{n-1})), \quad n = 0, 1, 2, \dots,$$
$$\mathsf{u}_1 = \mathsf{u}_0 + hf(\mathsf{u}_0). \tag{7}$$

Again, implementation of this scheme requires an an initial guess $\mathsf{u}_0 \in \mathbb{R}^d$. The parameter choice $a = 0$ gives HB and $a = \lambda$ gives NAG. In Appendix A we prove the following:

Theorem 3 *Suppose Assumption 2 holds and let $u \in C^3([0, \infty); \mathbb{R}^d)$ be the solution to*

$$\frac{du}{dt} = -(1 - \lambda)^{-1}\nabla\Phi(u)$$
$$u(0) = \mathsf{u}_0 \tag{8}$$

with $\lambda \in (0,1)$. For $n = 0, 1, 2, \dots$ let u_n be the sequence given by (7) and define $u_n := u(nh)$. Then for any $T \geq 0$, there is a constant $C = C(T) > 0$ such that

$$\sup_{0 \leq nh \leq T} |u_n - \mathsf{u}_n| \leq Ch.$$

Note that (8) is simply a sped-up version of (2): if v solves (2) and w solves (8) then $v(t) = w((1 - \lambda)t)$ for any $t \in [0, \infty)$. This demonstrates that introduction of momentum in the form used within both HB and NAG results in numerical methods that do not differ substantially from gradient descent.

2.2 Link to HB and NAG

The HB method is usually written as a two-step scheme taking the form (Sutskever et al. (2013))

$$\mathsf{v}_{n+1} = \lambda\mathsf{v}_n + hf(\mathsf{u}_n)$$
$$\mathsf{u}_{n+1} = \mathsf{u}_n + \mathsf{v}_{n+1}$$

with $\mathsf{v}_0 = 0$, $\lambda \in (0,1)$ the momentum factor, and $h > 0$ the learning rate. We can re-write this update as

$$\mathsf{u}_{n+1} = \mathsf{u}_n + \lambda\mathsf{v}_n + hf(\mathsf{u}_n)$$
$$= \mathsf{u}_n + \lambda(\mathsf{u}_n - \mathsf{u}_{n-1}) + hf(\mathsf{u}_n)$$

hence the method reads

$$\mathsf{u}_{n+1} = \mathsf{u}_n + \lambda(\mathsf{u}_n - \mathsf{u}_{n-1}) + hf(\mathsf{u}_n)$$
$$\mathsf{u}_1 = \mathsf{u}_0 + hf(\mathsf{u}_0). \tag{9}$$

Similarly NAG is usually written as (Sutskever et al. (2013))

$$\mathsf{v}_{n+1} = \lambda\mathsf{v}_n + hf(\mathsf{u}_n + \lambda\mathsf{v}_n)$$
$$\mathsf{u}_{n+1} = \mathsf{u}_n + \mathsf{v}_{n+1}$$

with $\mathsf{v}_0 = 0$. Define $\mathsf{w}_n := \mathsf{u}_n + \lambda\mathsf{v}_n$ then

$$\begin{aligned}\mathsf{w}_{n+1} &= \mathsf{u}_{n+1} + \lambda\mathsf{v}_{n+1}\\ &= \mathsf{u}_{n+1} + \lambda(\mathsf{u}_{n+1} - \mathsf{u}_n)\end{aligned}$$

and

$$\begin{aligned}\mathsf{u}_{n+1} &= \mathsf{u}_n + \lambda\mathsf{v}_n + hf(\mathsf{u}_n + \lambda\mathsf{v}_n)\\ &= \mathsf{u}_n + (\mathsf{w}_n - \mathsf{u}_n) + hf(\mathsf{w}_n)\\ &= \mathsf{w}_n + hf(\mathsf{w}_n).\end{aligned}$$

Hence the method may be written as

$$\begin{aligned}\mathsf{u}_{n+1} &= \mathsf{u}_n + \lambda(\mathsf{u}_n - \mathsf{u}_{n-1}) + hf(\mathsf{u}_n + \lambda(\mathsf{u}_n - \mathsf{u}_{n-1}))\\ \mathsf{u}_1 &= \mathsf{u}_0 + hf(\mathsf{u}_0).\end{aligned} \tag{10}$$

It is clear that (9) and (10) are special cases of (7) with $a = 0$ giving HB and $a = \lambda$ giving NAG. To intuitively understand Theorem 3, re-write (8) as

$$\frac{du}{dt} - \lambda\frac{du}{dt} = f(u).$$

If we discretize the du/dt term using forward differences and the $-\lambda du/dt$ term using backward differences, we obtain

$$\frac{u(t + h) - u(t)}{h} - \lambda\frac{u(t) - u(t - h)}{h} \approx f(u(t)) \approx f\left(u(t) + ha\frac{u(t) - u(t - h)}{h}\right)$$

with the second approximate equality coming from the Taylor expansion of f. This can be rearranged as

$$u(t + h) \approx u(t) + \lambda(u(t) - u(t - h)) + hf(u(t) + a(u(t) - u(t - h)))$$

which has the form of (7) with the identification $\mathsf{u}_n \approx u(nh)$.

2.3 Numerical Illustration

Figure 1 compares trajectories of the momentum numerical method (7) with the rescaled gradient flow (8), for the two-dimensional problem $\Phi(u) = \frac{1}{2}\langle u, Qu\rangle$. We pick Q to be positive-definite so that the minimum is achieved at the point $(0, 0)^T$ and make it diagonal so that we can easily control its condition number. In particular, the condition number of Q is given as

$$\kappa = \frac{\max\{Q_{11}, Q_{22}\}}{\min\{Q_{11}, Q_{22}\}}.$$

(a) HB: $\kappa = 5$ (b) HB: $\kappa = 10$ (c) HB: $\kappa = 20$

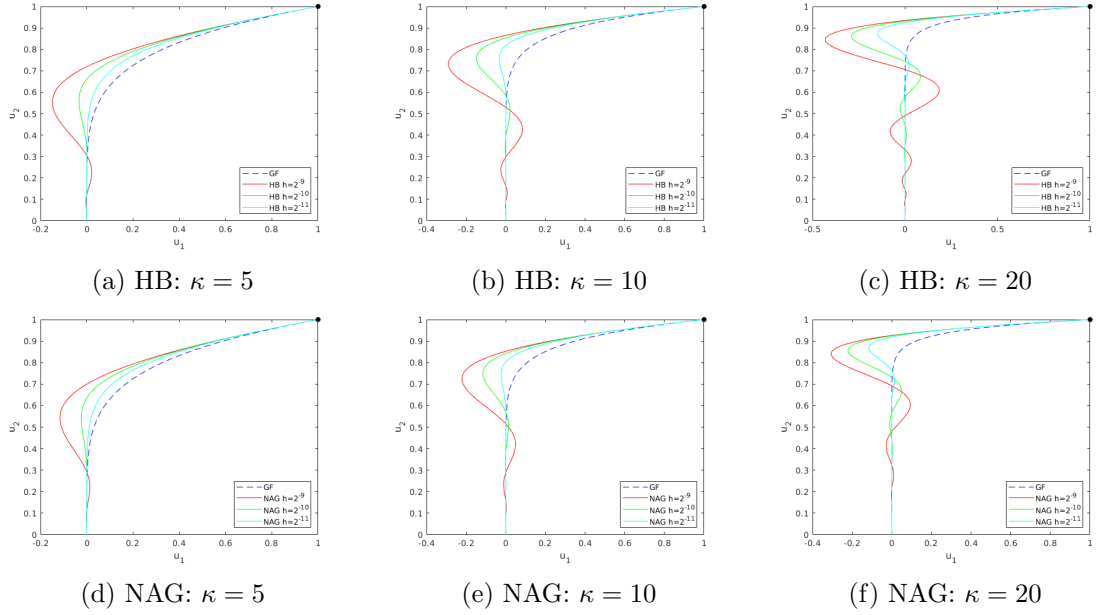(d) NAG: $\kappa = 5$ (e) NAG: $\kappa = 10$ (f) NAG: $\kappa = 20$

Figure 1: Comparison of trajectories for HB and NAG with the gradient flow (8) on the two-dimensional problem $\Phi(u) = \frac{1}{2}\langle u, Qu \rangle$ with $\lambda = 0.9$ fixed. We vary the condition number of Q as well as the learning rate h.
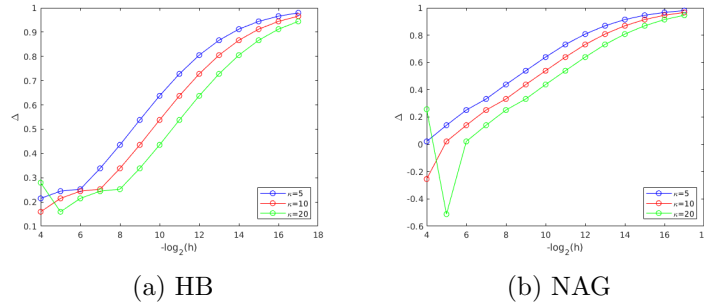

(a) HB   (b) NAG

Figure 2: The numerical rate of convergence, as a function of the learning rate h, of HB and NAG to the gradient flow (8) for the problem described in Figure 1.

We see that, as the condition number is increased, both HB and NAG exhibit more pronounced transient oscillations and are thus further away from the trajectory of (8), however, as the learning rate h is decreased, the oscillations dampen and the trajectories match more and more closely. This observation from Figure 1 is quantified in Figure 2 where we estimate the rate of convergence, as a function of h, which is defined as

$$\Delta = \log_2 \frac{\|\mathsf{u}^{(h)} - u\|_\infty}{\|\mathsf{u}^{(h/2)} - u\|_\infty}$$

where $\mathsf{u}^{(\alpha)}$ is the numerical solution using time-step α. The figure shows that the rate of convergence is indeed close to 1, as predicted by our theory. In summary the behavior

of the momentum methods is precisely that of a rescaled gradient flow, but with initial transient oscillations which capture momentum effects, but disappear as the learning rate is decreased. We model these oscillations in the next section via use of a modified equation.

3. Modified Equations

The previous section demonstrates how the momentum methods approximate a time rescaled version of the gradient flow (2). In this section we show how the same methods may also be viewed as approximations of the damped Hamiltonian system (3), with mass m on the order of the learning rate, using the method of modified equations. In subsection 3.1 we state and discuss the main result of the section, Theorem 4. Subsection 3.2 gives intuition for proof of Theorem 4; the proof itself is given in Appendix B. And the section also contains comments on generalizing the idea of modified equations. In subsection 3.3 we describe a numerical illustration of Theorem 4.

3.1 Main Result

The main result of this section quantifies the sense in which momentum methods do, in fact, approximate a damped Hamiltonian system; it is proved in Appendix B.

Theorem 4 *Fix $\lambda \in (0, 1)$ and assume that $a \geq 0$ is chosen so that $\alpha := \frac{1}{2}(1 + \lambda - 2a(1 - \lambda))$ is strictly positive. Suppose Assumption 2 holds and let $u \in C^4([0, \infty); \mathbb{R}^d)$ be the solution to*

$$h\alpha \frac{d^2 u}{dt^2} + (1 - \lambda)\frac{du}{dt} = -\nabla \Phi(u)$$
$$u(0) = \mathsf{u}_0, \quad \frac{du}{dt}(0) = \mathsf{u}_0'. \tag{11}$$

Suppose further that $h \leq (1 - \lambda)^2 / 2\alpha B_1$. For $n = 0, 1, 2, \ldots$ let u_n be the sequence given by (7) and define $u_n := u(nh)$. Then for any $T \geq 0$, there is a constant $C = C(T) > 0$ such that

$$\sup_{0 \leq nh \leq T} |u_n - \mathsf{u}_n| \leq Ch.$$

Theorem 3 demonstrates the same order of convergence, namely $\mathcal{O}(h)$, to the rescaled gradient flow equation (8), obtained from (11) simply by setting $h = 0$. In the standard method of modified equations the limit system (here (8)) is perturbed by small terms (in terms of the assumed small learning rate) and an increased rate of convergence is obtained to the modified equation (here (11)). In our setting however, because the small modification is to a higher derivative (here second) than appears in the limit equation (here first order), an increased rate of convergence is not obtained. This is due to the nature of the modified equation, whose solution has derivatives that are inversely proportional to powers of h; this fact is quantified in Lemma 9 from Appendix B. It is precisely because the modified equation does not lead to a higher rate of convergence that the initial parameter u_0' is arbitrary; the same rate of convergence is obtained no matter what value it takes.

It is natural to ask, therefore, what is learned from the convergence result in Theorem 4. The answer is that, although the modified equation (11) is approximated at the same order

10

as the limit equation (8), it actually contains considerably more qualitative information about the dynamics of the system, particularly in the early transient phase of the algorithm; this will be illustrated in subsection 3.3. Indeed we will make a specific choice of u_0' in our numerical experiments, namely

$$\frac{du}{dt}(0) = \frac{1 - 2\alpha}{2\alpha - \lambda + 1} f(u_0), \tag{12}$$

to better match the transient dynamics.

3.2 Intuition and Wider Context

3.2.1 Idea Behind The Modified Equations

In this subsection, we show that the scheme (7) exhibits momentum, in the sense of approximating a momentum equation, but the size of the momentum term is on the order of the step size h. To see this intuitively, we add and subtract $u_n - u_{n-1}$ to the right hand size of (7) then we can rearrange it to obtain

$$h\frac{u_{n+1} - 2u_n + u_{n-1}}{h^2} + (1 - \lambda)\frac{u_n - u_{n-1}}{h} = f(u_n + a(u_n - u_{n-1})).$$

This can be seen as a second order central difference and first order backward difference discretization of the momentum equation

$$h\frac{d^2u}{dt^2} + (1 - \lambda)\frac{du}{dt} = f(u)$$

noting that the second derivative term has size of order h.

3.2.2 Higher Order Modified Equations For HB

We will now show that, for HB, we may derive higher order modified equations that are consistent with (9). Taking the limit of these equations yields an operator that agrees with with our intuition for discretizing (8). To this end, suppose $\Phi \in C_b^\infty(\mathbb{R}^d, \mathbb{R})$ and consider the ODE(s),

$$\sum_{k=1}^{p} \frac{h^{k-1}(1 + (-1)^k\lambda)}{k!}\frac{d^ku}{dt^k} = f(u) \tag{13}$$

noting that $p = 1$ gives (8) and $p = 2$ gives (11). Let $u \in C^\infty([0, \infty), \mathbb{R}^d)$ be the solution to (13) and define $u_n := u(nh)$, $u_n^{(k)} := \frac{d^ku}{dt^k}(nh)$ for $n = 0, 1, 2, \ldots$ and $k = 1, 2, \ldots, p$. Taylor expanding yields

$$u_{n\pm1} = u_n + \sum_{k=1}^{p} \frac{(\pm1)^kh^k}{k!}u_n^{(k)} + h^{p+1}I_n^{\pm}$$

where

$$I_n^{\pm} = \frac{(\pm1)^{p+1}}{p!}\int_0^1 (1 - s)^p\frac{d^{p+1}u}{dt^{p+1}}((n \pm s)h)ds.$$

11

Then

$$u_{n+1} - u_n - \lambda(u_n - u_{n-1}) = \sum_{k=1}^{p} \frac{h^k}{k!} u_n^{(k)} + \lambda \sum_{k=1}^{p} \frac{(-1)^k h^k}{k!} u_n^{(k)} + h^{p+1}(I_n^+ - \lambda I_n^-)$$

$$= h \sum_{k=1}^{p} \frac{h^{k-1}(1 + (-1)^k \lambda)}{k!} u_n^{(k)} + h^{p+1}(I_n^+ - \lambda I_n^-)$$

$$= h f(u_n) + h^{p+1}(I_n^+ - \lambda I_n^-)$$

showing consistency to order $p + 1$. As is the case with (11) however, the $I_n^\pm$ terms will be inversely proportional to powers of h hence global accuracy will not improve.

We now study the differential operator on the l.h.s. of (13) as $p \to \infty$. Define the sequence of differential operators $T_p : C^\infty([0, \infty), \mathbb{R}^d) \to C^\infty([0, \infty), \mathbb{R}^d)$ by

$$T_p u = \sum_{k=1}^{p} \frac{h^{k-1}(1 + (-1)^k \lambda)}{k!} \frac{d^k u}{dt^k}, \quad \forall u \in C^\infty([0, \infty), \mathbb{R}^d).$$

Taking the Fourier transform yields

$$\mathcal{F}(T_p u)(\omega) = \sum_{k=1}^{p} \frac{h^{k-1}(1 + (-1)^k \lambda)(i\omega)^k}{k!} \mathcal{F}(u)(\omega)$$

where $i = \sqrt{-1}$ denotes the imaginary unit. Suppose there is a limiting operator $T_p \to T$ as $p \to \infty$ then taking the limit yields

$$\mathcal{F}(Tu)(\omega) = \frac{1}{h}(e^{ih\omega} + \lambda e^{-ih\omega} - \lambda - 1)\mathcal{F}(u)(\omega).$$

Taking the inverse transform and using the convolution theorem, we obtain

$$(Tu)(t) = \frac{1}{h} \mathcal{F}^{-1}(e^{ih\omega} + \lambda e^{-ih\omega} - \lambda - 1)(t) * u(t)$$

$$= \frac{1}{h} \left(-(1 + \lambda)\delta(t) + \lambda \delta(t + h) + \delta(t - h)\right) * u(t)$$

$$= \frac{1}{h} \int_{-\infty}^{\infty} \left(-(1 + \lambda)\delta(t - \tau) + \lambda \delta(t - \tau + h) + \delta(t - \tau - h)\right) u(\tau) \, d\tau$$

$$= \frac{1}{h} \left(-(1 + \lambda)u(t) + \lambda u(t - h) + u(t + h)\right)$$

$$= \frac{u(t + h) - u(t)}{h} - \lambda \left(\frac{u(t) - u(t - h)}{h}\right)$$

where $\delta(\cdot)$ denotes the Dirac-delta distribution and we abuse notation by writing its action as an integral. The above calculation does not prove convergence of T_p to T, but simply confirms our intuition that (9) is a forward and backward discretization of (8).
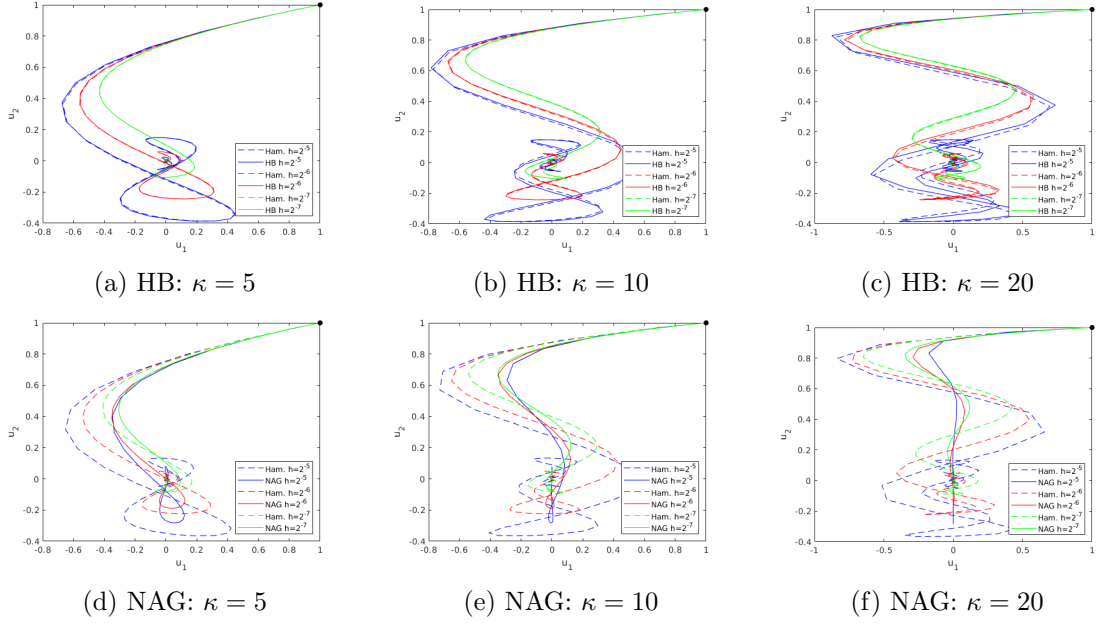
Figure 3: Comparison of trajectories for HB and NAG with the Hamiltonian dynamic (11) on the two-dimensional problem $\Phi(u) = \frac{1}{2}\langle u, Qu \rangle$ with $\lambda = 0.9$ fixed. We vary the condition number of Q as well as the learning rate h.
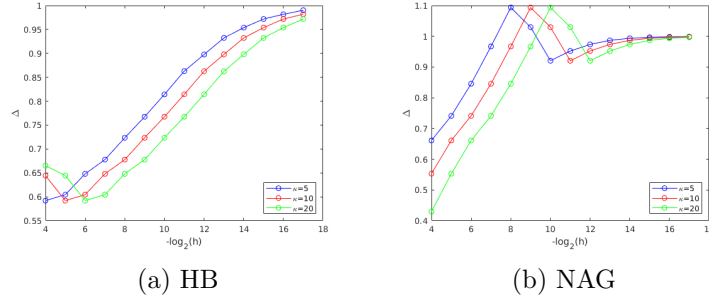

Figure 4: The numerical rate of convergence, as a function of the learning rate h, of HB and NAG to the momentum equation (11) for the problem described in Figure 3.

3.3 Numerical Illustration

Figure 3 shows trajectories of (7) and (11) for different values of a and h on the two-dimensional problem $\Phi(u) = \frac{1}{2}\langle u, Qu \rangle$, varying the condition number of Q. We make the specific choice of u_0' implied by the initial condition (12). Figure 4 shows the numerical order of convergence as a function of h, as defined in Section 2.3, which is near 1, matching our theory. We note that the oscillations in HB are captured well by (11), except for a slight shift when h and κ are large. This is due to our choice of initial condition which cancels the maximum number of terms in the Taylor expansion initially, but the overall rate of convergence remains $\mathcal{O}(h)$ due to Lemma 9. Other choices of u_0' also result in $\mathcal{O}(h)$

convergence and can be picked on a case-by-case basis to obtain consistency with different qualitative phenomena of interest in the dynamics. Note also that $\alpha|_{a=\lambda} < \alpha|_{a=0}$. As a result the transient oscillations in (11) are more quickly damped in the NAG case than in the HB case; this is consistent with the numerical results. However panels (d)-(f) in Figure 3 show that (11) is not able to adequately capture the oscillations of NAG when h is relatively large. We leave for future work, the task of finding equations that are able to appropriately capture the oscillations of NAG in the large h regime.

4. Invariant Manifold

The key lessons of the previous two sections are that the momentum methods approximate a rescaled gradient flow of the form (2) and a damped Hamiltonian system of the form (3), with small mass m which scales with the learning rate, and constant damping γ. Both approximations hold with the same order of accuracy, in terms of the learning rate, and numerics demonstrate that the Hamiltonian system is particularly useful in providing intuition for the transient regime of the algorithm. In this section we link the two theorems from the two preceding sections by showing that the Hamiltonian dynamics with small mass from section 3 has an exponentially attractive invariant manifold on which the dynamics is, to leading order, a gradient flow. That gradient flow is a small, in terms of the learning rate, perturbation of the time-rescaled gradient flow from section 2.

4.1 Main Result

Define

$$\mathsf{v}_n := (\mathsf{u}_n - \mathsf{u}_{n-1})/h \tag{14}$$

noting that then (7) becomes

$$\mathsf{u}_{n+1} = \mathsf{u}_n + h\lambda\mathsf{v}_n + hf(\mathsf{u}_n + ha\mathsf{v}_n)$$

and

$$\mathsf{v}_{n+1} = \frac{\mathsf{u}_{n+1} - \mathsf{u}_n}{h} = \lambda\mathsf{v}_n + f(\mathsf{u}_n + ha\mathsf{v}_n).$$

Hence we can re-write (7) as

$$\begin{aligned} \mathsf{u}_{n+1} &= \mathsf{u}_n + h\lambda\mathsf{v}_n + hf(\mathsf{u}_n + ha\mathsf{v}_n) \\ \mathsf{v}_{n+1} &= \lambda\mathsf{v}_n + f(\mathsf{u}_n + ha\mathsf{v}_n). \end{aligned} \tag{15}$$

Note that if $h = 0$ then (15) shows that $\mathsf{u}_n = \mathsf{u}_0$ is constant in n, and that v_n converges to $(1 - \lambda)^{-1}f(\mathsf{u}_0)$. This suggests that, for h small, there is an invariant manifold which is a small perturbation of the relation $\mathsf{v}_n = \bar{\lambda}f(\mathsf{u}_n)$ and is representable as a graph. Motivated by this, we look for a function $g : \mathbb{R}^d \to \mathbb{R}^d$ such that the manifold

$$\mathsf{v} = \bar{\lambda}f(\mathsf{u}) + hg(\mathsf{u}) \tag{16}$$

is invariant for the dynamics of the numerical method:

$$\mathsf{v}_n = \bar{\lambda}f(\mathsf{u}_n) + hg(\mathsf{u}_n) \iff \mathsf{v}_{n+1} = \bar{\lambda}f(\mathsf{u}_{n+1}) + hg(\mathsf{u}_{n+1}). \tag{17}$$

We will prove the existence of such a function g by use of the contraction mapping theorem to find fixed point of mapping T defined in subsection 4.2 below. We seek this fixed point in set Γ which we now define:

Definition 5 *Let $\gamma, \delta > 0$ be as in Lemmas 10, 11. Define $\Gamma := \Gamma(\gamma, \delta)$ to be the closed subset of $C(\mathbb{R}^d; \mathbb{R}^d)$ consisting of γ-bounded functions:*

$$\|g\|_\Gamma := \sup_{\xi \in \mathbb{R}^d} |g(\xi)| \leq \gamma, \quad \forall g \in \Gamma$$

that are δ-Lipshitz:

$$|g(\xi) - g(\eta)| \leq \delta |\xi - \eta|, \quad \forall g \in \Gamma, \xi, \eta \in \mathbb{R}^d.$$

Theorem 6 *Fix $\lambda \in (0,1)$. Suppose that h is chosen small enough so that Assumption 12 holds. For $n = 0, 1, 2, \ldots$, let $\mathsf{u}_n, \mathsf{v}_n$ be the sequences given by (15). Then there is a $\tau > 0$ such that, for all $h \in (0, \tau)$, there is a unique $g \in \Gamma$ such that (17) holds. Furthermore,*

$$|\mathsf{v}_n - \bar\lambda f(\mathsf{u}_n) - h g(\mathsf{u}_n)| \leq (\lambda + h^2 \lambda \delta)^n |\mathsf{v}_0 - \bar\lambda f(\mathsf{u}_0) - h g(\mathsf{u}_0)|$$

where $\lambda + h^2 \lambda \delta < 1$.

The statement of Assumption 12, and the proof of the preceding theorem, are given in Appendix C. The assumption appears somewhat involved at first glance but inspection reveals that it simply places an upper bound on the learning rate h, as detailed in Lemmas 10, 11. The proof of the theorem rests on the Lemmas 14, 15 and 16 which establish that the operator T is well-defined, maps Γ to Γ, and is a contraction on Γ. The operator T is defined, and expressed in a helpful form for the purposes of analysis, in the next subsection.

In the next subsection we obtain the leading order approximation for g, given in equation (31). Theorem 6 implies that the large-time dynamics are governed by the dynamics on the invariant manifold. Substituting the leading order approximation for g into the invariant manifold (16) and using this expression in the definition (14) shows that

$$\mathsf{v}_n = -(1-\lambda)^{-1} \nabla \left(\Phi(\mathsf{u}_n) + \frac{1}{2} h \bar\lambda (\bar\lambda - a) |\nabla \Phi(\mathsf{u}_n)|^2 \right), \tag{18a}$$

$$\mathsf{u}_n = \mathsf{u}_{n-1} - h(1-\lambda)^{-1} \nabla \left(\Phi(\mathsf{u}_n) + \frac{1}{2} h \bar\lambda (\bar\lambda - a) |\nabla \Phi(\mathsf{u}_n)|^2 \right). \tag{18b}$$

Setting

$$c = \bar\lambda \left(\bar\lambda - a + \frac{1}{2} \right) \tag{19}$$

we see that for large time the dynamics of momentum methods, including HB and NAG, are approximately those of the modified gradient flow

$$\frac{du}{dt} = -(1-\lambda)^{-1} \nabla \Phi_h(u) \tag{20}$$

15

with

$$\Phi_h(u) = \Phi(u) + \frac{1}{2}hc|\nabla\Phi(u)|^2. \tag{21}$$

To see this we proceed as follows. Note that from (20)

$$\frac{d^2u}{dt^2} = -\frac{1}{2}(1-\lambda)^{-2}\nabla|\nabla\Phi(u)|^2 + \mathcal{O}(h)$$

then Taylor expansion shows that, for $u_n = u(nh)$,

$$u_n = u_{n-1} + h\dot{u}_n - \frac{h^2}{2}\ddot{u}_n + \mathcal{O}(h^3)$$

$$= u_{n-1} - h\bar{\lambda}\left(\nabla\Phi(u_n) + \frac{1}{2}hc\nabla|\nabla\Phi(u_n)|^2\right) + \frac{1}{4}h^2\bar{\lambda}^2\nabla|\nabla\Phi(u_n)|^2 + \mathcal{O}(h^3)$$

where we have used that

$$Df(u)f(u) = \frac{1}{2}\nabla\left(|\nabla\Phi(u)|^2\right).$$

Choosing $c = \bar{\lambda}(\bar{\lambda} - a + 1/2)$ we see that

$$u_n = u_{n-1} - h(1-\lambda)^{-1}\nabla\left(\Phi(u_n) + \frac{1}{2}h\bar{\lambda}(\bar{\lambda}-a)|\nabla\Phi(u_n)|^2\right) + \mathcal{O}(h^3). \tag{22}$$

Notice that comparison of (18b) and (22) shows that, on the invariant manifold, the dynamics are to $\mathcal{O}(h^2)$ the same as the equation (20); this is because the truncation error between (18b) and (22) is $\mathcal{O}(h^3)$.

Thus we have proved:

Theorem 7 *Suppose that the conditions of Theorem 6 hold. Then for initial data started on the invariant manifold and any $T \geq 0$, there is a constant $C = C(T) > 0$ such that*

$$\sup_{0 \leq nh \leq T} |u_n - \mathsf{u}_n| \leq Ch^2,$$

where $u_n = u(nh)$ solves the modified equation (20) with $c = \bar{\lambda}(\bar{\lambda} - a + 1/2)$.

4.2 Intuition

We will define mapping $T : C(\mathbb{R}^d; \mathbb{R}^d) \to C(\mathbb{R}^d; \mathbb{R}^d)$ via the equations

$$p = \xi + h\lambda\left(\bar{\lambda}f(\xi) + hg(\xi)\right) + hf\left(\xi + ha\left(\bar{\lambda}f(\xi) + hg(\xi)\right)\right)$$

$$\bar{\lambda}f(p) + h(Tg)(p) = \lambda\left(\bar{\lambda}f(\xi) + hg(\xi)\right) + f\left(\xi + ha\left(\bar{\lambda}f(\xi) + hg(\xi)\right)\right). \tag{23}$$

A fixed point of the mapping $g \mapsto Tg$ will give function g so that, under (23), identity (17) holds. Later we will show that, for g in Γ and all h sufficiently small, ξ can be found from (23a) for every p, and that thus (23b) defines a mapping from $g \in \Gamma$ into $Tg \in C(\mathbb{R}^d; \mathbb{R}^d)$. We will then show that, for h sufficiently small, $T : \Gamma \mapsto \Gamma$ is a contraction.

For any $g \in C(\mathbb{R}^d; \mathbb{R}^d)$ and $\xi \in \mathbb{R}^d$ define

$$w_g(\xi) := \bar{\lambda} f(\xi) + h g(\xi) \tag{24}$$

$$z_g(\xi) := \lambda w_g(\xi) + f\big(\xi + h a w_g(\xi)\big). \tag{25}$$

With this notation the fixed point mapping (23) for g may be written

$$\begin{aligned} p &= \xi + h z_g(\xi), \\ \bar{\lambda} f(p) + h (Tg)(p) &= z_g(\xi). \end{aligned} \tag{26}$$

Then, by Taylor expansion,

$$\begin{aligned} f\Big(\xi + h a\big(\bar{\lambda} f(\xi) + h g(\xi)\big)\Big) &= f\big(\xi + h a w_g(\xi)\big) \\ &= f(\xi) + h a \int_0^1 Df\big(\xi + s h a w_g(\xi)\big) w_g(\xi) ds \\ &= f(\xi) + h a I_g^{(1)}(\xi) \end{aligned} \tag{27}$$

where the last line defines $I_g^{(1)}$. Similarly

$$\begin{aligned} f(p) &= f(\xi + h z_g(\xi)) \\ &= f(\xi) + h \int_0^1 Df\big(\xi + s h z_g(\xi)\big) z_g(\xi) ds \\ &= f(\xi) + h I_g^{(2)}(\xi), \end{aligned} \tag{28}$$

where the last line now defines $I_g^{(2)}$. Then (23b) becomes

$$\bar{\lambda}\big(f(\xi) + h I_g^{(2)}(\xi)\big) + h(Tg)(p) = \lambda \bar{\lambda} f(\xi) + h \lambda g(\xi) + f(\xi) + h a I_g^{(1)}(\xi)$$

and we see that

$$(Tg)(p) = \lambda g(\xi) + a I_g^{(1)}(\xi) - \bar{\lambda} I_g^{(2)}(\xi).$$

In this light, we can rewrite the defining equations (23) for T as

$$p = \xi + h z_g(\xi), \tag{29}$$

$$(Tg)(p) = \lambda g(\xi) + a I_g^{(1)}(\xi) - \bar{\lambda} I_g^{(2)}(\xi). \tag{30}$$

for any $\xi \in \mathbb{R}^d$.

Perusal of the above definitions reveals that, to leading order in h,

$$w_g(\xi) = z_g(\xi) = \bar{\lambda} f(\xi), \quad I_g^{(1)}(\xi) = I_g^{(2)}(\xi) = \bar{\lambda} Df(\xi) f(\xi).$$

Thus setting $h = 0$ in (29), (30) shows that, to leading order in h,

$$g(p) = \bar{\lambda}^2 (a - \bar{\lambda}) Df(p) f(p). \tag{31}$$

Note that since $f(p) = -\nabla \Phi(p)$, Df is the negative Hessian of Φ and is thus symmetric. Hence we can write g in gradient form, leading to

$$g(p) = \frac{1}{2} \bar{\lambda}^2 (a - \bar{\lambda}) \nabla\big(|\nabla \Phi(p)|^2\big). \tag{32}$$

Remark 8 *This modified potential (21) also arises in the construction of Lyapunov functions for the one-stage theta method – see Corollary 5.6.2 in Stuart and Humphries (1998).*

(a) HB: u_n given by (15)

(b) HB: v_n given by (15)

(c) HB: e_n given by (34)

(d) NAG: u_n given by (15)

(e) NAG: v_n given by (15)
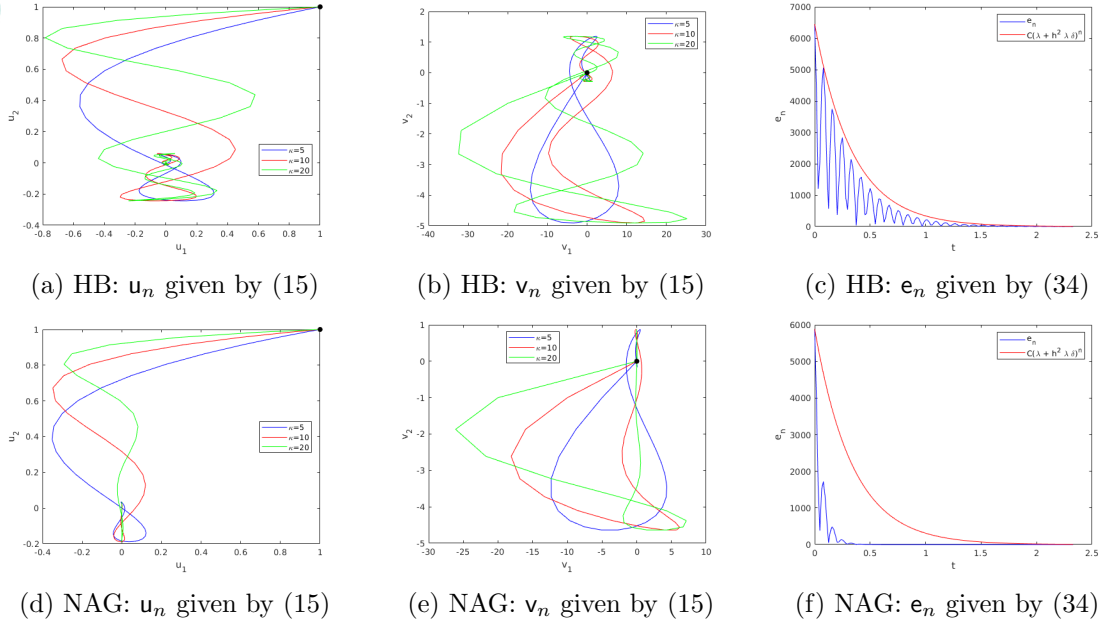
(f) NAG: e_n given by (34)

Figure 5: Invariant manifold for HB and NAG with $h = 2^{-6}$ and $\lambda = 0.9$ on the two-dimensional problem $\Phi(u) = \frac{1}{2}\langle u, Qu \rangle$, varying the condition number of Q. Panels (c), (f) show the distance from the invariant manifold for the largest condition number $\kappa = 20$.

4.3 Numerical Illustration

In Figure 5 panels (a),(b),(d),(e), we plot the components u_n and v_n found by solving (15) with initial conditions $\mathsf{u}_0 = (1,1)^T$ and $\mathsf{v}_n = (0,0)^T$ in the case where $\Phi(u) = \frac{1}{2}\langle u, Qu \rangle$. These initial conditions correspond to initializing the map off the invariant manifold. To leading order in h the invariant manifold is given by (see equation (18))

$$v = -(1-\lambda)^{-1}\nabla\left(\Phi(u) + \frac{1}{2}h\bar{\lambda}(\bar{\lambda} - a)|\nabla\Phi(u)|^2\right). \tag{33}$$

To measure the distance of the trajectory shown in panels (a),(b),(d),(e) from the invariant manifold we define

$$\mathsf{e}_n = \left|\mathsf{v}_n + (1-\lambda)^{-1}\nabla\left(\Phi(\mathsf{u}_n) + \frac{1}{2}h\bar{\lambda}(\bar{\lambda} - a)|\nabla\Phi(\mathsf{u}_n)|^2\right)\right|. \tag{34}$$

Panels (c),(f) show the evolution of e_n as well as the (approximate) bound on it found from substituting the leading order approximation of g into the following upper bound from Theorem 6:

$$(\lambda + h^2\lambda\delta)^n|\mathsf{v}_0 - \bar{\lambda}f(\mathsf{u}_0) - hg(\mathsf{u}_0)|.$$

5. Deep Learning Example

Our theory is developed under quite restrictive assumptions, in order to keep the proofs relatively simple and to allow a clearer conceptual development. The purpose of the numerical experiments in this section is twofold: firstly to demonstrate that our theory sheds

	$h = 2^0$	$h = 2^{-1}$	$h = 2^{-2}$	$h = 2^{-3}$	$h = 2^{-4}$	$h = 2^{-5}$	$h = 2^{-6}$
GF	n/a	4.3948	4.5954	5.6769	**7.0049**	**8.6468**	**10.6548**
HB	3.6775	4.0157	4.5429	5.6447	7.0720	8.7070	10.6848
NAG	**3.2808**	**3.7166**	**4.4579**	**5.6087**	7.0557	8.6987	10.6814
Wilson	6.7395	7.5177	8.3491	9.2543	10.2761	11.3776	12.4123
HB-μ	5.7099	6.6146	7.6202	8.6629	9.7838	11.0039	12.1743
NAG-μ	**5.6867**	**6.6033**	**7.6131**	**8.6556**	**9.7783**	**11.0015**	**12.1738**

Figure 6: Final training errors for the autoencoder on MNIST for six training methods over different learning rates. GF refers to equation (35) while HB and NAG to (7) all with fixed $\lambda = 0.9$.
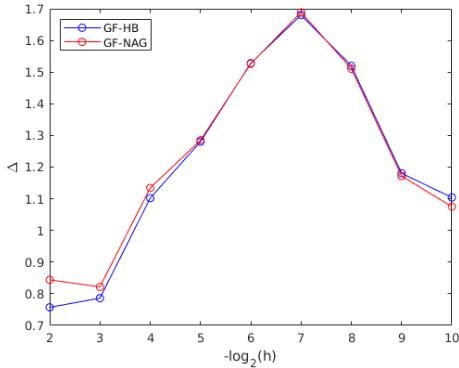
(a) HB, NAG to (35)
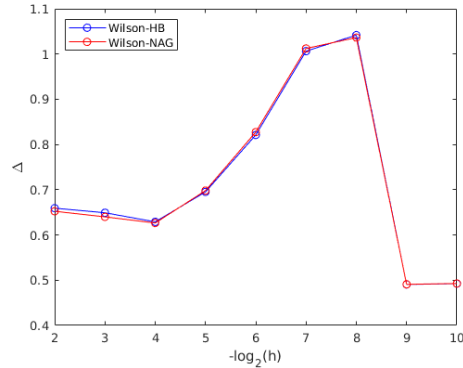
(b) HB-μ, NAG-μ to (36)

Figure 7: The numerical rate of convergence for the parameters of the autoencoder, as a function of the learning rate h, of HB and NAG to (35) (a), as well as of HB-μ and NAG-μ to (36) (b).

light on a stochastic version of gradient descent applied, furthermore, to a setting in which the objective function does not satisfy the global assumptions which facilitate our analysis; and second to show that methods implemented as we use them here (with learning-rate independent momentum, fixed at every step of the iteration) can out-perform other choices on specific problems.

Our numerical experiments in this section are undertaken with in the context of the example given in Sutskever et al. (2013). We train a deep autoencoder, using the architecture of Hinton and Salakhutdinov (2006) on the MNIST dataset LeCun and Cortes (2010). Since our work is concerned only with optimization and not generalization, we present our results only on the training set of 60,000 images and ignore the testing set. We fix an initialization of the autoencoder following Glorot and Bengio (2010) and use it to test every optimization method. Furthermore, we fix a batch size of 200 and train for 500 epochs, not shuffling the data set during training so that each method sees the same realization of the noise. We use the mean-squared error as our loss function.

We compare HB and NAG given by (7) to the re-scaled gradient flow (8) which we discretize in the standard way to yield the numerical method

$$\mathsf{u}_{n+1} = \mathsf{u}_n - \frac{h}{(1-\lambda)}\nabla\Phi(\mathsf{u}_n), \tag{35}$$

hence the momentum term λ only acts to re-scale the learning rate. We do not test against equation (11) because, to discretize it faithfully, we would need to use a time-step much lower than h (because (11) contains a term of order h), but doing so would mean that we need to train for many more epochs compared to HB and NAG so that the same final time is reached. This, in turn, implies that the methods would see different realization of the noise. Thus, to compare them well, we would need to perform a Monte Carlo simulation, however, since we do not state any of our results in a stochastic setting, we leave this for future work.

We also compare our results to those of Wilson et al. (2016) which analyze HB and NAG in the setting where Φ is μ-strongly convex and λ is given by (5) that is

$$\lambda = \frac{1-\sqrt{\mu h}}{1+\sqrt{\mu h}}.$$

They obtain the limiting equation

$$\ddot{u} + 2\sqrt{\mu}\dot{u} + \nabla\Phi(u) = 0$$

which we discretize via a split-step method to yield

$$\begin{aligned}
\mathsf{u}_{n+1} &= \mathsf{u}_n + \frac{1}{2\sqrt{\mu}}\left(1 - e^{-2\sqrt{\mu h}}\right)\mathsf{v}_n \\
\mathsf{v}_{n+1} &= e^{-2\sqrt{\mu h}}\mathsf{v}_n - \sqrt{h}\nabla\Phi(\mathsf{u}_{n+1})
\end{aligned} \tag{36}$$

where we have mapped the the time-step h in HB and NAG to $\sqrt{h}$ as in done in Wilson et al. (2016). We choose this discretization because it allows us to directly solve for the linear parts of the ODE (in the enlarged state-space), yielding a more accurate approximation than the forward-Euler method used to obtain (35). A detailed derivation is given in Appendix D. We will refer to the method in equation (36) as Wilson. Further we refer to equation (7) with λ given by (5) and $a = 0$ as HB-μ and equation (7) with λ given by (5) and $a = \lambda$ as NAG-μ. Since deep neural networks are not strongly convex, there is no single optimal choice of μ; we simply set $\mu = 1$ in our experiments.

Figure 6 gives the final training errors for each method for several learning rates. We were unable to train the autoencoder using (35) with $h = 1$ since $\lambda = 0.9$ implies an effective learning rate of 10 for which the system blows up. In general, NAG is the best performing method for relatively large h which is an observation that is consistently made in the deep learning literature. Further, we note that as the learning rate decreases, the final errors become closer indicating convergence to the appropriate limiting equations. Figure 6 showcases the practical effectiveness of momentum methods as they provide a way of discretizing the gradient flow (2) with a large effective learning rate that forward Euler cannot accommodate. From this perspective, we can view momentum methods as

20

providing a more stable discretization to gradient flows in a manner illustrated by (20). Such a viewpoint informs the works Scieur et al. (2017); Betancourt et al. (2018); Zhang et al. (2018).

To further illustrate the point of convergence to the limiting equation, we compute the numerical rate of convergence, defined in Section 2.3, as a function of h for the neural network parameters between (35) and HB and NAG as well as between (36) and HB-μ and NAG-μ. Figure 7 gives the results. We note that this rate is around 1 as predicted by our theory while the rate for (36) is around 0.5 which is also consistent with the theory in Wilson et al. (2016).

6. Conclusion

Together, equations (8), (11) and (20) describe the dynamical systems which are approximated by momentum methods, when implemented with fixed momentum, in a manner made precise by the four theorems in this paper. The insight obtained from these theorems sheds light on how momentum methods perform optimization tasks.

Acknowledgments

Appendix A

Proof [of Theorem 3] Taylor expanding yields

$$u_{n+1} = u_n + h\bar{\lambda}f(u_n) + \mathcal{O}(h^2)$$

and

$$u_n = u_{n-1} + h\bar{\lambda}f(u_n) + \mathcal{O}(h^2).$$

Hence

$$(1+\lambda)u_n - \lambda u_{n-1} = u_n + h\lambda\bar{\lambda}f(u_n) + \mathcal{O}(h^2).$$

Subtracting the third identity from the first, we find that

$$u_{n+1} - ((1+\lambda)u_n - \lambda u_{n-1}) = hf(u_n) + \mathcal{O}(h^2)$$

by noting $\bar{\lambda} - \bar{\lambda}\lambda = 1$. Similarly,

$$a(u_n - u_{n-1}) = ha\bar{\lambda}f(u_n) + \mathcal{O}(h^2)$$

hence Taylor expanding yields

$$f(u_n + a(u_n - u_{n-1})) = f(u_n) + aDf(u_n)(u_n - u_{n-1})$$
$$+ a^2 \int_0^1 (1-s)D^2f(u_n + sa(u_n - u_{n-1}))[u_n - u_{n-1}]^2 ds$$
$$= f(u_n) + ha\bar{\lambda}Df(u_n)f(u_n) + \mathcal{O}(h^2).$$

From this, we conclude that

$$hf(u_n + a(u_n - u_{n-1})) = hf(u_n) + \mathcal{O}(h^2)$$

hence

$$u_{n+1} = (1+\lambda)u_n - \lambda u_{n-1} + hf(u_n + a(u_n - u_{n-1})) + \mathcal{O}(h^2).$$

Define the error $e_n := u_n - \mathsf{u}_n$ then

$$e_{n+1} = (1+\lambda)e_n - \lambda e_{n-1} + h\left(f(u_n + a(u_n - u_{n-1})) - f(\mathsf{u}_n + a(\mathsf{u}_n - \mathsf{u}_{n-1}))\right) + \mathcal{O}(h^2)$$
$$= (1+\lambda)e_n - \lambda e_{n-1} + h\mathsf{M}_n((1+a)e_n - ae_{n-1}) + \mathcal{O}(h^2)$$

where, from the mean value theorem, we have

$$\mathsf{M}_n = \int_0^1 Df\left(s\left(u_n + a(u_n - u_{n-1})\right) + \left(1-s\right)\left(\mathsf{u}_n + a(\mathsf{u}_n - \mathsf{u}_{n-1})\right)\right)ds.$$

Now define the concatenation $E_{n+1} := [e_{n+1}, e_n] \in \mathbb{R}^{2d}$ then

$$E_{n+1} = A^{(\lambda)}E_n + hA_n^{(a)}E_n + \mathcal{O}(h^2)$$

where $A^{(\lambda)}, A_n^{(a)} \in \mathbb{R}^{2d \times 2d}$ are the block matrices

$$A^{(\lambda)} := \begin{bmatrix} (1+\lambda)I & -\lambda I \\ I & 0I \end{bmatrix}, \quad A_n^{(a)} := \begin{bmatrix} (1+a)\mathsf{M}_n & -a\mathsf{M}_n \\ 0I & 0I \end{bmatrix}$$

with $I \in \mathbb{R}^{d \times d}$ the identity. We note that $A^{(\lambda)}$ has minimal polynomial

$$\mu_{A^{(\lambda)}}(z) = (z - 1)(z - \lambda)$$

and is hence diagonalizable. Thus there is a norm on $\| \cdot \|$ on $\mathbb{R}^{2d}$ such that its induced matrix norm $\| \cdot \|_m$ satifies $\|A^{(\lambda)}\|_m = \rho(A^{(\lambda)})$ where $\rho : \mathbb{R}^{2d \times 2d} \to \mathbb{R}_+$ maps a matrix to its spectral radius. Hence, since $\lambda \in (0, 1)$, we have $\|A^{(\lambda)}\|_m = 1$. Thus

$$\|E_{n+1}\| \leq (1 + h\|A_n^{(a)}\|_m)\|E_n\| + \mathcal{O}(h^2).$$

Then, by finite dimensional norm equivalence, there is a constant $\alpha > 0$, independent of h, such that

$$\|A_n^{(a)}\|_m \leq \alpha \left\| \begin{bmatrix} 1 + a & -a \\ 0 & 0 \end{bmatrix} \otimes \mathsf{M}_n \right\|_2$$
$$= \alpha \sqrt{2a^2 + 2a + 1} \|\mathsf{M}_n\|_2$$

where $\| \cdot \|_2$ denotes the spectral 2-norm. Using Assumption 2, we have

$$\|\mathsf{M}_n\|_2 \leq B_1$$

thus, letting $c := \alpha\sqrt{2a^2 + 2a + 1}B_1$, we find

$$\|E_{n+1}\| \leq (1 + hc)\|E_n\| + \mathcal{O}(h^2).$$

Then, by Grönwall lemma,

$$\|E_{n+1}\| \leq (1 + hc)^n \|E_1\|_n + \frac{(1 + hc)^{n+1} - 1}{ch} \mathcal{O}(h^2)$$
$$= (1 + hc)^n \|E_1\|_n + \mathcal{O}(h)$$

noting that the constant in the $\mathcal{O}(h)$ term is bounded above in terms of T, but independently of h. Finally, we check the initial condition

$$E_1 = \begin{bmatrix} u_1 - \mathsf{u}_1 \\ u_0 - \mathsf{u}_0 \end{bmatrix} = \begin{bmatrix} h(\bar{\lambda} - 1)f(\mathsf{u}_0) + \mathcal{O}(h^2) \\ 0 \end{bmatrix} = \mathcal{O}(h)$$

as desired. ∎

Appendix B

Proof [of Theorem 4] Taylor expanding yields

$$u_{n\pm 1} = u_n \pm h\dot{u}_n + \frac{h^2}{2}\ddot{u}_n \pm \frac{h^3}{2}I_n^{\pm}$$

where

$$I_n^{\pm} = \int_0^1 (1 - s)^2 \dddot{u}((n \pm s)h)ds.$$

Then using equation (11)

$$u_{n+1} - u_n - \lambda(u_n - u_{n-1}) = h(1-\lambda)\dot{u}_n + \frac{h^2}{2}(1+\lambda)\ddot{u}_n + \frac{h^3}{2}(I_n^+ - \lambda I_n^-)$$
$$= hf(u_n) + h^2 a(1-\lambda)\ddot{u}_n + \frac{h^3}{2}(I_n^+ - \lambda I_n^-). \tag{37}$$

Similarly

$$a(u_n - u_{n-1}) = ha\dot{u}_n - \frac{h^2}{2}a\ddot{u}_n + \frac{h^3}{2}aI_n^-$$

hence

$$f(u_n + a(u_n - u_{n-1})) = f(u_n) + haDf(u_n)\dot{u}_n - Df(u_n)\left(\frac{h^2}{2}a\ddot{u}_n - \frac{h^3}{2}aI_n^-\right) + I_n^f$$

where

$$I_n^f = a^2 \int_0^1 (1-s)D^2 f(u_n + sa(u_n - u_{n-1}))[u_n - u_{n-1}]^2 ds.$$

Differentiating (11) yields

$$h\alpha \frac{d^3 u}{dt^3} + (1-\lambda)\frac{d^2 u}{dt^2} = Df(u)\frac{du}{dt}$$

hence

$$hf(u_n + a(u_n - u_{n-1})) = hf(u_n) + h^2 a\left(h\alpha\dddot{u}_n + (1-\lambda)\ddot{u}_n\right) - Df(u_n)\left(\frac{h^3}{2}a\ddot{u}_n - \frac{h^4}{2}aI_n^-\right) + hI_n^f$$
$$= hf(u_n) + h^2 a(1-\lambda)\ddot{u}_n + h^3 a\alpha\dddot{u}_n - Df(u_n)\left(\frac{h^3}{2}a\ddot{u}_n - \frac{h^4}{2}aI_n^-\right) + hI_n^f.$$

Rearranging this we obtain an expression for $hf(u_n)$ which we plug into equation (37) to yield

$$u_{n+1} - u_n - \lambda(u_n - u_{n-1}) = hf(u_n + a(u_n - u_{n-1})) + \mathrm{LT}_n$$

where

$$\mathrm{LT}_n = \underbrace{\frac{h^3}{2}(I_n^+ - \lambda I_n^-)}_{\mathcal{O}\left(h\exp\left(-\frac{(1-\lambda)}{2\alpha}n\right)\right)} - \underbrace{h^3 a\alpha\dddot{u}_n}_{\mathcal{O}\left(h\exp\left(-\frac{(1-\lambda)}{2\alpha}n\right)\right)} + \underbrace{Df(u_n)\left(\frac{h^3}{2}a\ddot{u}_n - \frac{h^4}{2}aI_n^-\right)}_{\mathcal{O}(h^2)} - \underbrace{hI_n^f}_{\mathcal{O}(h^3)}.$$

The bounds (in braces) on the four terms above follow from employing Assumption 2 and Lemma 9. From them we deduce the existence of constants $K_1, K_2 > 0$ independent of h such that

$$|\mathrm{LT}_n| \leq hK_1\exp\left(-\frac{(1-\lambda)}{2\alpha}n\right) + h^2 K_2.$$

We proceed similarly to the proof of Theorem 3, but with a different truncation error structure, and find the error satsifies

$$\|E_{n+1}\| \leq (1 + hc)\|E_n\| + hK_1\exp\left(-\frac{(1-\lambda)}{2\alpha}n\right) + h^2 K_2$$

where we abuse notation and continue to write K_1, K_2 when, in fact, the constants have changed by use of finite-dimensional norm equivalence. Define $K_3 := K_2/c$ then summing this error, we find

$$\|E_{n+1}\| \leq (1+hc)^n\|E_1\| + hK_3((1+hc)^{n+1} - 1) + hK_1\sum_{j=0}^{n}(1+hc)^j\exp\left(-\frac{(1-\lambda)}{2\alpha}(n-j)\right)$$

$$= (1+hc)^n\|E_1\| + hK_3((1+hc)^{n+1} - 1) + hK_1S_n.$$

where

$$S_n = \exp\left(-\frac{(1-\lambda)}{2\alpha}n\right)\left(\frac{(1+hc)^{n+1}\exp\left(\frac{(1-\lambda)}{2\alpha}(n+1)\right) - 1}{(1+hc)\exp\left(\frac{1-\lambda}{2\alpha}\right) - 1}\right).$$

Let $T = nh$ then

$$S_n \leq \frac{(1+hc)^{n+1}\exp\left(\frac{1-\lambda}{2\alpha}\right)}{(1+hc)\exp\left(\frac{1-\lambda}{2\alpha}\right) - 1}$$

$$\leq \frac{2\exp\left(cT + \frac{1-\lambda}{2\alpha}\right)}{\exp\left(\frac{1-\lambda}{2\alpha}\right) - 1}$$

From this we deduce that

$$\|E_{n+1}\| \leq (1+hc)^n\|E_1\| + \mathcal{O}(h)$$

noting that the constant in the $\mathcal{O}(h)$ term is bounded above in terms of T, but independently of h. For the initial condition, we check

$$u_1 - \mathsf{u}_1 = h(\mathsf{u}_0' - f(\mathsf{u}_0)) + \frac{h^2}{2}\ddot{u}_0 + \frac{h^3}{2}I_0^+$$

which is $\mathcal{O}(h)$ by Lemma 9. Putting the bounds together we obtain

$$\sup_{0\leq nh\leq T}\|E_n\| \leq C(T)h.$$

$\blacksquare$

Lemma 9 *Suppose Assumption 2 holds and let $u \in C^3([0,\infty);\mathbb{R}^d)$ be the solution to*

$$h\alpha\frac{d^2u}{dt^2} + (1-\lambda)\frac{du}{dt} = f(u)$$

$$u(0) = \mathsf{u}_0, \quad \frac{du}{dt}(0) = \mathsf{v}_0$$

for some $\mathsf{u}_0, \mathsf{v}_0 \in \mathbb{R}^d$ and $\alpha > 0$ independent of h. Suppose $h \leq (1-\lambda)^2/2\alpha B_1$ then there are constants $C^{(1)}, C_1^{(2)}, C_2^{(2)}, C_1^{(3)}, C_2^{(3)} > 0$ independent of h such that for any $t \in [0,\infty)$,

$$|\dot{u}(t)| \leq C^{(1)},$$

$$|\ddot{u}(t)| \leq \frac{C_1^{(2)}}{h}exp\left(-\frac{(1-\lambda)}{2h\alpha}t\right) + C_2^{(2)},$$

$$|\dddot{u}(t)| \leq \frac{C_1^{(3)}}{h^2}exp\left(-\frac{(1-\lambda)}{2h\alpha}t\right) + C_2^{(3)}.$$

One readily verifies that the result of Lemma 9 is tight by considering the one-dimensional case with $f(u) = -u$. This implies that the result of Theorem 4 cannot be improved without further assumptions.

Proof [of Lemma 9] Define $v \coloneqq \dot{u}$ then

$$\dot{v} = -\frac{1}{h\alpha}\left((1-\lambda)v - f(u)\right).$$

Define $w \coloneqq (1-\lambda)v - f(u)$ hence $\dot{v} = -(1/h\alpha)w$ and $\dot{u} = v = \bar{\lambda}(w + f(u))$. Thus

$$\dot{w} = (1-\lambda)\dot{v} - Df(u)\dot{u}$$
$$= -\frac{(1-\lambda)}{h\alpha}w - Df(u)(\bar{\lambda}(w + f(u))).$$

Hence we find

$$\frac{1}{2}\frac{d}{dt}|w|^2 = -\frac{(1-\lambda)}{h\alpha}|w|^2 - \bar{\lambda}\langle w, Df(u)w\rangle - \bar{\lambda}\langle w, Df(u)f(u)\rangle$$
$$\leq -\frac{(1-\lambda)}{h\alpha}|w|^2 + \bar{\lambda}|\langle w, Df(u)w\rangle| + \bar{\lambda}|\langle w, Df(u)f(u)\rangle|$$
$$\leq -\frac{(1-\lambda)}{h\alpha}|w|^2 + \bar{\lambda}B_1|w|^2 + \bar{\lambda}B_0B_1|w|$$
$$\leq -\frac{(1-\lambda)}{h\alpha}|w|^2 + \frac{(1-\lambda)}{2h\alpha}|w|^2 + \bar{\lambda}B_0B_1|w|$$
$$= -\frac{(1-\lambda)}{2h\alpha}|w|^2 + \bar{\lambda}B_0B_1|w|$$

by noting that our assumption $h \leq (1-\lambda)^2/2\alpha B_1$ implies $\bar{\lambda}B_1 \leq (1-\lambda)/2h\alpha$. Hence

$$\frac{d}{dt}|w| \leq -\frac{(1-\lambda)}{2h\alpha}|w| + \bar{\lambda}B_0B_1$$

so, by Grönwall lemma,

$$|w(t)| \leq \exp\left(-\frac{(1-\lambda)}{2h\alpha}t\right)|w(0)| + 2h\bar{\lambda}^2\alpha B_0B_1\left(1 - \exp\left(-\frac{(1-\lambda)}{2h\alpha}t\right)\right)$$
$$\leq \exp\left(-\frac{(1-\lambda)}{2h\alpha}t\right)|w(0)| + h\beta_1$$

where we define $\beta_1 \coloneqq 2\bar{\lambda}^2\alpha B_0B_1$. Hence

$$|\ddot{u}(t)| = |\dot{v}(t)|$$
$$= \frac{1}{h\alpha}|w(t)|$$
$$\leq \frac{1}{h\alpha}\exp\left(-\frac{(1-\lambda)}{2h\alpha}t\right)|w(0)| + \frac{\beta_1}{\alpha}$$
$$= \frac{|(1-\lambda)\mathsf{v}_0 - f(\mathsf{u}_0)|}{h\alpha}\exp\left(-\frac{(1-\lambda)}{2h\alpha}t\right) + \frac{\beta_1}{\alpha}$$

thus setting $C_1^{(2)} = |(1 - \lambda)\mathsf{v}_0 - f(\mathsf{u}_0)|/\alpha$ and $C_1^{(2)} = \beta_1/\alpha$ gives the desired result. Further,

$$
\begin{aligned}
|\dot{u}(t)| &= |v(t)| \\
&\leq \bar{\lambda}(|w(t)| + |f(u(t))|) \\
&\leq \bar{\lambda}(|w(0)| + h\beta_1 + B_0)
\end{aligned}
$$

hence we deduce the existence of $C^{(1)}$. Now define $z := \dot{w}$ then

$$
\dot{z} = -\frac{(1 - \lambda)}{h\alpha}z - \bar{\lambda}Df(u)z + G(u, v, w)
$$

where we define $G(u, v, w) := -\bar{\lambda}(Df(u)(Df(u)v) + D^2f(u)[v, w] + D^2f(u)[Df(u)v, f(u)])$. Using Assumption 2 and our bounds on w and v, we deduce that there is a constant $C > 0$ independent of h such that

$$
|G(u, v, w)| \leq C
$$

hence

$$
\begin{aligned}
\frac{1}{2}\frac{d}{dt}|z|^2 &= -\frac{(1 - \lambda)}{h\alpha}|z|^2 - \bar{\lambda}\langle z, Df(u)z \rangle + \langle z, G(u, v, w) \rangle \\
&\leq -\frac{(1 - \lambda)}{h\alpha}|z|^2 + \bar{\lambda}B_1|z|^2 + C|z| \\
&\leq -\frac{(1 - \lambda)}{2h\alpha}|z|^2 + C|z|
\end{aligned}
$$

as before. Thus we find

$$
\frac{d}{dt}|z| \leq -\frac{(1 - \lambda)}{2h\alpha}|z| + C
$$

so, by Grönwall lemma,

$$
|z(t)| \leq \exp\left(-\frac{(1 - \lambda)}{2h\alpha}t\right)|z(0)| + h\beta_2
$$

where we define $\beta_2 := 2\bar{\lambda}\alpha C$. Recall that

$$
\ddot{u} = \ddot{v} = -\frac{1}{h\alpha}\dot{w} = -\frac{1}{h\alpha}z
$$

and note

$$
|z(0)| \leq \frac{(1 - \lambda)|(1 - \lambda)\mathsf{v}_0 - f(\mathsf{u}_0)|}{h\alpha} + B_1|\mathsf{v}_0|
$$

hence we find

$$
|\dddot{u}(t)| \leq \left(\frac{(1 - \lambda)|(1 - \lambda)\mathsf{v}_0 - f(\mathsf{u}_0)|}{h^2\alpha^2} + \frac{B_1|\mathsf{v}_0|}{h\alpha}\right)\exp\left(-\frac{(1 - \lambda)}{2h\alpha}t\right) + \frac{\beta_2}{\alpha}.
$$

Thus we deduce that there is a constant $C_1^{(3)} > 0$ independent of h such that

$$
|\dddot{u}(t)| \leq \frac{C_1^{(3)}}{h^2}\exp\left(-\frac{(1 - \lambda)}{2h\alpha}t\right) + C_2^{(3)}
$$

as desired where $C_2^{(3)} = \beta_2/\alpha$. ∎

Appendix C.

For the results of Section 4 we make the following assumption on the size of h. Recall first that by Assumption 2 there are constants $B_0, B_1, B_2 > 0$ such that

$$\|D^{j-1}f\| = \|D^j\Phi\| \leq B_{j-1}$$

for $j = 1, 2, 3$.

Lemma 10 *Suppose $h > 0$ is small enough such that*

$$\lambda + hB_1(a + \lambda\bar{\lambda}) < 1$$

then there is a $\tau_1 > 0$ such that for any $\gamma \in [\tau_1, \infty)$

$$(\lambda + hB_1(a + \lambda\bar{\lambda}))\gamma + \bar{\lambda}B_0B_1(a + \bar{\lambda}) \leq \gamma. \tag{38}$$

Using Lemma 10 fix $\gamma \in [\tau_1, \infty)$ and define the constants

$$K_1 := \bar{\lambda}B_0 + h\gamma$$
$$K_3 := B_0 + \lambda K_1$$
$$\alpha_2 := h^2(\lambda + haB_1),$$
$$\alpha_1 := \lambda - 1 + h\left(B_1(\bar{\lambda} + a(1 + h\bar{\lambda}B_1)) + \lambda\bar{\lambda}(B_1 + hB_2K_3) + ha(aB_2K_1 + B_1\bar{\lambda}(B_1 + hB_2K_3))\right),$$
$$\alpha_0 := aB_2K_1(1 + ha\bar{\lambda}B_1) + \bar{\lambda}(aB_1^2 + B_2K_3) + \bar{\lambda}^2B_1(1 + haB_1)(B_1 + hB_2K_3). \tag{39}$$

Lemma 11 *Suppose $h > 0$ is small enough such that*

$$\alpha_1^2 > 4\alpha_2\alpha_0, \quad \alpha_1 < 0$$

then there are $\tau_2^\pm > 0$ such that for any $\delta \in (\tau_2^-, \tau_2^+]$

$$\alpha_2\delta^2 + \alpha_1\delta + \alpha_0 \leq 0. \tag{40}$$

Using Lemma 11 fix $\delta \in (\tau_2^-, \tau_2^+]$. We make the following assumption on the size of the learning rate h which is achievable since $\lambda \in (0, 1)$.

Assumption 12 *Let Assumption 2 hold and suppose $h > 0$ is small enough such that the assumptions of Lemmas 10, 11 hold. Define $K_2 := \bar{\lambda}B_1 + h\delta$ and suppose $h > 0$ is small enough such that*

$$c := h(\lambda K_2 + B_1(1 + haK_2)) < 1. \tag{41}$$

Define constants

$$Q_1 := \lambda\delta + a(B_1K_2 + B_2K_1(1 + haK_2)) + \bar{\lambda}((B_1 + hB_2K_3)(\lambda K_2 + B_1(1 + haK_2)) + B_2K_3),$$
$$Q_2 := h(a(B_1 + haB_2K_1) + \bar{\lambda}(\lambda + haB_1)(B_1 + hB_2K_3)),$$
$$Q_3 := h(\lambda K_2 + B_1(1 + haK_2)),$$
$$\mu := \lambda + Q_2 + \frac{h^2(\lambda + haB_1)Q_1}{1 - Q_3}. \tag{42}$$

Suppose $h > 0$ is small enough such that

$$Q_3 < 1, \quad \mu < 1. \tag{43}$$

Lastly assume $h > 0$ is small enough such that

$$\lambda + h^2 \lambda \delta < 1. \tag{44}$$

Proof [of Lemma 10.] Since $\lambda + h B_1(a + \lambda \bar{\lambda}) < 1$ and $\bar{\lambda} B_0 B_1(a + \bar{\lambda}) > 0$ the line defined by

$$(\lambda + h B_1(a + \lambda \bar{\lambda}))\gamma + \bar{\lambda} B_0 B_1(a + \bar{\lambda})$$

will intersect the identity line at a positive γ and lie below it thereafter. Hence setting

$$\tau_1 = \frac{\bar{\lambda} B_0 B_1(a + \bar{\lambda})}{1 - \lambda + h B_1(a + \lambda \bar{\lambda})}$$

completes the proof. ∎

Proof [of Lemma 11.] Note that since $\alpha_2 > 0$, the parabola defined by

$$\alpha_2 \delta^2 + \alpha_1 \delta + \alpha_0$$

is upward-pointing and has roots

$$\zeta_\pm = \frac{-\alpha_1 \pm \sqrt{\alpha_1^2 - 4\alpha_2 \alpha_0}}{2\alpha_2}.$$

Since $\alpha_1^2 > 4\alpha_2 \alpha_0$, $\zeta_\pm \in \mathbb{R}$ with $\zeta_+ \neq \zeta_-$. Since $\alpha_1 < 0$, $\zeta_+ > 0$ hence setting $\tau_2^+ = \zeta_+$ and $\tau_2^- = \max\{0, \zeta_-\}$ completes the proof. ∎

The following proof refers to four lemmas whose statement and proof follow it.
Proof [of Theorem 6.] Define $\tau > 0$ as the maximum h such that Assumption 12 holds. The contraction mapping principle together with Lemmas 14, 15, and 16 show that the operator T defined by (29) and (30) has a unique fixed point in Γ. Hence, from its definition and equation (23b), we immediately obtain the existence result. We now show exponential attractivity. Recall the definition of the operator T namely equations (29), (30):

$$p = \xi + h z_g(\xi)$$
$$(Tg)(p) = \lambda g(\xi) + a I_g^{(1)}(\xi) - \bar{\lambda} I_g^{(2)}(\xi).$$

Let $g \in \Gamma$ be the fixed point of T and set

$$p = \mathsf{u}_n + h z_g(\mathsf{u}_n)$$
$$g(p) = \lambda g(\mathsf{u}_n) + a I_g^{(1)}(\mathsf{u}_n) - \bar{\lambda} I_g^{(2)}(\mathsf{u}_n).$$

Then

$$|\mathsf{v}_{n+1} - \bar{\lambda}f(\mathsf{u}_{n+1}) - hg(\mathsf{u}_{n+1})| \leq |\mathsf{v}_{n+1} - \bar{\lambda}f(\mathsf{u}_{n+1}) - hg(p)| + h|g(p) - g(\mathsf{u}_{n+1})|$$
$$\leq |\mathsf{v}_{n+1} - \bar{\lambda}f(\mathsf{u}_{n+1}) - hg(p)| + h\delta|p - \mathsf{u}_{n+1}|$$

since $g \in \Gamma$. Since, by definition,

$$\mathsf{v}_{n+1} = \lambda\mathsf{v}_n + f(\mathsf{u}_n + ha\mathsf{v}_n)$$

we have,

$$|\mathsf{v}_{n+1} - \bar{\lambda}f(\mathsf{u}_{n+1}) - hg(p)| = |\lambda\mathsf{v}_n + f(\mathsf{u}_n + ha\mathsf{v}_n) - \bar{\lambda}f(\mathsf{u}_{n+1}) - h(\lambda g(\mathsf{u}_n) + aI_g^{(1)}(\mathsf{u}_n) - \bar{\lambda}I_g^{(2)}(\mathsf{u}_n))|$$
$$= \lambda|\mathsf{v}_n - \bar{\lambda}f(\mathsf{u}_n) - hg(\mathsf{u}_n)|$$

by noting that

$$f(\mathsf{u}_n + ha\mathsf{v}_n) = f(\mathsf{u}_n) + haI_g^{(1)}(\mathsf{u}_n)$$
$$f(\mathsf{u}_{n+1}) = f(\mathsf{u}_n) + hI_g^{(2)}(\mathsf{u}_n).$$

From definition,

$$\mathsf{u}_{n+1} = \mathsf{u}_n + h\lambda\mathsf{v}_n + hf(\mathsf{u}_n + ha\mathsf{v}_n)$$

thus

$$|p - \mathsf{u}_{n+1}| = |\mathsf{u}_n + hz_g(\mathsf{u}_n) - \mathsf{u}_n - h\lambda\mathsf{v}_n - hf(\mathsf{u}_n + ha\mathsf{v}_n)|$$
$$= h|\lambda(\bar{\lambda}f(\mathsf{u}_n) + hg(\mathsf{u}_n)) + f(\mathsf{u}_n + ha\mathsf{v}_n) - \lambda\mathsf{v}_n - f(\mathsf{u}_n + ha\mathsf{v}_n)|$$
$$= h\lambda|\mathsf{v}_n - \bar{\lambda}f(\mathsf{u}_n) - hg(\mathsf{u}_n)|.$$

Hence

$$|\mathsf{v}_{n+1} - \bar{\lambda}f(\mathsf{u}_{n+1}) - hg(\mathsf{u}_{n+1})| \leq (\lambda + h^2\lambda\delta)|\mathsf{v}_n - \bar{\lambda}f(\mathsf{u}_n) - hg(\mathsf{u}_n)|$$

as desired. By Assumption 12, $\lambda + h^2\lambda\delta < 1$. ∎

The following lemma gives basic bounds which are used in the proof of Lemmas 14, 15, 16.

Lemma 13 *Let $g, q \in \Gamma$ and $\xi, \eta \in \mathbb{R}^d$ then the quantities defined by (24), (25), (27), (28) satisfy the following:*

$$|w_g(\xi)| \leq K_1,$$
$$|w_g(\xi) - w_g(\eta)| \leq K_2|\xi - \eta|,$$
$$|w_g(\xi) - w_q(\xi)| \leq h|g(\xi) - q(\xi)|,$$
$$|z_g(\xi)| \leq K_3,$$
$$|z_g(\xi) - z_g(\eta)| \leq (\lambda K_2 + B_1(1 + haK_2))|\xi - \eta|,$$
$$|z_g(\xi) - z_q(\xi)| \leq h(\lambda + haB_1)|g(\xi) - q(\xi)|,$$
$$|I_g^{(1)}(\xi)| \leq B_1 K_1,$$
$$|I_g^{(1)}(\xi) - I_g^{(1)}(\eta)| \leq (B_1 K_2 + B_2 K_1(1 + haK_2))|\xi - \eta|,$$
$$|I_g^{(1)}(\xi) - I_q^{(1)}(\xi)| \leq h(B_1 + haB_2 K_1)|g(\xi) - q(\xi)|,$$
$$|I_g^{(2)}(\xi)| \leq B_1 K_3$$
$$|I_g^{(2)}(\xi) - I_g^{(2)}(\eta)| \leq ((B_1 + hB_2 K_3)(\lambda K_2 + B_1(1 + haK_2)) + B_2 K_3)|\xi - \eta|,$$
$$|I_g^{(2)}(\xi) - I_q^{(2)}(\xi)| \leq h(\lambda + hB_1 a)(B_1 + hB_2 K_3)|g(\xi) - q(\xi)|.$$

Proof These bounds relay on applications of the triangle inequality together with boundedness of f and its derivatives as well as the fact that functions in Γ are bounded and Lipschitz. To illustrate the idea, we will prove the bounds for $w_g, w_q, I_g^{(1)}$, and $I_q^{(1)}$. To that end,

$$\begin{aligned}
|w_g(\xi)| &= |\bar{\lambda}f(\xi) + hg(\xi)| \\
&\leq \bar{\lambda}|f(\xi)| + h|g(\xi)| \\
&\leq \bar{\lambda}B_0 + h\gamma \\
&= K_1
\end{aligned}$$

establishing the first bound. For the second,

$$\begin{aligned}
|w_g(\xi) - w_g(\eta)| &\leq \bar{\lambda}|f(\xi) - f(\eta)| + h|g(\xi) - g(\eta)| \\
&\leq \bar{\lambda}B_1|\xi - \eta| + h\delta|\xi - \eta| \\
&= K_2|\xi - \eta|
\end{aligned}$$

as desired. Finally,

$$\begin{aligned}
|w_g(\xi) - w_q(\xi)| &= |\bar{\lambda}f(\xi) + hg(\xi) - \bar{\lambda}f(\xi) - hq(\xi)| \\
&= h|g(\xi) - q(\xi)|
\end{aligned}$$

as desired. We now turn to the bounds for $I_g^{(1)}, I_q^{(1)}$,

$$\begin{aligned}
|I_g^{(1)}(\xi)| &\leq \int_0^1 |Df(\xi + shaw_g(\xi))||w_g(\xi)|ds \\
&\leq \int_0^1 B_1 K_1 ds \\
&= B_1 K_1
\end{aligned}$$

establishing the first bound. For the second bound,

$$
\begin{aligned}
|I_g^{(1)}(\xi) - I_g^{(1)}(\eta)| &\leq \int_0^1 |Df(\xi + shaw_g(\xi))w_g(\xi) - Df(\eta + shaw_g(\eta))w_g(\xi)|ds \\
&\quad + \int_0^1 |Df(\eta + shaw_g(\eta))w_g(\xi) - Df(\eta + shaw_g(\eta))w_g(\eta)|ds \\
&\leq K_1 B_2 \int_0^1 (|\xi - \eta| + sha|w_g(\xi) - w_g(\eta)|)ds + B_1|w_g(\xi) - w_g(\eta)| \\
&\leq K_1 B_2(|\xi - \eta| + haK_2|\xi - \eta|) + B_1 K_2|\xi - \eta| \\
&= (B_1 K_2 + B_2 K_1(1 + haK_2))|\xi - \eta|
\end{aligned}
$$

as desired. Finally

$$
\begin{aligned}
|I_g^{(1)}(\xi) - I_q^{(1)}(\xi)| &\leq \int_0^1 |Df(\xi + shaw_g(\xi))w_g(\xi) - Df(\xi + shaw_g(\xi))w_q(\xi)|ds \\
&\quad + \int_0^1 |Df(\xi + shaw_g(\xi))w_q(\xi) - Df(\xi + shaw_q(\xi))w_q(\xi)|ds \\
&\leq B_1 \int_0^1 |w_g(\xi) - w_q(\xi)|ds + K_1 B_2 \int_0^1 |\xi + shaw_g(\xi) - \xi - shaw_q(\xi)|ds \\
&\leq hB_1|g(\xi) - q(\xi)| + h^2 aB_2 K_1|g(\xi) - q(\xi)| \\
&= h(B_1 + haB_2 K_1)|g(\xi) - q(\xi)|
\end{aligned}
$$

as desired. The bounds for $z_g, z_q, I_g^{(2)}$, and $I_q^{(2)}$ follow similarly. ∎

We also need the following three lemmas:

Lemma 14 *Suppose Assumption 12 holds. For any $g \in \Gamma$ and $p \in \mathbb{R}^d$ there exists a unique $\xi \in \mathbb{R}^d$ satisfying (29).*

Lemma 15 *Suppose Assumption 12 holds. The operator T defined by (30) satisfies $T : \Gamma \to \Gamma$.*

Lemma 16 *Suppose Assumption 12 holds. For any $g_1, g_2 \in \Gamma$, we have*

$$
\|Tg_1 - Tg_2\|_\Gamma \leq \mu\|g_1 - g_2\|_\Gamma
$$

where $\mu < 1$.

Now we prove these three lemmas.
Proof [of Lemma 14.] Consider the iteration of the form

$$
\xi^{k+1} = p - hz_g(\xi^k).
$$

For any two sequences $\{\xi^k\}, \{\eta^k\}$ generated by this iteration we have, by Lemma 13,

$$
\begin{aligned}
|\xi^{k+1} - \eta^{k+1}| &\leq h|z_g(\eta^k) - z_g(\xi^k)| \\
&\leq h(\lambda K_2 + B_1(1 + haK_2))|\xi^k - \eta^k| \\
&= c|\xi^k - \eta^k|
\end{aligned}
$$

which is a contraction by (41). ∎

Proof [of Lemma 15.] Let $g \in \Gamma$ and $p \in \mathbb{R}^d$ then by Lemma 14 there is a unique $\xi \in \mathbb{R}^d$ such that (29) is satisfied. Then

$$
\begin{aligned}
|(Tg)(p)| &\leq \lambda|g(\xi)| + a|I_g^{(1)}(\xi)| + \tilde{\lambda}|I_g^{(2)}(\xi)| \\
&\leq \lambda\gamma + aB_1(\tilde{\lambda}B_0 + h\gamma) + \tilde{\lambda}B_1(\lambda(\tilde{\lambda}B_0 + h\gamma) + B_0) \\
&= (\lambda + hB_1(a + \lambda\tilde{\lambda}))\gamma + \tilde{\lambda}B_0B_1(a + \tilde{\lambda}) \\
&\leq \gamma
\end{aligned}
$$

with the last inequality following from (38).

Let $p_1, p_2 \in \mathbb{R}^d$ then, by Lemma 14, there exist $\xi_1, \xi_2 \in \mathbb{R}^d$ such that (29) is satisfied with $p = \{p_1, p_2\}$. Hence, by Lemma 13,

$$
\begin{aligned}
|(Tg)(p_1) - (Tg)(p_2)| &\leq \lambda|g(\xi_1) - g(\xi_2)| + a|I_g^{(1)}(\xi_1) - I_g^{(1)}(\xi_2)| + \tilde{\lambda}|I_g^{(2)}(\xi_1) - I_g^{(2)}(\xi_2)| \\
&\leq K|\xi_1 - \xi_2|
\end{aligned}
$$

where we define

$$
K := \lambda\delta + a(B_1K_2 + B_2K_1(1 + haK_2)) + \tilde{\lambda}((B_1 + hB_2K_3)(\lambda K_2 + B_1(1 + haK_2)) + B_2K_3).
$$

Now, using (29) and the proof of Lemma 14,

$$
\begin{aligned}
|\xi_1 - \xi_2| &\leq |p_1 - p_2| + h|z_g(\xi_1) - z_g(\xi_2)| \\
&\leq |p_1 - p_2| + c|\xi_1 - \xi_2|.
\end{aligned}
$$

Since $c < 1$ by (41), we obtain

$$
|\xi_1 - \xi_2| \leq \frac{1}{1 - c}|p_1 - p_2|
$$

thus

$$
|(Tg)(p_1) - (Tg)(p_2)| \leq \frac{K}{1 - c}|p_1 - p_2| \leq \delta|p_1 - p_2|.
$$

To see the last inequality, we note that

$$
\frac{K}{1 - c} \leq \delta \iff K - \delta(1 - c) \leq 0
$$

and $K - \delta(1 - c) = \alpha_2\delta^2 + \alpha_1\delta + \alpha_0$ by (39) hence (40) gives the desired result. ∎

Proof [of Lemma 16.] By Lemma 14, for any $p \in \mathbb{R}^d$ and $g_1, g_2 \in \Gamma$, there are $\xi_1, \xi_2 \in \mathbb{R}^d$ such that

$$
p = \xi_j + hz_{g_j}(\xi_j)
$$
$$
(Tg_j)(p) = \lambda g_j(\xi_j) + aI_{g_j}^{(1)}(\xi_j) - \tilde{\lambda}I_{g_j}^{(2)}(\xi_j)
$$

33

for $j = 1, 2$. Then

$$|(Tg_1)(p) - (Tg_2)(p)| \leq \lambda|g_1(\xi_1) - g_2(\xi_2)| + a|I_{g_1}^{(1)}(\xi_1) - I_{g_2}^{(1)}(\xi_2)| + \tilde{\lambda}|I_{g_1}^{(2)}(\xi_1) - I_{g_2}^{(2)}(\xi_2)|.$$

Note that

$$\begin{aligned}
|g_1(\xi_1) - g_2(\xi_2)| &= |g_1(\xi_1) - g_2(\xi_2) - g_2(\xi_1) + g_2(\xi_1)| \\
&\leq |g_1(\xi_1) - g_2(\xi_1)| + \delta|\xi_1 - \xi_2|.
\end{aligned}$$

Similarly, by Lemma 13,

$$\begin{aligned}
|I_{g_1}^{(1)}(\xi_1) - I_{g_2}^{(1)}(\xi_2)| &= |I_{g_1}^{(1)}(\xi_1) - I_{g_2}^{(1)}(\xi_2) - I_{g_2}^{(1)}(\xi_1) + I_{g_2}^{(1)}(\xi_1)| \\
&\leq |I_{g_1}^{(1)}(\xi_1) - I_{g_2}^{(1)}(\xi_1)| + |I_{g_2}^{(1)}(\xi_1) - I_{g_2}^{(1)}(\xi_2)| \\
&\leq h(B_1 + haB_2K_1)|g_1(\xi_1) - g_2(\xi_1)| + (B_1K_2 + B_2K_1(1 + haK_2))|\xi_1 - \xi_2|
\end{aligned}$$

Finally,

$$\begin{aligned}
|I_{g_1}^{(2)}(\xi_1) - I_{g_2}^{(2)}(\xi_2)| &= |I_{g_1}^{(2)}(\xi_1) - I_{g_2}^{(2)}(\xi_2) - I_{g_2}^{(2)}(\xi_1) + I_{g_2}^{(2)}(\xi_1)| \\
&\leq |I_{g_1}^{(2)}(\xi_1) - I_{g_2}^{(2)}(\xi_1)| + |I_{g_2}^{(2)}(\xi_1) - I_{g_2}^{(2)}(\xi_2)| \\
&\leq h(\lambda + hB_1a)(B_1 + hB_2K_3)|g_1(\xi_1) - g_2(\xi_1)| + \\
&\quad + ((B_1 + hB_2K_3)(\lambda K_2 + B_1(1 + haK_2)) + B_2K_3)|\xi_1 - \xi_2|
\end{aligned}$$

Putting these together and using (42), we obtain

$$|(Tg_1)(p) - (Tg_2)(p)| \leq (\lambda + Q_2)|g_1(\xi_1) - g_2(\xi_1)| + Q_1|\xi_1 - \xi_2|.$$

Now, by Lemma 13,

$$\begin{aligned}
|\xi_1 - \xi_2| &\leq h|z_{g_1}(\xi_1) - z_{g_2}(\xi_2) - z_{g_2}(\xi_1) + z_{g_2}(\xi_1)| \\
&\leq h(|z_{g_1}(\xi_1) - z_{g_2}(\xi_1)| + |z_{g_2}(\xi_1) - z_{g_2}(\xi_2)|) \\
&\leq h^2(\lambda + haB_1)|g_1(\xi) - g_2(\xi_1)| + h(\lambda K_2 + B_1(1 + haK_2))|\xi_1 - \xi_2| \\
&= h^2(\lambda + haB_1)|g_1(\xi) - g_2(\xi_1)| + Q_3|\xi_1 - \xi_2|
\end{aligned}$$

using (42). Since, by (43), $Q_3 < 1$, we obtain

$$|\xi_1 - \xi_2| \leq \frac{h^2(\lambda + haB_1)}{1 - Q_3}|g_1(\xi_1) - g_2(\xi_1)|$$

and thus

$$\begin{aligned}
|(Tg_1)(p) - (Tg_2)(p)| &\leq \left(\lambda + Q_2 + \frac{h^2(\lambda + haB_1)Q_1}{1 - Q_3}\right)|g_1(\xi_1) - g_2(\xi_1)| \\
&= \mu|g_1(\xi_1) - g_2(\xi_1)|
\end{aligned}$$

by (42). Taking the supremum over ξ_1 then over p gives the desired result. Since $\mu < 1$ by (43), we obtain that T is a contraction on Γ. ∎

34

Appendix D

We consider the equation

$$\ddot{u} + 2\sqrt{\mu}\dot{u} + \nabla\Phi(u) = 0$$
$$u(0) = \mathsf{u}_0, \quad \dot{u}(0) = \mathsf{v}_0.$$

Set $v = \dot{u}$ then we have

$$\begin{bmatrix} \dot{u} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} v \\ -2\sqrt{\mu}v - \nabla\Phi(u) \end{bmatrix}.$$

Define the maps

$$f_1(u,v) := \begin{bmatrix} v \\ -2\sqrt{\mu}v \end{bmatrix}, \quad f_2(u,v) := \begin{bmatrix} 0 \\ -\nabla\Phi(u) \end{bmatrix}$$

then

$$\begin{bmatrix} \dot{u} \\ \dot{v} \end{bmatrix} = f_1(u,v) + f_2(u,v).$$

We first solve the system

$$\begin{bmatrix} \dot{u} \\ \dot{v} \end{bmatrix} = f_1(u,v).$$

Clearly

$$v(t) = e^{-2\sqrt{\mu}t}\mathsf{v}_0$$

hence

$$u(t) = \mathsf{u}_0 + \int_0^t e^{-2\sqrt{\mu}s}\mathsf{v}_0 \, ds$$
$$= \mathsf{u}_0 + \frac{1}{2\sqrt{\mu}}\left(1 - e^{-2\sqrt{\mu}t}\right)\mathsf{v}_0.$$

This gives us the flow map

$$\psi_1(\mathsf{u},\mathsf{v};t) = \begin{bmatrix} \mathsf{u} + \frac{1}{2\sqrt{\mu}}\left(1 - e^{-2\sqrt{\mu}t}\right)\mathsf{v} \\ e^{-2\sqrt{\mu}t}\mathsf{v} \end{bmatrix}.$$

We now solve the system

$$\begin{bmatrix} \dot{u} \\ \dot{v} \end{bmatrix} = f_2(u,v).$$

Clearly

$$u(t) = \mathsf{u}_0$$

hence

$$v(t) = \mathsf{v}_0 - t\nabla\Phi(\mathsf{u}_0).$$

This gives us the flow map

$$\psi_2(\mathsf{u},\mathsf{v};t) = \begin{bmatrix} \mathsf{u} \\ \mathsf{v} - t\nabla\Phi(\mathsf{u}) \end{bmatrix}.$$

35

The composition of the flow maps is then

$$(\psi_2 \circ \psi_1)(\mathsf{u}, \mathsf{v}; t) = \begin{bmatrix} \mathsf{u} + \frac{1}{2\sqrt{\mu}} \left(1 - e^{-2\sqrt{\mu}t}\right) \mathsf{v} \\ e^{-2\sqrt{\mu}t} \mathsf{v} - t\nabla\Phi \left(\mathsf{u} + \frac{1}{2\sqrt{\mu}} \left(1 - e^{-2\sqrt{\mu}t}\right) \mathsf{v}\right) \end{bmatrix}.$$

Mapping t to the time-step $\sqrt{h}$ gives the numerical method (36).

References

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. URL https://www.tensorflow.org/. Software available from tensorflow.org.

Dimitris Bertsimas, John Tsitsiklis, et al. Simulated annealing. *Statistical science*, 8(1): 10–15, 1993.

Michael Betancourt, Michael I. Jordan, and Ashia C. Wilson. On symplectic optimization, 2018.

Jack Carr. *Applications of centre manifold theory*, volume 35. Springer Science & Business Media, 2012.

Philippe Chartier, Ernst Hairer, and Gilles Vilmart. Numerical integrators based on modified differential equations. *Mathematics of computation*, 76(260):1941–1953, 2007.

Aymeric Dieuleveut, Alain Durmus, and Francis Bach. Bridging the gap between constant step size stochastic gradient descent and markov chains. *arXiv preprint arXiv:1707.06386*, 2017.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *J. Mach. Learn. Res.*, 12:2121–2159, July 2011. ISSN 1532-4435. URL http://dl.acm.org/citation.cfm?id=1953048.2021068.

Mohammad Farazmand. Multiscale analysis of accelerated gradient methods. *arXiv:1807.11354*, 2018.

Mohammad Farazmand. Multiscale analysis of accelerated gradient methods. *SIAM Journal on Optimization*, 30(3):2337–2354, 2020.

Yuanyuan Feng, Lei Li, and Jian-Guo Liu. Semigroups of stochastic gradient descent and online principal component analysis: properties and diffusion approximations. *Communications in Mathematical Sciences*, 16(3):777–789, 2018.

Sébastien Gadat, Fabien Panloup, Sofiane Saadane, et al. Stochastic heavy ball. *Electronic Journal of Statistics*, 12(1):461–529, 2018.

Stuart Geman and Donald Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. In *Readings in computer vision*, pages 564–584. Elsevier, 1987.

Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *In Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS'10). Society for Artificial Intelligence and Statistics*, 2010.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

DF Griffiths and JM Sanz-Serna. On the scope of the method of modified equations. *SIAM Journal on Scientific and Statistical Computing*, 7(3):994–1008, 1986.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

Geoffrey Hinton and Ruslan Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504 – 507, 2006.

Morris W Hirsch, Charles Chapman Pugh, and Michael Shub. *Invariant manifolds*, volume 583. Springer, 2006.

Bin Hu and Laurent Lessard. Dissipativity theory for nesterov's accelerated method. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1549–1557. JMLR. org, 2017.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014. URL http://arxiv.org/abs/1412.6980.

Nikola B. Kovachki and Andrew M. Stuart. Continuous time analysis of momentum methods. *Journal of Machine Learning Research*, 22(17):1–40, 2021. URL http://jmlr.org/papers/v22/19-466.html.

Harold J Kushner. Asymptotic global behavior for stochastic approximation and diffusions with slowly decreasing noise effects: global minimization via monte carlo. *SIAM Journal on Applied Mathematics*, 47(1):169–185, 1987.

Harold Joseph Kushner and Dean S Clark. *Stochastic approximation methods for constrained and unconstrained systems*, volume 26. Springer Science & Business Media, 2012.

Yann LeCun and Corinna Cortes. MNIST handwritten digit database. 2010. URL http://yann.lecun.com/exdb/mnist/.

Yann LeCun, Yoshua Bengio, and Geoffrey E. Hinton. Deep learning. *Nature*, 521 (7553):436–444, 2015. doi: 10.1038/nature14539. URL https://doi.org/10.1038/nature14539.

Laurent Lessard, Benjamin Recht, and Andrew Packard. Analysis and design of optimization algorithms via integral quadratic constraints. *SIAM Journal on Optimization*, 26(1): 57–95, 2016.

Qianxiao Li, Cheng Tai, and Weinan E. Stochastic modified equations and adaptive stochastic gradient algorithms. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 2101–2110, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR.

Nicolas Loizou and Peter Richtárik. Linearly convergent stochastic heavy ball method for minimizing generalization error. *arXiv preprint arXiv:1710.10737*, 2017.

Jonathan C Mattingly, Andrew M Stuart, and Michael V Tretyakov. Convergence of numerical time-averaging and stationary measures via poisson equations. *SIAM Journal on Numerical Analysis*, 48(2):552–577, 2010.

Yurii Nesterov. A method of solving a convex programming problem with convergence rate o(1/k2). *Soviet Mathematics Doklady*, 27(2):372–376, 1983.

Yurii Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*. Springer Publishing Company, Incorporated, 1 edition, 2014. ISBN 1461346916, 9781461346913.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *NIPS-W*, 2017.

Grigorios Pavliotis and Andrew Stuart. *Multiscale Methods: Averaging and Homogenization*, volume 53. 01 2008. doi: 10.1007/978-0-387-73829-1.

Boris Polyak. Some methods of speeding up the convergence of iteration methods. *Ussr Computational Mathematics and Mathematical Physics*, 4:1–17, 12 1964. doi: 10.1016/0041-5553(64)90137-5.

Boris T. Polyak. *Introduction to optimization*. New York: Optimization Software, Inc., 1987.

Ning Qian. On the momentum term in gradient descent learning algorithms. *Neural Netw.*, 12(1):145–151, January 1999. ISSN 0893-6080. doi: 10.1016/S0893-6080(98)00116-6. URL http://dx.doi.org/10.1016/S0893-6080(98)00116-6.

Herbert Robbins and Sutton Monro. A stochastic approximation method. *Ann. Math. Statist.*, 22(3):400–407, 09 1951. doi: 10.1214/aoms/1177729586. URL https://doi.org/10.1214/aoms/1177729586.

D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Parallel distributed processing: Explorations in the microstructure of cognition, vol. 1. chapter Learning Internal Representations by Error Propagation, pages 318–362. MIT Press, Cambridge, MA, USA, 1986. ISBN 0-262-68053-X. URL `http://dl.acm.org/citation.cfm?id=104279.104293`.

Damien Scieur, Vincent Roulet, Francis Bach, and Alexandre d'Aspremont. Integration methods and optimization algorithms. In *Advances in Neural Information Processing Systems*, pages 1109–1118, 2017.

Bin Shi, Simon S Du, Michael I Jordan, and Weijie J Su. Understanding the acceleration phenomenon via high-resolution differential equations. *arXiv preprint arXiv:1810.08907*, 2018.

Andrew Stuart and Anthony R Humphries. *Dynamical systems and numerical analysis*, volume 2. Cambridge University Press, 1998.

MA Styblinski and T-S Tang. Experiments in nonconvex optimization: stochastic approximation with function smoothing and simulated annealing. *Neural Networks*, 3(4):467–483, 1990.

Weijie Su, Stephen Boyd, and Emmanuel Candes. A differential equation for modeling nesterov's accelerated gradient method: Theory and insights. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2510–2518. Curran Associates, Inc., 2014.

Ilya Sutskever, James Martens, George Dahl, and Geoffrey Hinton. On the importance of initialization and momentum in deep learning. In *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28*, ICML'13, pages III–1139–III–1147. JMLR.org, 2013. URL `http://dl.acm.org/citation.cfm?id=3042817.3043064`.

T. Tieleman and G. Hinton. Lecture 6.5—RmsProp: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural Networks for Machine Learning, 2012.

Andre Wibisono, Ashia C Wilson, and Michael I Jordan. A variational perspective on accelerated methods in optimization. *proceedings of the National Academy of Sciences*, 113(47):E7351–E7358, 2016.

Stephen Wiggins. *Normally hyperbolic invariant manifolds in dynamical systems*, volume 105. Springer Science & Business Media, 2013.

Ashia C. Wilson, Benjamin Recht, and Michael I. Jordan. A lyapunov analysis of momentum methods in optimization. *CoRR*, abs/1611.02635, 2016. URL `http://arxiv.org/abs/1611.02635`.

Ashia C Wilson, Rebecca Roelofs, Mitchell Stern, Nati Srebro, and Benjamin Recht. The marginal value of adaptive gradient methods in machine learning. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan,

and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 4148–4158. Curran Associates, Inc., 2017. URL `http://papers.nips.cc/paper/7003-the-marginal-value-of-adaptive-gradient-methods-in-machine-learning.pdf`.

Tianbao Yang, Qihang Lin, and Zhe Li. Unified convergence analysis of stochastic momentum methods for convex and non-convex optimization. *arXiv preprint arXiv:1604.03257*, 2016.

Jingzhao Zhang, Aryan Mokhtari, Suvrit Sra, and Ali Jadbabaie. Direct runge-kutta discretization achieves acceleration. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, page 3904–3913, Red Hook, NY, USA, 2018. Curran Associates Inc.