Online Learning via Congregational Gradient Descent

Kim L. Blackmore Robert C. Williamson Iven M. Y. Mareels
William A. Sethares

April 19, 1995

Abstract

We propose and analyse a populational version of stepwise gradient descent suitable for a wide range of learning problems. The algorithm is motivated by genetic algorithms which update a population of solutions rather than just a single representative as is typical for gradient descent. This modification of traditional gradient descent (as used for example in the backpropagation algorithm) avoids getting trapped in local minima. We use an averaging analysis of the algorithm to relate its behaviour to an associated ordinary differential equation. We derive a result concerning how long one has to wait in order that with a given high probability, the algorithm is within a certain neighbourhood of the global minimum. We also analyse the effect of different population sizes. An example is presented which corroborates our theory very well.

1 Introduction

Stepwise Gradient Descent (SGD) schemes are widely used in practice for a range of learning and optimisation problems. In some simple cases, such as a linearly parametrised hypothesis class with quadratic cost, one can ensure there is only one local minimum which is thus the global minimum of the cost function. However in many practically interesting cases, such as neural networks, there can be a large number of local minima. Adaptation of the parameters can thus get stuck and a suboptimal solution can be produced. This paper proposes and analyses a general scheme for modifying online SGD algorithms to alleviate this problem. We show how the running of several versions of a stepwise gradient algorithm in parallel (a "congregation") with periodic selection of the fittest, and concomitant random restarting of the less fit, can ensure convergence to the global minimum. We show this both theoretically and practically via simulation examples. Furthermore we analyse the speed of convergence, and determine the "correct" congregation size to use. The algorithm can be applied in the case where the best parameter value does not give zero cost.

There are several motivations to this work. The first, mentioned in the previous paragraph, has motivated similar work for fixing [18, 17] blind equalisation algorithms such as the Constant Modulus Algorithm (CMA) [35]. This algorithm solves a special sort of learning problem, and is based on the wide-spread LMS algorithm. It has been shown that there are non-global local minima in many practical situations [16]. Techniques have been proposed [18, 17] which statistically detect when the CMA is stuck in a local minimum, and then randomly restart it. We show, as an alternative, how our method can be applied to the CMA, and can ensure global convergence, at relatively little additional computational cost.

The original motivation for this work was to provide a way of making a fair comparison between gradient descent based algorithms and Genetic Algorithms (GAs) for a range of optimisation and learning problems. Genetic algorithms [29, 21] (described in more detail below) are optimisation/adaptation techniques based on an hypothesised model of biological evolution. A key difference between GAs and standard SGD is that GAs evolve a *population* of solutions, whereas SGD evolves only a single solution, which unsurprisingly can get stuck. Our congregational algorithm is perhaps the simplest populational SGD algorithm one can envisage. We have chosen this algorithm because it demonstrates the power of simply running multiple solutions of an existing (locally convergent) online algorithm, and because its simplicity allows the application of averaging theory for a deterministic convergence analysis.

1.1 Relationship to Existing Work

Online algorithms are widely used for learning problems. In practical neural networks, the widespread backpropagation algorithm is a form of SGD. In adaptive signal processing the LMS algorithm is widely used [49]. Computational Learning theorists have analysed

simple cases of online algorithms, often based on SGD [13] in their own framework. There are other analyses of SGD algorithms based on the more traditional adaptive filtering approach [10, 19, 36]. The perceived advantages of SGD are that it is computationally simple, and because of the algorithm's (often exponential) stability, it is robust to noise and model mismatches. Various analyses have shown that there is a non-zero probability of the algorithm actually escaping from local minima, since for non-vanishing adaptation gain the actual algorithm jiggles about the average trajectory [23, 19, 37, 27]. However, one has to wait a time exponential in the depth of the local minima for the escape to occur [23].

Since we plan to compare our scheme with GAs, we should say a little about simple versus complex adaptive systems (GAs are often held to be complex systems, and thus *by definition* not amenable to theoretical analysis). The distinction between simple and complex is usually made in a vague way, and seems simply to denote the complexity of the parametrisation or extent of adaptation undergone. Gell-Mann [24] (pp.292ff) tries to clarify this distinction, but even his classification is still a matter of degree. For the purposes of this work we do not draw *any* distinction between the use of SGD for "simple" problems such as adaptive filtering, and "complex" tasks such as learning in complicated neural networks. Both can be smoothly parametrised, and thus SGD, and the variant proposed in this paper, can be applied to them.

Many people have recognised the relationship between populational methods of optimisation and biological evolution (see e.g. [11] and references 24–39 of [20]). The idea now comes under the heading of Evolutionary Computation (EC) [20]. Apart from genetic algorithms [29, 21] (which are the most widely known form of EC), there are a variety of other methods such as Evolutionary Strategies (ES) [3] and Evolutionary Algorithms (EA) [4]. In order to use most of these algorithms, problems are usually encoded in some binary form. GAs combine different elements of their populations using operations based on theories of biological genetics. EAs and ESs tend to rely more on random perturbations (mutation) of members of the populations. The key idea of these EC techniques is the use of selection (survival of the fittest). According to some schedule, the fitness of the various members is evaluated, and the less fit are removed. New members are then created, in a variety of ways, and the process is repeated. There is a long running debate about the efficacy of crossover versus mutation in creating the next generation and what the "right" method of simulating evolution is. We take the view articulated by Atmar [1] that the key point is the process, not the symptoms; in particular there is a population of solutions, and the unfit are removed. This idea is captured in our proposed algorithm.

Because of the complexity of the algorithms, a theoretical analysis of GAs is quite difficult. Although there have appeared papers proving convergence of GAs [43], such proofs essentially rely on showing that in the limit GAs reduce to a random search, and if one samples the whole space one will eventually find the global minimum. There have been some (not completely successful) attempts to characterise what sorts of problems GAs are good at [40, 22]. For our simpler congregational algorithm we can easily state what the key factor in problem difficulty is. GAs have been used for a range of learning problems [34], but so far there has been only one paper on a PAC analysis of a GA for a discrete

learning problem [42]. See also [7, 48].

Often real problems come with a natural continuous parametrisation, such as a feedforward neural network. GAs usually require a binary coding, which leaves open the problem of which mapping to use. (There exist real coded GAs, but these still require a choice of crossover operator, which is usually dependent on some coding scheme for its very definition.) This can make a large difference [6]. Our motivation is that if there is a smooth (differentiable) parametrisation of the target class of functions available, one may as well utilise that. (This is not to say that there is no question concerning choice of parametrisation or that the parametrisation would not affect the performance of a SGD algorithm.) Curiously, although it has been recently shown [41] how mutation driven EAs can be interpreted in certain limits as effectively simulating Newton's algorithm when operated on quadratic cost surfaces, we have been unable to find in the previous literature the simple idea of the algorithm proposed in this paper; namely to run a bunch of SGD algorithms as a population. Most existing random search techniques for optimisation which we are aware of [40, 51, 46, 11, 2, 32, 39] are based on local random perturbations (mutation) to perform the local search. One exception we have found is the "multistart" algorithm described in [51] (pp.24ff). This algorithm works when one has a fully known cost function, rather than just samples of it, and is somewhat different to our algorithm in other ways. Certainly the analysis of our algorithm is different to that presented by Zhigljavsky. Another is "Branin's method" [51] (pp.32–33), which is essentially a deterministic method of escaping from local minima in descent algorithms.

Finally let us simply note that the present work is quite different in style to GA classifier systems as presented in [34] and [29] (chapter 10). Such systems could not directly be put into a form amenable to a congregational SGD.

1.2 What this Paper Shows

This paper formally states the congregational algorithm and derives a result describing its behaviour. We extend the existing averaging theory [44], which is not general enough for our case, to allow us to describe the behaviour of the SGD in terms of an ordinary differential equation (ODE). We then apply a stability result applicable to local minima based on a result in [50]. The stability technique is different to that which sufficed in [10], in order to have it apply about local minima, where one will not necessarily get uniform exponential stability of the associated ODE, but rather just uniform asymptotic stability. (These differences are detailed in the appendix.) We show that, for certain parameter settings, the output of the algorithm will be close to the solution of the ODE, and furthermore the ODE associated with one of the members of the congregation will converge as required, such that the algorithm will, with high probability, produce an estimate in a small ball about the global minimum of the cost function after some finite time.

Since we can describe the behaviour of the algorithm in terms of an associated ODE we can thus talk of the basins of attraction [28] of that ODE. Our subsequent analysis is

couched in such terms. Note that whilst the notion of a basin of attraction has been utilised in the analysis of GAs [48, 31], the very concept is rather more problematical there [33].

We then perform an analysis of the expected amount of computation required by the congregational algorithm. We derive a formula for the expected time to convergence in terms of N and σ, where N is the size of the congregation used and σ is the probability of the initial point of a member of the congregation being chosen to be in the basin of attraction of the global minimum. The expected computational cost will be proportional to N times the expected time to convergence. We then show that for small σ the optimal (in the sense of minimising expected computation) choice of N is $N_{opt} \approx (2/\sigma)^{1/2}$, and more interestingly, that using $N = 2$ will result in an algorithm which will never (in expectation) use more than twice as much computation as one using N_{opt}. This argument is rather different to analyses of GAs that attempt to determine the right population size to use [25].

Finally we apply the algorithm to two examples: blind equalisation of a linear communications channel using the Constant Modulus Algorithm and a simple nonlinear regression problem. We show that our theoretical analysis is well corroborated by our simulations. Some open problems and directions for future work are stated in the conclusions.

2 Notation and Dynamical Systems Theory

For any $a \in \mathbb{R}^m$, $\|a\|$ denotes the Euclidean norm of a and $B(a,r) := \{b \in \mathbb{R}^m : \|b-a\| \leq r\}$ is the closed ball with centre a and radius $r > 0$.

For any function $f : A \times X \to \mathbb{R}$, where $A \subset \mathbb{R}^m$ and $X \subset \mathbb{R}^n$, $\frac{\partial f}{\partial a}$ denotes the gradient of f with respect to the first argument. For $a : \mathbb{R} \to \mathbb{R}^n$, $a : t \mapsto a(t)$, $\dot{a}$ denotes the derivative of a with respect to t.

Definition 2.1 *A function $h : \mathbb{R}^+ \to \mathbb{R}$ is called an* order function *if $h(\mu)$ is continuous and sign definite on $(0, \mu_0]$ for some $\mu_0 > 0$, and if $\lim_{\mu \downarrow 0} h(\mu)$ exists.*

Definition 2.2 *Let $h(\mu)$ and $l(\mu)$ be order functions. Then the notations $O_\mu(l(\mu))$, $o_\mu(l(\mu))$ and $\Omega_\mu(l(\mu))$ are defined by*

1. *$h(\mu) = O_\mu(l(\mu))$ if there exists a constant $L > 0$ such that $|h(\mu)| \leq L|l(\mu)|$ on $(0, \mu_1]$, for some $\mu_1 > 0$.*
2. *$h(\mu) = o_\mu(l(\mu))$ if $\lim_{\mu \downarrow 0} \frac{h(\mu)}{l(\mu)} = 0$.*
3. *$h(\mu) = \Omega_\mu(l(\mu))$ if there exists a constant $L > 0$ such that $|h(\mu)| \geq L|l(\mu)|$ on $(0, \mu_1]$, for some $\mu_1 > 0$.*

Consider the initial value problem

$$\dot{a} = F(a(t)) \qquad ; \qquad a(0) = a_0 \tag{2.1}$$

$t \geq 0$; $a(t) \in \mathbb{R}^m$. Suppose $F(a^*) = 0$ for some $a^* \in \mathbb{R}^m$.

Definition 2.3 *The solution $a \equiv a^*$ of the initial value problem (2.1) is* uniformly asymptotically stable *with basin of attraction $A^* \subset \mathbb{R}^m$ if:*

1. *it is* stable*:*
   for all $\varepsilon > 0$ there exists $\delta > 0$ such that for all $a_0 \in A^*$,

$$\|a_0 - a^*\| \leq \delta \Rightarrow \|a(t) - a^*\| < \varepsilon \qquad \forall t \geq 0.$$

2. *it is* uniformly attractive *in A^*:*
   for all $\delta > 0$ and $\varepsilon > 0$, there exists $\sigma > 0$ such that for all $a_0 \in A^*$,

$$\|a_0 - a^*\| < \delta \Rightarrow \|a(t) - a^*\| < \varepsilon \qquad \forall t \geq \sigma.$$

Definition 2.4 *The ODE (2.1) is* Lagrange stable *if, for all $a_0 \in \mathbb{R}^m$ there exists $\delta \geq 0$ such that*

$$\|a(t)\| \leq \delta \qquad \forall t \geq 0.$$

Definition 2.5 *The ODE (2.1) has an* attractor at infinity *if there exists a set A^∞ such that*

$$a_0 \in A^\infty \Rightarrow \lim_{t \to \infty} \|a(t)\| = \infty.$$

The largest such A^∞ is the basin of attraction of infinity.

If the function F is continuous then (2.1) is Lagrange stable if and only if it does not have an attractor at infinity.

The following theorem is a deterministic averaging result that relates the solution of a difference equation depending on a sequence (x_k) of inputs to the corresponding solution of an *averaged* ordinary differential equation. In essence, it says that if there is a uniformly asymptotically stable critical point of the ODE then solutions of the difference equation originating within the basin of attraction of the critical point converge to a small neighbourhood of the critical point. Thus it is possible to use results about the existence of asymptotically stable critical points of an ODE in order to characterise the behaviour of the solution of a difference equation. In particular, it will be shown that the averaged ODE corresponding to the parameter update equation for the CGD algorithm presented in the next section is a gradient equation. Therefore all (isolated) local minima of the cost function are uniformly asymptotically stable critical points of the ODE.

Assumptions:

A1 $A \subset \mathbb{R}^m$ *and* $X \subset \mathbb{R}^n$, X *is compact, and* $(x_k)_{k\in\mathbb{N}_0}$ *is a sequence of points in* X.

A2 $H : A\times X \to A$ *is bounded and Lipschitz continuous in the first argument (uniformly in the second argument) on a compact domain: There exist functions* $M, \lambda : \mathbb{R} \to \mathbb{R}$ *such that for all* $r > 0$, $\|a\|, \|b\| \le r$ *and* $x \in X$,

$$\|H(a,x)\| \le M(r)$$
$$\|H(a,x) - H(b,x)\| \le \lambda(r)\|a-b\|.$$

A3 The function

$$H^{av}(a) = \lim_{L\to\infty} \frac{1}{L}\sum_{k=0}^{L-1} H(a,x_k)$$

exists uniformly for all $a \in \mathbb{R}^m$. *That is, for any* $L \in \mathbb{N}$, $\delta(\mu) = o_\mu(1)$, *where, for each* $\mu > 0$, $\delta(\mu)$ *is equal to*

$$\sup_{k_0\in\mathbb{N}_0}\sup_{a\in A}\sup_{k\in\left[0,\frac{L}{\mu}\right)} \mu\left\|\sum_{l=k_0}^{k_0+k-1}(H(a,x_l) - H^{av}(a))\right\|.$$

A4 $\beta : \mathbb{R}^+ \to \mathbb{R}$ *satisfies* $\beta(\mu) = o_\mu(1)$.

A5 For each $k \in \mathbb{N}_0$, $h_k : A \times X \to A$ *is bounded and Lipschitz continuous in the first argument (uniformly in the second argument and in* k*) on a compact domain.*

Theorem 2.6 *With Assumptions A1 to A5, let* a_k *and* $a_{av}(t)$ *be defined according to the following equations for all* $k \in \mathbb{N}_0$ *and* $t \ge 0$*:*

$$a_{k+1} = a_k - \mu H(a_k,x_k) - \mu\beta(\mu)h_k(a_k,x_k) \qquad ; \qquad a_0 \in A \tag{2.2}$$
$$\dot{a}_{av} = -\mu H^{av}(a_{av}(t)) \qquad ; \qquad a_{av}(0) = a_0 \tag{2.3}$$

Assume $a^* \in$ *interior of* A *is a uniformly asymptotically stable critical point of equation 2.3, with basin of attraction* $A^0 \subset A$. *Then for any compact set* $B^0 \subset A^0$ *there exists an* $o_\mu(1)$ *function* $l(\mu)$ *and a constant* $\mu_0 > 0$ *such that if* $\mu \le \mu_0$, *then there exists* $k_\mu \in \mathbb{N}_0$ *such that if* $a_0 \in B^0$ *then*

$$\|a_k - a^*\| \le l(\mu) \qquad \forall k \ge k_\mu.$$

Theorem 2.6 is proved in Appendix A.

3 The Congregational Gradient Descent Algorithm

This paper addresses the problem of locating the global minimum of some cost function $J : A \to \mathbb{R}$, where $A \subset \mathbb{R}^m$. The cost function is not known explicitly, but rather it is the average of a known function $\phi : A \times X \to \mathbb{R}$ over a known sequence (x_k) of points in X. That is,

$$J(a) = \lim_{K\to\infty} \frac{1}{K} \sum_{k=0}^{K-1} \phi(a, x_k). \tag{3.1}$$

We say that $\phi(a, x_k)$ is the instantaneous cost at time k. Points $a \in A$ are called *parameters* and points $x_k \in X \subset \mathbb{R}^n$ are called *inputs*. The inputs are received sequentially, and it is desired to have a parameter estimate which is updated as each input is received.

Stepwise gradient descent of J is achieved by updating estimate parameters a_k according to

$$a_{k+1} = a_k - \mu \left.\frac{\partial \phi}{\partial a}\right|_{(a_k, x_k)}. \tag{3.2}$$

From Theorem 2.6, it can be shown that the estimate parameters generated by this recursion will converge to a neighbourhood of the global minimiser of J provided that the initial parameter estimate is in a certain region of parameter space. As $\mu \to 0$, this region approaches the basin of attraction of the global minimiser in the associated averaged ODE. However, if there are non-global local minima of J, some choices of the initial estimate will cause the estimate parameters to converge to a local minimiser. In general the basin of attraction of the global minimum is not known, so SGD cannot be guaranteed to find the global minimum.

The CGD algorithm is a modification of SGD which gets around the problem of local minima. It is perhaps the simplest possible globally convergent populational algorithm for online minimisation. Instead of choosing one initial parameter estimate and updating it as each input is received, a number of estimates with randomly chosen initial values are run in parallel. At the same time, an estimate of the cost function at each of the parameter estimates is calculated. Periodically, the estimates are compared and all but the best are restarted according to some continuous probability distribution D_a with compact support $A^0 \subset \mathbb{R}^n$. The time between restarts is called an *epoch*.

A similar non-populational modification of SGD would be to run a number of SGD estimates serially. Again, an online cost estimate could be kept, and at the end of the epoch the parameter estimate could be kept only if the estimate cost is better than the estimate cost for all previous parameter estimates. The congregational algorithm requires slightly more computation than this serial algorithm, because we continue to update the best estimate through all epochs. However, it requires considerably less input than the serial algorithm, because all N members in the congregation are updated using the same inputs. Moreover, the continued updating of the best estimate allows the estimate to continue to improve, which can be useful in cases where the cost at the global minimum is much better than at all local minima.

The CGD Algorithm

Choose the cost stepsize $\alpha \in (0,1)$*;*
Choose the parameter stepsize $\mu > 0$*;*
Choose the epoch length $K > 0$*;*
Choose the congregation size $N \geq 2$*;*

for $n \in \{1, \ldots, N\}$ **do**
  $a^n_{0,1} := random(A)$*;*
  $\Phi^n_{0,1} := 0$*;*
od
$T := 1$*;*
while *(true)* **do**
  for $k = 0$ **to** $k = K - 1$ **do**
    for $n \in \{1, \ldots, N\}$ **do**

$$a^n_{k+1,T} := a^n_{k,T} - \mu \left. \frac{\partial \phi}{\partial a} \right|_{(a^n_{k,T}, x_{(T-1)K+k})}; \tag{3.3}$$

$$\Phi^n_{k+1,T} := (1-\alpha)\Phi^n_{k,T} + \alpha\phi(a^n_{k,T}, x_{(T-1)K+k}); \tag{3.4}$$

    od
  od
  $\hat{n} := \arg\min_{n \in \{1,\ldots,N\}} \Phi^n_{K,T}$*;*
  $a^1_{0,T+1} := a^{\hat{n}}_{K,T}$*;*
  $\Phi^1_{0,T+1} := \Phi^{\hat{n}}_{K,T}$*;*
  $T := T + 1$*;*
  for $n \in \{2, \ldots, N\}$ **do**
    $a^n_{0,T} := random(A)$*;*
    $\Phi^n_{0,T} := 0$*;*
  od
od

The function $random(A)$ generates independent and identically distributed random variables according to some fixed distribution D_a which has compact support $A^0 \subset A$.

At time k in epoch T, member n of the congregation takes on the value $a^n_{k,T}$. In Section 5, is shown that, as T increases, the probability that $a^1_{0,T}$ is close to the global minimum of J is bounded below by a quantity that depends on various parameters of the problem.

The subscript $(T-1)K + k$ on the samples in equations 3.3 and 3.4 ensures that the algorithm is online, in that each update is made according to a new sample. This is not necessary—the algorithm also works if the same set of samples is used for each epoch. However if it is possible to store and reuse the samples, there is less point in using an online algorithm.

Equation 3.4 defines an online estimate $\Phi^n_{k,T}$ of the average cost at $a^n_{k,T}$. The online estimate is a weighted average of all instantaneous cost estimates since the beginning of

the epoch. It can be written

$$\Phi^n_{k,T} = \alpha \sum_{j=0}^{k-1} (1-\alpha)^{k-j+1} \phi(a^n_{j,T}, x_{(T-1)K+j}).$$

The weighting causes the instantaneous cost at the beginning of the epoch to have less effect than the instantaneous cost at the end of the epoch. As $\alpha \to 0$, the cost estimate updates slower, so more averaging occurs. However, this also implies that the effect of the changing parameter estimate is larger. As $\alpha \to 1$ the cost estimate more closely resembles the instantaneous cost. As the cost estimate is only used at the end of the epoch for testing fitness of the members, it is required that the estimate cost at the end of the epoch is close to the average cost for the final value of the estimate parameter. In Lemma 5.3 it is shown that $\Phi^n_{K,T}$ can be made arbitrarily close to $J(a^n_{K,T})$ by choosing α and μ sufficiently small and K sufficiently large.

The connection between the recursions defined in (3.3) and (3.4) can be viewed in a number of ways, depending on the relationship between the small parameters μ and α. It has been shown [30] that the rate of convergence of online estimates such as these decreases as the dimension of the estimate increases. Since the parameter estimate is an m dimensional vector and the cost estimate is a scalar, the parameter estimate can be expected to converge more slowly than the cost estimate. This would seem to indicate that μ should be chosen larger than α. In this case α is small, so the cost estimate averages over the trajectory of the estimate parameters. On the other hand, since the cost estimate is only used at the end of an epoch, μ could be chosen smaller than α, so the cost estimate is more closely related to the current value of the estimate parameter.

Choosing $\mu = o_\alpha(\alpha)$ or $\alpha = o_\mu(\mu)$ would lead to an averaging analysis using *split time scales*, such as appears in [5]. Split time scales are not used in the analysis that appears in this paper—it is assumed only that $\mu = o_\alpha(1)$. Furthermore, in the simulation results in Section 7, μ and α are chosen to be identical. This is possible because the fact that the average ODE for the estimate parameters is a gradient system enables the application of infinite horizon averaging result in Theorem 2.6. Thus the epoch length is chosen long enough that the estimate parameters converge to local minima and sit there, and the cost estimates converge to the average cost near the corresponding local minima. However the use of split time scales would be necessary if the online cost estimate were to be used before the parameters have converged to local minima.

4 Analysis for a Simpler Case

In this section, the probability of convergence of a simplified version of the CGD algorithm calculated. The simplification takes the form of the following assumptions, which are not valid in the online optimisation context of the problem, but which do not change the overall behaviour of the algorithm significantly.

1. The parameters are updated according to continuous time gradient descent on the average cost function J;

2. All estimates converge to critical points of J by the end of each epoch;

3. The exact value of the average error function J, rather than its estimate Φ, is used for testing fitness at the end of the epochs.

When these simplifications are made, so that the estimates are defined in continuous time, the notation $a^n_{k,T}$ is misleading. However the notation is still used, because changing the notation would require formally defining the simplified algorithm, and distract from the essential ideas expressed in this section. Note that the second simplification is not valid even for solutions of (4.1), since the difference between solutions of (4.1) and the critical points of J decays exponentially.

Here and in the rest of this paper the probabilities are with respect to the randomly chosen initial estimates $a^n_{0,T}$, where either $n > 1$ or $T \neq 1$. For any event E which depends on the value of the initial estimates, the probability that E occurs is written $Pr\{E\}$. Occasionally in Section 5 the same probability is written $Pr\{E \text{ where } a \sim D_a\}$. This second notation is redundant, but is used anyway to reduce the confusion caused by the other complicated notation that is necessary. For instance, it is simpler to interpret "$a^n_{0,T} \sim D_a$" than "$a^n_{0,T}$ where either $n > 1$ or $T \neq 1$".

Let a^* be the global minimiser of J, and let A^* be the basin of attraction for a^* in

$$\dot{a} = -\left.\frac{\partial J}{\partial a}\right|_{a(t)} \tag{4.1}$$

and let $A^0(a^*) := A^* \cap A^0$, where A^0 is the set of all possible initial estimates. Let σ be the probability of initialising a member of the congregation in the basin of attraction of the global minimum. That is,

$$\sigma := Pr\{a \in A^* \text{ where } a \sim D_a\} = \int_{A^0(a*)} dD_a. \tag{4.2}$$

Except for trivial cases σ is an unknown quantity. It may be seen as providing a measure of the difficulty of the task of finding the global minimum. This crucial parameter appears in all of the results of this paper. In Section 7 a method for estimating σ from simulation curves is demonstrated.

The probability of convergence of the simplified algorithm by the end of the epoch is derived as follows. The numbered steps in the derivation form the basis of the proof of Theorem 5.1 in the next section.

Step 1. Since the average ODE is used to determine the estimate parameters, and all estimates converge to critical points by the end of the epoch, if member n is initialised in A^* at the beginning of epoch t, then at the end of the epoch t member n is equal to a^*. If member n is initialised outside A^* at the beginning of epoch t then at the end of epoch t member n does not equal a^*.

Step 2. At the end of an epoch estimates can be divided into "good" estimates, for which $a^n_{K,T} = a^*$, and "bad" estimates, for which $a^n_{K,T} \neq a^*$. Since a^* is the global minimiser of J, the average cost at all bad estimates is larger than the average cost at good estimates.

Step 3. Since the exact value of the average cost J is used for testing if a good estimate exists at the end of an epoch, the estimate that is chosen to be kept at the end of the epoch will be a good estimate.

Step 4. Using steps 1 and 3, the probability that $a^1_{0,2} = a^*$ is equal to the probability of choosing at least one initial estimate in A^*, i.e.

$$Pr\{a^1_{0,2} = a^*\} = 1 - (1-\sigma)^N. \tag{4.3}$$

Step 5. For later epochs, only $N-1$ of the members are restarted. The first member of the population does not move from the critical point that it converged to by the end of the previous epoch. Therefore the probability that $a^1_{0,t+1} = a^*$ is equal to the probability that one of the new members is initialised in A^* plus the probability that none of them are, but the member that was carried over from the previous epoch was equal to a^*.

$$Pr\{a^1_{0,t+1} = a^*\} = 1 - (1-\sigma)^{N-1} + (1-\sigma)^N Pr\{a^1_{0,t} = a^*\}. \tag{4.4}$$

Step 6. The recursive relationship (4.4) yields

$$\begin{aligned} Pr\{a^1_{0,T} = a^*\} = &\left[1 - (1-\sigma)^{N-1}\right] \sum_{i=0}^{T-2} (1-\sigma)^{(N-1)i} \\ &+ (1-\sigma)^{(N-1)(T-1)} Pr\{a^1_{0,2} \in B(a^*, r)\} \end{aligned} \tag{4.5}$$

Use of the geometric sum and (4.3) shows that

$$\begin{aligned} Pr\{a^1_{0,T} = a^*\} &= \left[1 - (1-\sigma)^{N-1}\right] \frac{1 - (1-\sigma)^{(N-1)(T-1)}}{1 - (1-\sigma)^{N-1}} \\ &\quad + (1-\sigma)^{(N-1)(T-1)} Pr\{a^1_{0,2} \in B(a^*, r)\} \\ &= 1 - (1-\sigma)^{N+(N-1)(T-1)}. \end{aligned} \tag{4.6}$$

In the following section, analagous steps will be taken in order to establish the probability of convergence of the CGD algorithm. Without the simplifications mentioned above, the results of steps 1, 2 and 3 can not apply. However, somewhat weaker results can be derived with some extra work.

5 Convergence Analysis

For any $r > 0$ such that $B(a^*, r) \subset A^0(a^*)$, the CGD algorithm is said to have *converged after epoch T (to accuracy r)* if the best estimate is no further than r from the global

minimiser of J (i.e. $a^1_{0,T+1} \in B(a^*, r)$). Clearly r has to be chosen small enough for such a definition to be of value.

In Theorem 5.1, the following assumptions are used to show that the probability that the algorithm has converged to accuracy r after epoch T is greater than or equal to a function which is monotonically increasing with T. In the process, it is shown that there exists an $o_\mu(1)$ function $l(\mu)$ such that it is possible to let $r = l(\mu)$. Thus when the algorithm converges, the estimate parameters will be very close to the global minimiser if μ is very small.

Assumptions:

C1 $A^0 \subset A \subset \mathbb{R}^m$, $X \subset \mathbb{R}^n$, A^0 *and* X *are compact, and* $(x_k)_{k\in\mathbb{N}_0}$ *is a sequence of points in* X.

C2 *For* $T = 1$ *and* $n \in \{1, \ldots, N\}$, *or* $T \in \mathbb{N}$, $T > 1$ *and* $n \in \{2, \ldots, N\}$, *the initial estimates* $a^n_{0,T} \in A^0$ *are i.i.d. random variables distributed according to a continuous probability distribution* D_a *with support* A^0.

C3 *Both* $\phi(a, x)$ *and* $\frac{\partial \phi}{\partial a}$ *are bounded and Lipschitz continuous in the first parameter (uniformly in the second) on a compact domain in* $\mathbb{R}^m$.

C4 *The average* J *defined in (3.1) exists, and for any* $L \in \mathbb{N}$ *(independent of* μ*),*

$$\delta(\mu) = \sup_{k_0\in\mathbb{N}_0} \sup_{a\in A} \sup_{k\in[0,\frac{L}{\mu})} \mu \left\| \sum_{l=k_0}^{k_0+k-1} (\phi(a, x_l) - J(a)) \right\|$$

exists and is $o_\mu(1)$.

C5 J *has a (unique) global minimum at some point* a^* *in the interior of* A^0. *Furthermore,* J *has a finite number of local minima which have basins of attraction intersecting* A, *and (4.1) is Lagrange stable.*

C6 $\mu = o_\alpha(1)$.

Assumption *C4* implies that the average cost function converges uniformly to J with respect to the initial tie k_0. This implies that the error introduced by the use of the instantaneous cost rather than the average cost in the CGD algorithm updates can be bounded independently of the epoch under consideration.

Assumption *C5* requires that there is a unique global minimum of the cost function (or only one global minimum for which the basin of attraction intersects A), and that (4.1) is Lagrange stable. These assumptions are not necessary, but have been included in order to streamline the notation. The assumption that (4.1) is Lagrange stable is discussed further at the end of the section. The assumption that J has a finite number of local minima precludes the existence of an attracting manifold in the parameter space. It can

be interpreted as including a *persistence of excitation* condition. For example, assume ϕ is the output error squared for a linear system: $\phi(a,x) = \left((a-a^*)^\top x\right)^2$. If the input (x_k) does not span $\mathbb{R}^n$, $\phi(\cdot, x_k)$ will have a unique local minimum for each value of x_k, but J will not have a unique local minimum. Instead, there will be a line of points in $\mathbb{R}^n$, passing through a^*, which are all global minimisers of J.

The choice of $\mu = o_\alpha(1)$ is used in Theorem 5.3 in order to ensure that the estimate parameters converge to an $o_\alpha(1)$ neighbourhood of the local minimisers. Once the estimate parameters have converged, the averaging result in Theorem 2.6 is used with small parameter α. For sufficiently large k and K, the difference between the instantaneous cost at the estimate $a^n_{K,T}$ and at $a^n_{k,T}$ is a second order effect, so it can be dismissed.

Theorem 5.1 *Consider the CGD algorithm with Assumptions C1 to C6. There exists $r_0 > 0$ such that for all $0 < r \le r_0$, $\gamma \in (0,1)$ there exists $\alpha_0 > 0$ such that if $\alpha \le \alpha_0$ then there exists $K_0(\alpha) \in \mathbb{N}$ such that if $K \ge K_0(\alpha)$ then, for all $T \in \mathbb{N}$, the probability that the estimate that is kept at the end of the T-th epoch is no further than r from the global minimiser satisfies*

$$
\begin{aligned}
&Pr\{\|a^1_{0,T+1} - a^*\| \le r\} \\
&\quad \ge (1-\gamma)^{N-1}\left[1 - \left(\frac{1-\sigma}{1-\sigma\gamma}\right)^N\right]\left[\frac{(1-\gamma)^2(1-\sigma)}{1-\sigma\gamma}\right]^{(N-1)(T-1)} \\
&\qquad + \frac{(1-\gamma)^{N-1}\left[(1-\sigma\gamma)^{N-1} - (1-\sigma)^{N-1}\right]}{(1-\sigma\gamma)^{N-1} - (1-\gamma)^{2(N-1)}(1-\sigma)^{N-1}}\left(1 - \left[\frac{(1-\gamma)^2(1-\sigma)}{1-\sigma\gamma}\right]^{(N-1)(T-1)}\right).
\end{aligned}
$$

Theorem 5.1 does not provide a practical method for choosing the quantities α, μ, and K required for application of the algorithm. However, it does prove that suitable quantities exist.

In the limit as $\gamma \to 0$, the lower bound in Theorem 5.1 is equal to the probability (4.6) derived in Section 4. The parameter γ arises from the discrete nature of the algorithm. In order to make γ close to 0, μ and α must be allowed to approach zero and K must be allowed to approach infinity. The relationship between γ, μ, α and K is not simple, and is constrained explicitly in the first step of the proof. In Appendix B it is shown that,for $\gamma \in (0,1)$, the lower bound in Theorem 5.1 is less than the probability in (4.6). Thus the lower bound on the probability of convergence is weaker here than the corresponding result for the simplified version described in Section 4, as would be expected.

The proof of Theorem 5.1 is given at the end of this section. First two technical lemmas are derived using Theorem 2.6.

Assume either $n > 1$ or $T > 1$, so that $a^n_{0,T} \sim D_a$. For any $r > 0$ such that $B(a^*, r) \subset A^0(a^*)$, let

$$
p(K,T,r) := Pr\{a^n_{K,T} \in B(a^*, r) \text{ where } a^n_{0,T} \sim D_a\}, \tag{5.1}
$$

where $a^n_{k,T}$ is defined according to (3.3). In addition, define

$$J^{loc} := \min\{J(a) : J(a) \text{ is a non-global local minimum of } J\} \tag{5.2}$$

$$q(K,T) := Pr\{J(a^n_{K,T}) \geq J^{loc} \text{ where } a^n_{0,T} \sim D_a\}. \tag{5.3}$$

Then $p(K,T,r)$ is the probability that a newly initialised estimate converges to accuracy r by the end of the T-th epoch, and $q(K,T)$ can be regarded as the probability that the estimate converges to some other local minimum. Both p and q are independent of n because all members are initialised according to the same distribution D_a and are updated according to (3.3). Since $B(a^*,r) \subset A^0(a^*)$, the events defined in (5.1) and (5.3) are mutually exclusive, so $q(K,T) \geq 1 - p(K,T,r)$ and $p(K,T,r) \geq 1 - q(K,T)$. If there are no non-global local minima of J, any value of J^{loc} satisfying $J^{loc} > J(a^*)$ can be used. In the following lemma it is shown that α and K can be chosen in order to make p arbitrarily close to σ and q arbitrarily close to $1-\sigma$.

Lemma 5.2 *With Assumptions C1 to C6, let $a^n_{k,T}$ be defined according to the CGD algorithm. If $a^n_{0,T} \sim D_a$ then for all $r > 0$ such that $B(a^*,r) \subset A^0(a^*)$, and all $\eta \in (0,1)$, there exists $\alpha_r > 0$ such that if $\alpha \leq \alpha_r$ then there exists $K_r(\alpha)$ such that if $K \geq K_r(\alpha)$, for all $T \in \mathbb{N}$*

$$(1-\eta)\sigma \leq p(K,T,r) \leq (1-\eta)\sigma + \eta \tag{5.4}$$

$$(1-\eta)(1-\sigma) \leq q(K,T) \leq (1-\eta)(1-\sigma) + \eta. \tag{5.5}$$

If $a^n_{0,T} \in B(a^*,r)$ then α_r and $K_r(\alpha)$ can be found such that, for all $T \in \mathbb{N}$,

$$a^n_{0,T} \in B(a^*,r) \Rightarrow a^n_{K,T} \in B(a^*,r).$$

The bounds α_r and $K_r(\alpha)$ depend on the particular sequence (x_k), the distribution of the local minima, and the boundaries of the basins of attraction.

Proof In the following, the lower bounds on $p(K,T,r)$ and $q(K,T)$ are determined. The upper bounds follow from the fact that the events defined in (5.1) and (5.3) are mutually exclusive.

As before, the average equation (4.1) is a gradient equation. Since the local minima of J are isolated points, they are uniformly asymptotically stable critical points of (4.1). For any local minimiser a^{loc} of J, let $A^0(a^{loc})$ denote the intersection of the associated basin of attraction with A^0. For all a^{loc} such that $A^0(a^{loc}) \neq \emptyset$, it is possible to choose compact sets $B^0(a^{loc}) \subset A^0(a^{loc})$ such that:

- $B^0(a^*) \supset B(a^*,r)$;
- For all a^{loc}, $B^0(a^{loc})$ contains an open neighbourhood of a^{loc};

- $Pr\{a \in B^0(a^*) \text{ where } a \sim D_a\} \geq (1-\eta)\sigma$;
- $Pr\{a \in B(loc) \text{ where } a \sim D_a\} \geq (1-\eta)(1-\sigma)$, where $B(loc) := \bigcup_{a^{loc} \neq a^*} B^0(a^{loc})$.

Consider Theorem 2.6, with the function $H(a,x)$ identified with $\frac{\partial \phi}{\partial a}\Big|_{(a,x)}$ and $h_k(\cdot,\cdot) = 0$ for all k. The assumptions of Theorem 2.6 are satisfied—in particular, Assumptions *C3* and *C4* imply Assumptions *A2* and *A3*. Therefore for any a^{loc} there exists an $o_\mu(1)$ function $l^{loc}(\mu)$ and a constant $\mu_0^{loc} > 0$ such that if $\mu \leq \mu_0^{loc}$ then the solution of (4.1) with initial condition $a_{0,T}^n \in B^0(a^{loc})$ enters and remains in a ball centred at a^{loc} with radius $l^{loc}(\mu)$. Let $\mu_0 = \min_{\text{all local minima of } J} \mu_0^{loc}$ and for all $\mu \leq \mu_0$ let $l(\mu) = \min_{\text{all local minima of } J} l^{loc}(\mu)$. These minima exist since there is a finite number of local minima of J.

Choose $\mu_r \leq \mu_0$ such that $l(\mu) \leq r$ for all $\mu \leq \mu_r$. Theorem 2.6 says that for all $\mu \leq \mu_r$, there exists $K_r(\alpha) \in \mathbb{N}_0$ such that if $a_0 \in B^0(a^*)$ then $a_k \in B(a^*, r)$ for all $k \geq K_r(\alpha)$. Since $\mu = o_\alpha(1)$, there exists α_r such that $\mu \leq \mu_r$ whenever $\alpha \leq \alpha_r$. For any $\alpha \leq \alpha_r$, if $K \geq K_r(\alpha)$, let

$$B^*(K,T,r) := \{a_{0,T}^n \in A^0 : a_{K,T}^n \in B(a^*, r)\}. \tag{5.6}$$

Then $B^0(a^*) \subset B^*(K,T,r)$ for all $T \in \mathbb{N}$, so comparing with the definition of $p(K,T,r)$ shows

$$\begin{aligned} p(K,T,r) =& Pr\{a_{0,T}^n \in B^*(K,T,r) \text{ where } a_{0,T}^n \sim D_a\} \\ \geq& Pr\{a_{0,T}^n \in B^0(a^*)\} \\ \geq& (1-\eta)\sigma \end{aligned}$$

from the choice of $B^0(a^*)$.

For the second result, choose $\mu_1 \leq \mu_0$ such that $B(a^{loc}, l(\mu)) \subset A^0$ for all $\mu \leq \mu_1$ and a^{loc}. Then Theorem 2.6 implies that for all $\mu \leq \mu_1$, there exists $K_r(\alpha) \in \mathbb{N}_0$ such that if $a_{0,T}^n \in B^0(a^{loc})$ then $a_{k,T}^n \in A^0$ for all $k \geq K_r(\alpha)$. Now $a \in A^0(a^{loc})$ implies $J(a) \geq J^{loc}$, so

$$q(K,T) \geq Pr\left\{a_{0,T}^n \in B(loc)\right\} \geq (1-\eta)(1-\sigma).$$

The last claim follows similarly, since $B(a^*, r) \subset B^0(a^*) \subset B^*(K,T,r)$. ∎

Lemma 5.3 *Consider the CGD algorithm with Assumptions C1 to C6. For any $\varepsilon > 0$, with probability 1, there exists α_ε such that if $\alpha \leq \alpha_\varepsilon$ there exists $K_\varepsilon(\alpha)$ such that if $K \geq K_\varepsilon(\alpha)$ then*

$$|\Phi_{K,T}^n - J(a_{K,T}^n)| \leq \varepsilon \tag{5.7}$$

for all $T \in \mathbb{N}$ and $n \in \{1, \ldots, N\}$.

Proof Let $\varepsilon > 0$ and choose $T \in \mathbb{N}$, and $n \in \{1, \ldots, N\}$. The parameter estimate $a^n_{K,T}$ is the solution of (3.3) randomly chosen according to D_a, evolved for at least K time steps. The union of the basins of attraction of the local minima are dense in A, so, by Assumption *C2*, with probability 1 the initial condition is contained in the basin of attraction of some local minimiser $a^{loc} \in A$.

Consider Theorem 2.6, with the function $H(a, x)$ identified with $\frac{\partial \phi}{\partial a}\Big|_{(a,x)}$ and $h_k(\cdot, \cdot) = 0$ for all k. Theorem 2.6 applies, so there exists $\mu_{0^n} > 0$ such that for all $\mu \leq \mu^n_{0,T}$, there exists $K^0_{n,T}(\mu)$ such that for all $k \geq K^0_{n,T}(\mu)$,

$$\|a^n_{k,T} - a^{loc}\| \leq l(\mu), \tag{5.8}$$

where $l(\mu)$ is some $o_\mu(1)$ function. Since $\mu = o_\alpha(1)$, there exists $\alpha^0_{n,T}$ such that $\mu \leq \mu^0_{n,T}$ whenever $\alpha \leq \alpha^0_{n,T}$, and $l(\mu) = o_\alpha(1)$.

Equation 3.4 can be written

$$\Phi^n_{k+1,T} = \Phi^n_{k,T} - \alpha \left(\Phi^n_{k,T} - \phi(a^n_{k,T}, x_{(T-1)K+k}) \right). \tag{5.9}$$

This can be put into the framework of Theorem 2.6 by using the small parameter α instead of μ, and identifying a_k in Theorem 2.6 with $\Phi^n_{k,T}$ here. Identify $H(\Phi, x)$ with $\Phi - \phi(a^n_{K,T}, x)$, $h_k(\Phi, x)$ with $\frac{\phi(a^n_{K,T}, x) - \phi(a^n_{k,T})}{l(\mu)}$, and $\beta(\alpha)$ with $l(\mu(\alpha)) = o_\alpha(1)$. The averaged ODE associated with (5.9) is

$$\dot{\Phi} = -\alpha(\Phi - J(a^n_{K,T})). \tag{5.10}$$

The ODE (5.10) has a globally uniformly asymptotically stable critical point $J(a^n_{K,T})$. Moreover, H is bounded and Lipschitz continuous in its first argument (uniformly in the second argument) on a compact domain.

Since ϕ is Lipschitz continuous, there exists a constant $\lambda_\phi > 0$ such that $|h_k(\Phi, x)| \leq \frac{\lambda_\phi \|a^n_{K,T} - a^n_{k,T}\|}{l(\mu)} \leq 2\lambda_\phi$ if $k, K \geq K^0_{n,T}(\mu)$, using (5.8). Theorem 2.6 applies, with small parameter α and initial condition $\Phi^n_{K^0_{n,T}(\mu),T}$ at time $K^0_{n,T}(\mu) \geq 0$. Thus there exists $\alpha^1_{n,T}$ such that if $\alpha \leq \alpha^1_n$ there exists $K^1_{n,T}(\alpha)$ such that if $k \geq K^0_{n,T}(\alpha) + K^1_{n,T}(\mu)$ then (5.7) holds.

Now let the bounds α_ε and $K_\varepsilon(\alpha)$ be given by $\alpha_\varepsilon = \sup_{T \in \mathbb{N}} \min_{n \in \{1,\ldots,N\}} \{\alpha^0_{n,T}, \alpha^1_{n,T}\}$, and, for all $\alpha \leq \alpha_\varepsilon$, $K_\varepsilon(\alpha) = \sup_{T \in \mathbb{N}} \max_{n \in \{1,\ldots,N\}} \{K^0_{n,T}(\mu(\alpha)) + K^1_{n,T}(\alpha)\}$. In both cases the existence of the supremum is guaranteed by the the assumption in *C4* that the short term average converges uniformly to J. ∎

Proof Sketch for Theorem 5.1 The proof of Theorem 5.1 is conceptually the same as the derivation in Section 4. The steps involved in the proof are outlined below.

Step 1. It is shown that Lemmas 5.2 and 5.3 apply for particular choices of η and ε. The bounds α_0 and $K_0(\alpha)$ that appear in the theorem statement are defined by taking the

minimum and maximum, respectively, of the corresponding bounds that appear in the lemmas.

Step 2. Lemma 5.2 is used to bound the probability that, at the end of an epoch, the difference between the average cost at a good estimate (an estimate in the ball $B(a^*, r)$) and the average cost at a "bad" estimate (an estimate that is not in the ball $B(a^*, r)$) is more than

$$D := \frac{J(a^*) - J^{loc}}{2}. \tag{5.11}$$

$D > 0$ by (5.2). This probability is not equal to 1 because the behaviour of estimates which start on (or arbitrarily close to) the boundaries of the basins of attraction of the various minima is not known. In particular, the estimate may be very close to, but not in, the ball $B(a^*, r)$.

Step 3. The bound derived in step 2 is combined with the result of Lemma 5.3 to bound the probability of keeping a good estimate at the end of the epoch, given that one exists.

Step 4. A lower bound on the probability that the algorithm has converged at the end of the first epoch is calculated, using the bound found in 3 and the fact that all members are randomly initialised at the beginning of the first epoch.

Step 5. For any $T > 1$, the final result from Lemma 5.2 is used to show how the probability that the algorithm has converged at the end of the T-th epoch depends on the probability that the algorithm has converged at the end of the $T-1$-th epoch. This uses the bound found in 3 and the fact that only $N-1$ members are randomly initialised at the beginning of the epoch.

Step 6. The recursive relationship derived in step 5 is combined with the bound in step 4 and a closed form expression for the lower bound on the probability that the algorithm has converged at the end of the T-th epoch is derived. ∎

Proof of Theorem 5.1

Step 1. Let r_0 be sufficiently small that $J(a) \leq J(a^*) + D$ for all $a \in B(a^*, r_0)$, where D is defined in (5.11). Then $B(a^*, r_0) \subset A^*$, so any $r < r_0$ is sufficiently small for Lemma 5.2. Let

$$\eta = \frac{(1-\sigma)\gamma}{1-\sigma\gamma}. \tag{5.12}$$

Since $\gamma,\ \sigma \in (0,1)$, it follows that $\eta \in (0,1)$i, so Lemma 5.2 applies. Letting $\varepsilon = \frac{D}{2}$, Lemma 5.3 applies. Define $\alpha_0 = \min\{\alpha_r, \alpha_\varepsilon\}$ and for any $\alpha \leq \alpha_0$ let $K_0(\alpha) = \max\{K_r(\alpha), K_\varepsilon(\alpha)\}$, where $\alpha_r,\ K_r(\alpha)$ are determined by Lemma 5.2 and $\alpha_\varepsilon,\ K_\varepsilon(\alpha)$ are

determined by Lemma 5.3. Then if $\alpha \leq \alpha_0$, $K \geq K_0(\alpha)$ and $r \leq r_0$, for all $T \in \mathbb{N}$,

$$\frac{(1-\gamma)\sigma}{1-\sigma\gamma} \;\leq\; p(K,T,r) \;\leq\; \frac{(1-\gamma)\sigma + (1-\sigma)\gamma}{1-\sigma\gamma} \tag{5.13}$$

$$\frac{(1-\gamma)(1-\sigma)}{1-\sigma\gamma} \;\leq\; q(K,T) \quad\leq\; \frac{(1-\sigma)}{1-\sigma\gamma} \tag{5.14}$$

$$a^n_{0,T} \in B(a^*,r) \Rightarrow a^n_{K,T} \in B(a^*,r) \tag{5.15}$$

$$\|\Phi^n_{K,T} - J(a^n_{K,T})| \leq \frac{D}{2} \qquad \forall n \in \{1,\ldots N\}. \tag{5.16}$$

Equations 5.13 and 5.14 are derived from equations 5.4 and 5.5, using the definition of η in equation 5.12. The above facts are used throughout the rest of the proof.

Step 2. Let $r \leq r_0$. Let a^n be a good estimate and a^m be a bad estimate, in the sense that $a^n \in B(a^*,r)$, and $J(a^m) \geq J^{loc}$. By the definition of r_0, $J(a^n) \leq J(a^*) + D \leq J(a^m) - D$. That is, the cost at a bad estimate is at least D larger than the cost at a good estimate. Now assume that a^m is not good (but not necessarily bad). That is $a^m \notin B(a^*,r)$. The probability that this D separation between the costs still exists is

$$\begin{aligned} Pr\left\{J(a^n) < J(a^m) - D \text{ given } \left(a^n \in B(a^*,r) \text{ and } a^m \notin B(a^*,r)\right)\right\} \\ \geq Pr\{J(a^m) \geq J(a^*) + 2D \text{ given } a^m \notin B(a^*,r)\} \\ = \frac{Pr\{J(a^m) \geq J^{loc} \text{ and } a^m \notin B(a^*,r)\}}{Pr\{a^m \notin B(a^*,r)\}}, \end{aligned}$$

using the definition of D

$$= \frac{Pr\{J(a^m) \geq J^{loc}\}}{Pr\{a^m \notin B(a^*,r)\}}$$

since $a^m \notin B(a^*,r)$ whenever $J(a^m) \geq J^{loc}$. Now assume that a^n and a^m correspond to estimates at the end of an epoch. From the definitions of p and q in (5.1) and (5.3), we have

$$\begin{aligned} Pr\left\{J(a^n_{K,T}) < J(a^m_{K,T}) - D \text{ given } \left(a^n_{K,T} \in B(a^*,r) \text{ and } a^m_{K,T} \notin B(a^*,r)\right)\right\} \\ \geq \frac{q(K,T)}{1-p(K,T,r)}, \\ \geq (1-\gamma), \end{aligned} \tag{5.17}$$

using the lower bounds in equations 5.13 and 5.14.

Step 3. The probability of keeping a good estimate at the end of the T-th epoch, given

that a good estimate exists, is

$$
\begin{aligned}
&Pr\{a^1_{0,T+1} \in B(a^*,r) \text{ given } a^n_{K,T} \in B(a^*,r) \text{ for some } n\} \\
&\quad = Pr\{\Phi^n_{K,T} < \Phi^m_{K,T} \text{ for all } m \text{ such that } a^m_{K,T} \notin B(a^*,r) \text{ given } a^n_{K,T} \in B(a^*,r)\} \\
&\quad \geq Pr\Big\{ J(a^n_{K,T}) < J(a^m_{K,T}) - D \text{ for all } m \text{ such that } a^m_{K,T} \notin B(a^*,r) \\
&\qquad\qquad \text{given } a^n_{K,T} \in B(a^*,r)\Big\},
\end{aligned}
$$

using (5.16)

$$
\begin{aligned}
&\geq Pr\Big\{ J(a^n_{K,T}) < J(a^m_{K,T}) - D \text{ for all } m \neq 1 \\
&\qquad\qquad \text{given } (a^m_{K,T} \notin B(a^*,r) \text{ and } a^n_{K,T} \in B(a^*,r))\Big\}, \\
&\geq (1-\gamma)^{N-1}, \qquad (5.18)
\end{aligned}
$$

using (5.17).

Step 4. The probability that the algorithm has converged at the end of the first epoch is

$$
\begin{aligned}
Pr\{a^1_{0,2} \in B(a^*,r)\} &= Pr\{a^1_{0,2} \in B(a^*,r) \text{ given } a^n_{K,1} \in B(a^*,r) \text{ for some } n\} \\
&\quad \times Pr\{a^n_{K,1} \in B(a^*,r) \text{ for some } n\}. \qquad (5.19)
\end{aligned}
$$

All N members in the congregation are randomly restarted at the beginnning of the first epoch, so the probabilty that at least one converges to the ball around a^* is equal to 1 minus the probability that none do. By definition of $p(K,T,r)$ in (5.1), this gives

$$
Pr\{a^n_{K,1} \in B(a^*,r) \text{ for some } n\} = 1 - (1 - p(K,1,r))^N.
$$

Combining with the lower bounds in (5.13) and (5.18), (5.19) becomes

$$
Pr\{a^1_{0,2} \in B(a^*,r)\} \geq (1-\gamma)^{N-1} \left(1 - \left(\frac{1-\sigma}{1-\sigma\gamma}\right)^N\right). \qquad (5.20)
$$

Step 5. Assume $T > 1$. The probability that the algorithm has converged at the end of the T-th epoch is

$$
\begin{aligned}
Pr\{a^1_{0,T+1} \in B(a^*,r)\} &= Pr\{a^1_{0,T+1} \in B(a^*,r) \text{ given } a^n_{K,T} \in B(a^*,r) \text{ for some } n\} \\
&\quad \times Pr\{a^n_{K,T} \in B(a^*,r) \text{ for some } n\}. \qquad (5.21)
\end{aligned}
$$

The probability that at least one of the $N-1$ restarted members converges to the ball around a^* is equal to

$$
Pr\{a^n_{K,T} \in B(a^*,r) \text{ for some } n \neq 1\} = 1 - (1 - p(K,T,r))^{N-1}. \qquad (5.22)
$$

The first member of the congregation is not restarted, but (5.15) shows that

$$Pr\{a^1_{K,T} \in B(a^*, r)\} \geq Pr\{a^1_{0,T} \in B(a^*, r)\}. \tag{5.23}$$

For independent events $E = \{a^n_{K,T} \in B(a^*, r) \text{ for some } n \neq 1\}$ and $F = \{a^1_{K,T} \in B(a^*, r)\}$, $Pr\{E \text{ or } F\} = Pr\{E\} + (1 - Pr\{E\})Pr\{F\}$. Therefore (5.22) and (5.23) imply

$$Pr\{a^n_{K,T} \in B(a^*, r) \text{ for some } n\} \geq 1 - (1 - p_T)^{N-1} + (1 - p_T)^{N-1} Pr\{a^1_{0,T} \in B(a^*, r)\}, \tag{5.24}$$

where $p_T = p(K, T, r)$. Combining with (5.18) and (5.24), equation 5.21 becomes

$$Pr\{a^1_{0,T+1} \in B(a^*, r)\} \geq (1 - \gamma)^{N-1}\Big[1 - (1 - p_T)^{N-1} + (1 - p_T)^{N-1} Pr\{a^1_{0,T} \in B(a^*, r)\}\Big].$$

Using the bounds in (5.13),

$$\begin{aligned} Pr\{a^1_{0,T+1} \in B(a^*, r)\} \geq & (1 - \gamma)^{N-1} \\ & \times \left[1 - \left(\frac{1 - \sigma}{1 - \sigma\gamma}\right)^{N-1} + \left(\frac{(1 - \gamma)(1 - \sigma)}{1 - \sigma\gamma}\right)^{N-1} Pr\{a^1_{0,T} \in B(a^*, r)\}\right]. \end{aligned} \tag{5.25}$$

Step 6. The recursive relationship (5.25) applied T times gives

$$\begin{aligned} Pr\{a^1_{0,T+1} \in B(a^*, r)\} \geq & (1 - \gamma)^{N-1} \left[1 - \left(\frac{1 - \sigma}{1 - \sigma\gamma}\right)^{N-1}\right] \sum_{t=0}^{T-2} \left(\frac{(1 - \gamma)^2(1 - \sigma)}{1 - \sigma\gamma}\right)^{(N-1)t} \\ & + \left(\frac{(1 - \gamma)^2(1 - \sigma)}{1 - \sigma\gamma}\right)^{(N-1)(T-1)} Pr\{a^1_{0,2} \in B(a^*, r)\} \\ = & (1 - \gamma)^{N-1} \left[1 - \left(\frac{1 - \sigma}{1 - \sigma\gamma}\right)^{N-1}\right] \frac{1 - \left(\frac{(1-\gamma)^2(1-\sigma)}{1-\sigma\gamma}\right)^{(N-1)(T-1)}}{1 - \left(\frac{(1-\gamma)^2(1-\sigma)}{1-\sigma\gamma}\right)^{N-1}} \\ & + (1 - \gamma)^{N-1} \left[1 - \left(\frac{1 - \sigma}{1 - \sigma\gamma}\right)^{N}\right] \left(\frac{(1 - \gamma)^2(1 - \sigma)}{1 - \sigma\gamma}\right)^{(N-1)(T-1)}. \end{aligned}$$

using the geometric sum and (5.20). Rearranging the first term gives the result. ∎

The assumption that (4.1) is Lagrange stable (Assumption *C5*) is used in the proof of Lemma 5.3 in order to show that the estimate cost $\Phi^n_{K,T}$ is a good estimate of $J(a^n_{K,T})$. For all estimates starting in the basin of attraction of a local minimum, once the estimate parameters have converged to the local minimum, the difference between the instantaneous cost at a_k and a_K is $o_\mu(1)$ for all x. If the initial estimate lands in the basin of attraction of an attractor at infinity this does not apply.

A gradient system such as (4.1) is not Lagrange stable if the cost function is decreasing as the size of the parameter increases. Here the global minimum of J is assumed to occur at some finite point a^*, so that the value of the cost function cannot keep decreasing at a rapid rate as $\|a\| \to \infty$. Rather, the gradient of J must decrease, so that J "flattens out". In such cases, the estimate parameters move very slowly if $\|a\|$ is large. This fact can be used to show that the estimate cost will (eventually) be a good estimate of the average cost even if $a^n_{0,T}$ lands in the basin of attraction of an attractor at infinity. Moreover, for any finite epoch length K, $\|a^n_{K,T}\|$ is bounded, so even if $J(a) \to J(a^*)$ as $\|a\| \to \infty$ there is a positive minimum value of $J(a^n_{K,T}) - J(a^*)$ for estimates not originating in A^*. In this way the assumption that there is no attractor at infinity could be avoided in Theorem 5.1.

6 Expected Time to Convergence

In Section 5 a lower bound on the probability of convergence after T epochs was derived. Under the assumptions of this paper, it is not possible to know exactly the probability of convergence after T epochs unless further assumptions are made. This is because whenever the algorithm is implemented, μ is non-zero and K is finite, so there is always some non-zero probability that estimates do not converge to a local minimum of J by the end of each epoch. That is, the quantity η used in Lemma 5.2 must be non-zero. However this probability of non-convergence decreases as μ decreases and K increases. That is, $p \to \sigma$ and $q \to 1 - \sigma$. Moreover, as $\alpha \to 0$ the online gradient estimate $\Phi^n_{k,T}$ approaches the true cost $J(a^n_{k,T})$. Therefore if the algorithm has converged, the best estimate will never be restarted.

Choose some fixed r such that $B(a^*, r) \subset A^0(a^*)$. Let $\hat{T}^N$ be the first epoch for which $a^1_{0,T+1} \in B(a^*, r)$, i.e. $\hat{T}^N$ is the number of epochs until the algorithm first converges. The size of the congregation is used as a superscript because, as the next lemma shows, the expected time until convergence varies with N. Using the above limiting argument, we will prove the following lemma about the expected time to convergence.

Lemma 6.1 *Consider the CGD algorithm with Assumptions C1 to C6. Set $K = K(\alpha) = L/\mu(\alpha)$ for some fixed nonzero L. As $\alpha \to 0$ the expected number of epochs until convergence satisfies:*

$$E(\hat{T}^N) \to \frac{1 - \sigma(1-\sigma)^{N-1}}{1 - (1-\sigma)^{N-1}}. \tag{6.1}$$

Inspection of the proof of Lemma 5.2 reveals that there exists an $o_\mu(1)$ function $l(\mu)$, such that it is possible to let $r = l(\mu)$ in the definition of $\hat{T}^n$. Thus the estimate parameters will be very close to the global minimum if μ is very small and K is sufficiently large. The

result of Lemma 6.1 still holds, but the fact that r shrinks with μ makes it necessary to allow K to grow faster than $\frac{1}{\mu}$ in the assumptions of the lemma.

Proof By Assumption 6, $\mu = o_\alpha(1)$, so $\mu \to 0$ as $\alpha \to 0$. The linking of K to α via $K = L/\mu(\alpha)$ ensures the averaging results in the appendix still hold. Therefore $\alpha \to 0$, $K \to \infty$, and $p \to \sigma$, so the probability that $\hat{T}^N = 1$ satisfies

$$\begin{aligned} Pr\{\hat{T}^N = 1\} &= 1 - Pr\left\{a^n_{K,1} \notin B(a^*, r) \text{ for all } n \in \{1, \ldots, N\}\right\} \\ &\to 1 - (1-\sigma)^N. \end{aligned}$$

The probability that $\hat{T}^N = T > 1$ satisfies

$$\begin{aligned} Pr\{\hat{T}^N = T\} = {} & Pr\left\{a^n_{K,T} \in B(a^*, r) \text{ for some } n \in \{2, \ldots, N\}\right\} \\ & \times Pr\left\{a^n_{K,t} \notin B(a^*, r) \text{ for all } n \in \{2, \ldots, N\} \text{ and } t \in \{2, \ldots, T-1\}\right\} \\ & \times Pr\left\{a^n_{K,1} \notin B(a^*, r) \text{ for all } n \in \{1, \ldots, N\}\right\} \\ \to {} & \left(1 - (1-\sigma)^{N-1}\right)(1-\sigma)^{(N-1)(T-2)}(1-\sigma)^N \qquad (T > 1). \end{aligned}$$

Therefore the expected time until convergence satisfies

$$\begin{aligned} E(\hat{T}^N) &= \sum_{T=1}^{\infty} T\, Pr\{\hat{T}^N = T\} \\ &\to (1 - (1-\sigma)^N) + (1-\sigma)\left(1 - (1-\sigma)^{N-1}\right) \sum_{T=2}^{\infty} T(1-\sigma)^{(N-1)(T-1)}. \end{aligned}$$

Using the geometric series, it can be seen that

$$\begin{aligned} E(\hat{T}^N) &\to (1 - (1-\sigma)^N) + (1-\sigma)\left(1 - (1-\sigma)^{N-1}\right)\left(\frac{1}{(1-(1-\sigma)^{N-1})^2} - 1\right) \\ &= \frac{1 - (1-\sigma)^N - (1-\sigma)^{N-1} + (1-\sigma)^{2N-1} + (1-\sigma)^N(2 - (1-\sigma)^{N-1})}{1 - (1-\sigma)^{N-1}} \\ &= \frac{1 + (1-\sigma)^N - (1-\sigma)^{N-1}}{1 - (1-\sigma)^{N-1}}, \end{aligned}$$

which gives (6.1). ∎

From (6.1), it can be seen that $E(\hat{T}^N) \to 1$ as $N \to \infty$, which may lead one to the conclusion that it is best to have a very large congregation. However, in order to make a fair comparison between different population sizes, the expected computation for each must be compared. The expected computation increases like $NE(\hat{T}^N)$ as N increases. Thus the expected computation is unbounded in the limit $N \to \infty$. The expected computation also increases as $\alpha \to 0$ (and thus $\mu \to 0$) since $K \to \infty$.

For small values of σ, Laurent series expansion of (6.1) reveals that

$$NE(\hat{T}^N) \approx \frac{N}{(N-1)\sigma} + \frac{N(N-4)}{2(N-1)}. \tag{6.2}$$

For $N > 2$ small, the first term is dominant, and is decreasing as N increases from 2. The right hand side of (6.2) is a minimum when $N \approx 1 + \sqrt{\frac{2}{\sigma}}$. Therefore for small values of σ the optimal value of N is approximately $1 + \sqrt{\frac{2}{\sigma}}$. The following lemma shows that, for any $\sigma \in (0,1)$, in the limit as $\mu \to 0$ and $K \to \infty$, the expected computation for a congregation with N members is never less than half of the expected computation for a congregation with 2 members. Thus the reduction in computation gained by using the optimal value of N is not more than a factor of 2.

Lemma 6.2 *Consider the CGD algorithm with Assumptions C1 to C6. Set $K = K(\alpha) = L/\mu(\alpha)$ for some fixed nonzero L. As $\alpha \to 0$, the expected number of epochs until convergence satisfies:*

$$\frac{NE(\hat{T}^N)}{2E(\hat{T}^2)} \geq \frac{1}{2}. \tag{6.3}$$

Proof From Lemma 6.1, it is known that

$$NE(\hat{T}^N) = N\frac{1-\sigma(1-\sigma)^{N-1}}{1-(1-\sigma)^{N-1}}.$$

Substituting $N = 2$ gives

$$2E(\hat{T}^2) = 2\frac{1-\sigma(1-\sigma)}{\sigma}. \tag{6.4}$$

Since $N \geq 2$ and $\sigma \in (0,1)$, $1-(N-1)\sigma \leq (1-\sigma)^{N-1} \leq (1-\sigma)$. Therefore

$$NE(\hat{T}^N) \geq N\frac{1-\sigma(1-\sigma)}{(N-1)\sigma}. \tag{6.5}$$

Combining (6.4) and (6.5) gives

$$\frac{NE(\hat{T}^N)}{2E(\hat{T}^2)} \geq \frac{N}{2(N-1)} \geq \frac{1}{2}. \tag{6.6}$$

∎

The limit of the variance of $\hat{T}^N$ as $\alpha \to 0$ and $K \to \infty$ can also be determined. We have $\mathrm{var}(\hat{T}^N) = E((\hat{T}^N)^2) - E(\hat{T}^N)^2$, where

$$\begin{aligned} E((\hat{T}^N)^2) &= \sum_{T=1}^{\infty} T^2 Pr\{\hat{T}^N = T\} \\ &\to 1 + 3(1-\sigma)^N + \frac{5(1-\sigma)^{2N-1} - 3(1-\sigma)^{3N-2}}{(1-(1-\sigma)^{N-1})^2} \end{aligned}$$

which can be obtained from a straightforward but tedious evaluation of the sum. Thus substituting (6.1) for $E(\hat{T}^N)$, gives (after some further manipulation)

$$\mathrm{var}(\hat{T}^N) \to (1-\sigma)^N \left(3 + \frac{6(1-\sigma)^{N-1} - (1-\sigma)^N - 1}{(1-(1-\sigma)^{N-1})^2}\right) =: v(\sigma, N) \tag{6.7}$$

as $\alpha \to 0$ and $K \to \infty$. By inspection, $v(\sigma, N) = \Omega_\sigma(\sigma^{-2})$ but $v(\sigma, N)$ is monotonically decreasing in N, and goes to zero exponentially fast in N for fixed σ. This suggests a slight advantage in a larger value of N not apparent from solely considering the expected number of epochs (or expected amount of computation) required for convergence.

7 CMA Simulation Results

In this section one application of the congregational gradient descent algorithm is discussed, and results of simulation studies are presented. In particular, the expected time relationships derived in Lemmas 6.1 and 6.2 are illustrated.

In band-limited data communication systems, the transmitted signals can be extended (smeared out) by the distortion of an analog channel over a much longer interval than their original duration. Adaptive equalizers are used to remove the resulting intersymbol interference, and thus reconstruct the original signal [8, 16].

Blind equalizers are a special kind of adaptive equalizers which do not require a known training sequence. Instead, they aim to restore known generic properties of the original signal. The constant modulus algorithm (CMA) is a popular algorithm for adaptive blind channel equalisation. The original signals are assumed to have constant modulus, and the algorithm minimises a cost function defined by both the modulus of the original signal and of the reconstructed signal. It is known that the underlying cost function possesses non-global local minima for even very simple channel models [16]. Some schemes for fixing the ill-convergence caused by local minima have been devised [18, 17]. These schemes use more information than is assumed for the congregational gradient algorithm.

A sequence of i.i.d. binary valued signals ($u_k \in \{-1, 1\}$) is sent by a transmitter through a channel exhibiting linear distortion. In the following, it is assumed that the channel has an AR(n) structure. Therefore the transmitted signal satisfies

$$u_k = \sum_{i=0}^{n} a^*(i+1) y_{k-i}$$

for some parameter vector $a^* \in \mathbb{R}^{n+1}$, where y_k is the received signal. This can be written

$$y_k = \frac{1}{a^*(1)} \left(u_k - \sum_{i=1}^{n} a^*(i+1) y_{k-i} \right).$$

Let $x_k = (y_k, y_{k-1}, \ldots, y_{k-n})^\top$. The objective of the equaliser is to recover the original sequence (u_k) from the received sequence (x_k). In the following an MA(n) equaliser is used,[1] which gives the reconstructed signal

$$z_k = \sum_{i=0}^{n} a(i+1) y_{k-i} = a^\top x_k$$

for some parameter vector $a \in \mathbb{R}^n$. The ordinary CMA algorithm is simply stepwise gradient descent with the instantaneous cost function

$$\phi(a, x) = \frac{1}{4}(z^2 - 1)^2 = \frac{1}{4}((a^\top x)^2 - 1)^2. \tag{7.1}$$

Therefore application of the congregational gradient descent algorithm to this problem requires only minimal alteration of the ordinary CMA algorithm.

Clearly $\phi \geq 0$ and $\phi(\pm a^*, \cdot) = 0$. Moreover, if the received sequence x_k is sufficiently exciting, $\phi \equiv 0$ if and only if $a = \pm a^*$. Hence the average cost function J has exactly two global minima: a^* and $-a^*$. The congregational gradient descent algorithm can be used for this cost function. In order to comply with Assumption 5 of Theorem 5.1, the sign of the first component of the estimated parameter can be fixed, so that only one of a^* and $-a^*$ is in A. For any $a \in \mathbb{R}^n$, $\phi(ca, x) \to \infty$ as $c \to \infty$ for almost all x, so there is no attractor at infinity.

Figure 1 shows the results of a series of experiments using the above setup. In this case $n = 7$ was used, the channel parameters were

$$a^* = (1, -0.25, -0.5, 0.2, 0.1, -0.2, -0.1)^\top \tag{7.2}$$

and the initial parameter estimates were chosen in $A = [0, 2] \times [-2, 2]^6 \subset \mathbb{R}^7$. The signal u_k took on the values ± 1 with approximately equal probability. The stepsize and approximation parameters were $\mu = \alpha = 0.005$. The epochs were $K = 1999$ iterations long, and the algorithm was said to have converged when $\|a^1_{0,T} - a^*\|^2 \leq 0.02$ (i.e. $r^2 = 0.02$). For each $N \in \{2, \ldots, 10\}$, the algorithm was run and $\hat{T}^N$, the number of epochs until convergence, was recorded. This was repeated 1000 times using the same binary signal u_k (and hence the same sequence (x_k)), but different initial estimates. For each $N \in \{2, \ldots, 10\}$, the average, over all 1000 trials, of $N\hat{T}^N$, is marked with a circle in Figure 1. The solid curve plotted in Figure 1 is the expected value of $N\hat{T}^N$, calculated by multiplying N times the limiting value of $E(\hat{T}^N)$ that appears in Lemma 6.1, with $\sigma = 0.167$. The dashed lines are calculated by adding $\pm 3N\sqrt{\frac{v(0.167,N)}{1000}}$, where $v(\cdot, \cdot)$ is

[1] Although we are assuming here that the channel is in fact exactly invertible by an MA(n) equaliser, such an assumption is not necessary for our algorithm. Nor is the assumption necessarily valid in practice.

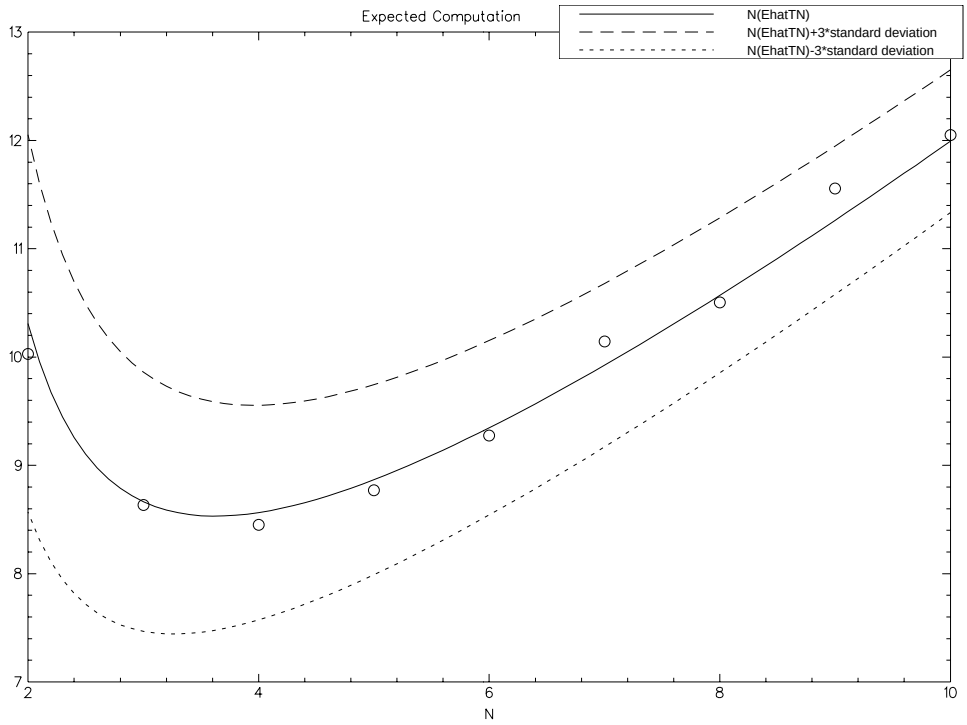

Figure 1: Expected computation of the Congregational Algorithm as a function of N, the population size, when applied to a blind equalisation problem.

the limiting value of the variance given in (6.7). The initial estimates were uniformly distributed in A, so it can be surmised that the volume of $A^0(a^*)$ is approximately 0.167 times the volume of A^0.

The congregational algorithm has also been successfully applied to a nonlinear regression problem. The input-output relationship of the system to be identified was

$$y(x) = \sum_{i=1}^{9} \frac{h_i}{1 + 100 * \|x - c_i\|}$$

for particular values of h_i, c_i, and the model class was defined by the input-output relationship

$$f(a, x) = \frac{1}{1 + 100 * \|x - a\|}.$$

The instantaneous cost function $\phi(a, x) = (f(a, x) - y(x))^2$ was used, so that congregational gradient descent performed minimisation of the output error. This problem was shown numerically to have multiple local minima.

8 Conclusions

We have proposed and analysed a version of stepwise gradient descent which is based upon the idea of evolving a population of solutions. It is suitable for a wide range of learning and

optimisation problems. We have determined the expected computation required for the algorithm to locate the global minimum of the expected cost function and have applied the algorithm to some examples and shown that our predictions about its behaviour are well corroborated in our experiments. The algorithm is well suited to problems where there is a naturally occuring differentiable parametrisation; of course not all learning problems fit into this category.

The most obvious further work to be done on the algorithm is to perform a *stochastic* averaging analysis [9, 47, 12], where the results would depend in some explicit way on the distribution of the (x_k) sequence. (All our results are in terms of a given (fixed) (x_k) sequence, as are similar analyses such as [10].) The extension to stochastic analysis is not straightforward, since standard techniques would introduce a non-zero probability of escape from local minima during each epoch. However, it seems that such an analysis would allow one to determine the rate with which η (and thus γ) shrinks as $\alpha \to 0$ and $K \to \infty$. If that were done, then one could make rather strong (PAC-like) assertions about the performance of the algorithm.

One other question concerns the adaptation of a solution to a changing environment. In the genetic algorithm literature there has been much discussion about the importance of diversity of the population in a GA in order for the GA to be able to respond well to changes in the environment [14, 45, 38]. In parametric optimisation it is well known [26] that one can not expect to be able to follow a smooth trajectory and stay at the optimal solution even if the environment changes are themselves smooth. Thus one can *not* rely on the algorithm presented here to always sit at the global optimum, after the initial convergence phase. There is an obvious question concerning our algorithm and whether one needs to modify the restart schedule in order to optimise the algorithm's performance under a changing cost function.

9 Acknowledgements

This work was supported by the Australian Research Council. Thanks to Stephanie Forrest for helpful and enjoyable discussions and pointers to the GA literature.

References

[1] W. Atmar. Notes on the simulation of evolution. *IEEE Transactions on Neural Networks*, 5(1):130–147, January 1994.

[2] N. Baba. Global optimization of functions by the random optimization method. *International Journal of Control*, 30:1061–1065, 1977.

[3] T. Bäck, F. Hoffmeister, and H.-P. Schwefel. A survey of evolutionary strategies. In R.K. Belew and L.B. Booker, editors, *Proceedings of the 4th International Conference on Genetic*

Algorithms, pages 2–9. Morgan Kaufmann, La Jolla, 1991. //ftp:lumpi.informatik.uni-dortmund.de/pub/EA/icga91.ps.gz.

[4] T. Bäck and H.-P. Schwefel. An overview of evolutionary algorithms for parametric optimization. *Evolutionary Computation*, 1(1):1–23, 1993. //ftp:lumpi.informatik.uni-dortmund.de/pub/EA/ec1:1.ps.Z.

[5] P.R. Barros. *Robust Performance in Adaptive Control.* PhD thesis, The University of Newcastle, March 1990.

[6] D.L. Battle and M.D. Vose. Isomorphisms of genetic algorithms. In G.J.E. Rawlins, editor, *Foundations of Genetic Algorithms*, pages 243–251. Morgan Kaufmann, San Mateo, 1991.

[7] E.B. Baum, D. Boueh, and C. Garrett. On genetic algorithms. In *Proceedings of the Eigth ACM Annual Workshop on Computational Learning Theory*, July 1995. To appear.

[8] A. Benveniste and M. Goursat. Blind equalizers. *IEEE Transactions on Communications*, 32:871–883, August 1984.

[9] A. Benveniste, M. Métivier, and P. Prioret. *Adaptive Algorithms and Stochastic Approximations.* Springer-Verlag, Berlin Heidelberg, 1990.

[10] K.L. Blackmore, R.C. Williamson, and I.M.Y. Mareels. Learning nonlinearly parametrized decision regions. *Journal of Mathematical Systems, Estimation, and Control*, 1995. To appear.

[11] S.H. Brooks. A discussion of random methods for seeking minima. *Operations Research*, 6(2):244–251, 1958.

[12] J.A. Bucklew, T.G. Kurtz, and W.A. Sethares. Weak convergence and local stability properties of fixed step size recursive algorithms. *IEEE Transactions on Information Theory*, 39:966–978, 1993.

[13] N. Cesa-Bianchi, P.M. Long, and M.K. Warmuth. Worst-case quadratic loss bounds for a generalization of the Widrow-Hoff rule. In L. Pitt, editor, *Proceedings of the Sixth ACM Annual Workshop on Computational Learning Theory*, pages 429–438, July 1993.

[14] K.A. DeJong. Genetic algorithms are not function optimizers. In L.D. Whitely, editor, *Foundations of Genetic Algorithms 2*, pages 5–17. Morgan Kaufmann, San Mateo, 1993.

[15] C.A. Desoer and M. Vidyasagar. *Feedback Systems: Input-Output Properties.* Academic Press, New York, 1975.

[16] Z. Ding, R.A. Kennedy, B.D.O. Anderson, and C.R. Johnson Jr. Ill-convergence of Godard blind equalizers in data communication systems. *IEEE Transactions on Communications*, 39(9):1313–1327, 1991.

[17] K. Dogancay and R.A. Kennedy. Testing for the convergence of a linear decision directed equalizer. *IEE Proc.–Vis. Image Signal Process.*, 141(2):129–136, April 1994.

[18] K. Dogancay and R.A. Kennedy. Testing output performance in blind adaptation. In *Proceedings of the 33rd IEEE Conference on Decision and Control*, pages 2817–2818, December 1994.

[19] W. Finnoff. Diffusion approximations for the constant learning rate backpropagation algorithm and resistance to local minima. In *Advances in Neural Information Processing 5*, pages 459–466. Morgan Kaufmann, 1993.

[20] D.B. Fogel. An introduction to simulated evolutionary optimization. *IEEE Transactions on Neural Networks*, 5(1):3–14, January 1994.

[21] S. Forrest. Genetic algorithms: Principles of natural selection applied to computation. *Science*, 261:972–978, 13 August 1993.

[22] S. Forrest and M. Mitchell. What makes a problem hard for a genetic algorithm? Some anomalous results and their explanation. *Machine Learning*, 13:285–319, 1993.

[23] M.R. Frater, R.R. Bitmead, and C.R. Johnson Jr. Escape from stable equilibria in blind adaptive equalization. In *Proceedings of the 31st IEEE Conference on Decision and Control*, pages 1756–1761, December 1992.

[24] M. Gell-Mann. *The Quark and the Jaguar*. Little, Brown and Company, London, 1994.

[25] D.E. Goldberg, K. Deb, and J.H. Clark. Accounting for noise in the sizing of populations. In L.D. Whitely, editor, *Foundations of Genetic Algorithms 2*, pages 127–140, San Mateo, 1993. Morgan Kaufmann.

[26] J. Guddat, F. Guerra Vasquez, and H.Th. Jongen. *Parametric Optimization: Singularities, Pathfollowing and Jumps*. B.G. Teubner, Stuttgart, 1990. Published simultaneously by John Wiley, Chichester.

[27] T.M. Heskes and B. Kappen. On-line learning processes in artificial neural networks. In J.G. Taylor, editor, *Mathematical Approaches to Neural Networks*, pages 199–233. North-Holland, Amsterdam, 1993.

[28] M.W. Hirsch and S. Smale. *Differential Equations, Dynamical Systems, and Linear Algebra*. Academic Press, New York, 1974.

[29] J.H. Holland. *Adaptation in Natural and Artificial Systems*. MIT press, Cambridge, MA, 1992.

[30] J. Homer. *Adaptive Echo Cancellation in Telecommunications*. PhD thesis, Australian National University, April 1994.

[31] J. Horn and D.E. Goldberg. Genetic algorithm difficulty and the modality of fitness landscapes. to appear in the Proceedings of the Foundations of Genetic Algorithms (FOGA) 3 Workshop held July 30 — August 2, 1994, Estes Park, Colorado, IlliGAL Report 94006 (//ftp:gal4.ge.uiuc.edu/pub/papers/Publications/94006.ps.Z).

[32] R.A. Jarvis. Adaptive global search by the process of competitive evolution. *IEEE Transactions on Systems, Man and Cybernetics*, 5:297–311, 1975.

[33] T. Jones. A model of fitness landscapes. Santa Fe Institute Technical Report TR 94-02-002 (February 1994) (//ftp:ftp.santafe.edu/pub/terry/model-of-landscapes.ps.gz).

[34] K. De Jong. Learning with genetic algorithms: An overview. *Machine Learning*, 3:121–138, 1988.

[35] C.R. Johnson Jr, S. Dasgupta, and W.A. Sethares. Averaging analysis of local stability of a real constant modulus algorithm adaptive filter. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 36(6):900–910, 1988.

[36] C.-M. Kuan and K. Hornik. Convergence of learning algorithms with constant learning rates. *IEEE Transactions on Neural Networks*, 2(5):484–489, 1991.

[37] T.K. Leen and J. Moody. Weight space probability densities in stochastic learning: I. Dynamics and equilibria. In *Advances in Neural Information Processing 5*, pages 451–458. Morgan Kaufmann, 1993.

[38] S.W. Mahfoud. Population sizing for sharing methods. to appear in the Proceedings of the Foundations of Genetic Algorithms (FOGA) 3 Workshop held July 30 — August 2, 1994, Estes Park, Colorado, IlliGAL Report 94005 (//ftp:gal4.ge.uiuc.edu/pub/papers/Publications/94005.ps.Z).

[39] C.J. McMurty and K.S. Fu. A variable structure automaton used as a multimodal searching technique. *IEEE Transactions on Automatic Control*, 11:379–387, 1966.

[40] M. Mitchell, J.H. Holland, and S. Forrest. When will a genetic algorithm outperform hill climbing? In G. Tesuaro J.D. Cowan and J. Alspector, editors, *Advances in Neural Information Processing Systems 6*. Morgan Kaufmann, 1994.

[41] X. Qi and F. Palmieri. Theoretical analysis of evolutionary algorithms with an infinite population size in continuous space part 1: Basic properties of selection and mutation. *IEEE Transactions on Neural Networks*, 5(1):102–119, 1994.

[42] J.P. Ros. Learning boolean functions with genetic algorithms: A PAC analysis. In L.D. Whitely, editor, *Foundations of Genetic Algorithms 2*, pages 257–275, San Mateo, 1993. Morgan Kaufmann.

[43] G. Rudolph. Convergence analysis of canonical genetic algorithms. *IEEE Transactions on Neural Networks*, 5(1):96–101, 1994.

[44] J.A. Sanders and F. Verhulst. *Averaging Methods in Nonlinear Dynamical Systems*. Applied Mathematical Sciences; v. 59. Springer-Verlag, New York, 1985.

[45] R.E. Smith, S. Forrest, and A.S. Perelson. Searching for diverse, cooperative populations with genetic algorithms. *Evolutionary Computation*, 1(2):127–149, 1993.

[46] F.J. Solis and R. J.-B. Wets. Minimization by random search techniques. *Mathematics of Operations Research*, 6(1):19–30, 1981.

[47] V. Solo and X. Kong. *Adaptive Signal Processing Algorithms: Stability and Performance*. Prentice Hall, Englewood Cliffs, New Jersey, 1995.

[48] M.D. Vose. Modelling simple genetic algorithms. In L.D. Whitely, editor, *Foundations of Genetic Algorithms 2*, pages 63–73, San Mateo, 1993. Morgan Kaufmann.

[49] B. Widrow and S.D. Stearns. *Adaptive Signal Processing*. Prentice-Hall, Englewood Cliffs, NJ, 1985.

[50] T. Yoshizawa. *Stability Theory and the Existence of Periodic Solutions and Almost Periodic Solutions*. Applied Mathematical Sciences; v. 14. Springer-Verlag, New York, 1975.

[51] A.A. Zhigljavsky. *Theory of Global Random Search*. Kluwer Academic Publishers, Dordrecht, 1991.

A Averaging Theory

In this appendix Theorem 2.6 is proved. The derivation of Theorem 2.6 is very similar to the derivation of Theorem 4.2.1 in Sanders and Verhulst. In particular, the results in Theorem A.4 parallel Lemma 3.2.6 to Theorem 3.3.3 in [44]. Theorem 2.6 differs from Theorem 4.2.1 in [44] in two ways: in [44] the original equation is a differential equation rather than a difference equation; and in [44] the critical point must be a uniformly asymptotically stable critical point of the *linearisation* of the averaged ODE, rather than of the averaged ODE itself. Their condition is much stronger than the condition used here—it is equivalent to saying that the critical point is a uniformly *exponentially* stable critical point of the ODE. In order to impose only the weaker condition, an inverse Lyapunov function result (Theorem A.5) is used. However the weaker condition results in a weaker approximation result—the error has the form $o_\mu(1)$ instead of the $O_\mu(\delta^{\frac{1}{2}}(\mu))$ error in [44].

Before proving Theorem 2.6, some useful results will be stated. The following lemma is a special case of the Bellman-Gronwall Lemma [15], and Lemma A.2 is a special case of the Comparison Principle [50]. In Theorem A.4 finite time averaging results are derived. The final result in Theorem A.4 is used repeatedly in the proof of Theorem 2.6.

Lemma A.1 Bellman-Gronwall *Assume that for all* $k \in \mathbb{N}$,

$$a_k \leq c_1 \sum_{l=0}^{k-1} a_l + c_2,$$

where $a_l \geq 0$ *for all* $l = 0, \ldots, k$, $c_1 \in \mathbb{R}$ *and* $c_2 \geq 0$. *Then*

$$a_k \leq (c_1 a_0 + c_2)(1 + c_1)^{k-1}.$$

Lemma A.2 Comparison Principle *Assume that for all* $t > t_0$,

$$\dot{a}(t) \leq c a(t)$$

where $a(t)$ *is continuous and nonnegative for all* $t > t_0$. *Then*

$$a(t) \leq a(t_0) e^{a(t-t_0)}.$$

Definition A.3 *A property is said to hold for k* on the time scale $l(\mu)$ *if it is true for all k satisfying $0 \leq k \leq Kl(\mu)$, where K is a constant independent of μ and $l(\mu)$ is an order function.*

Theorem A.4 *With Assumptions A1 to A5, let*

$$\phi(k) = \sum_{l=k_0}^{k_0+k-1} H(a_l, x_l)$$

$$\phi_L(k) = \frac{1}{L} \sum_{m=k_0}^{k_0+L-1} \phi(m+k)$$

$$H_L(a,k) = \frac{1}{L} \sum_{k=k_0}^{k_0+L-1} H(a, x_{k+l})$$

for any $k_0 \in \mathbb{N}_0$ and $L \in \mathbb{N}$. For some $\hat{a} \in A$, let a_k, b_k and $a_{av}(t)$ be defined according to the following equations:

$$a_{k+1} = a_k - \mu H(a_k, x_k) - \mu\beta(\mu) h_k(a_k, x_k) \quad ; \quad a_{k_0} = \hat{a} \tag{A.1}$$

$$b_{k+1} = b_k - \mu H_L(b_k, k) \quad ; \quad b_{k_0} = \hat{a}. \tag{A.2}$$

$$\dot{a}_{av} = -\mu H^{av}(a_{av}(t)) \quad ; \quad a_{av}(k_0) = \hat{a} \tag{A.3}$$

for all $k \in \mathbb{N}_0$, $t \in \mathbb{R}$ such that $k, t \geq k_0$. Then

1. $\phi(k) = \phi_L(k) + O_\mu(L)$
2. $\phi_L(k) = \sum_{l=k_0}^{k_0+k-1} H_L(a_l, l) + O_\mu(L)$
3. $a_k = b_k + O_\mu(\mu L)$
4. $H_L(a,k) = H^{av}(a) + O_\mu\left(\frac{\delta(\mu)}{\mu L}\right)$
5. $b_k = a_{av}(k) + O_\mu(\mu) + O_\mu\left(\frac{\delta(\mu)}{\mu L}\right)$
6. $a_k = a_{av}(k) + o_\mu(1)$

for k on the time scale $\frac{1}{\mu}$.

Proof Since H is bounded on a compact domain, so are H_L and H^{av}. The initial condition $\hat{a}$ is finite, so a_k and b_k, and $a_{av}(k)$ are bounded for k on the time scale $\frac{1}{\mu}$. This is because if a_k and b_k are bounded at time k then the increment $|a_{k+1} - a_k|$ is bounded by $\mu M(\|a_k\|) + o_\mu(\mu)$. This fact can be applied iteratively to find a constant $\hat{r}$ such that a_k, $b_k \leq \hat{r}$ for k on the time scale $\frac{1}{\mu}$. Let $M := M(\hat{r})$ and $\lambda := \lambda(\hat{r})$. Then

$$\|H(a,x)\| \leq M$$

$$\|H(a,x) - H(b,x)\| \leq \lambda \|a - b\|$$

where both a and b are solutions to one of the equations (A.1) to (A.3) for k on the time scale $\frac{1}{\mu}$. These constants are used in the rest of the proof.

Result 1:

$$\begin{aligned}
\|\phi(k) - \phi_L(k)\| &\leq \frac{1}{L} \sum_{m=k_0}^{k_0+L-1} \|\phi(k) - \phi(m+k)\| \\
&\leq \frac{1}{L} \sum_{m=k_0}^{k_0+L-1} \sum_{l=k_0+k}^{k_0+k+m-1} \|H(a_l, x_l)\| \\
&\leq \frac{1}{L} \sum_{m=k_0}^{k_0+L-1} mM \\
&= \frac{M(L-1)(L-2)}{2L} \\
&= O_\mu(L).
\end{aligned}$$

Result 2: From the definitions of ϕ and ϕ_L,

$$\begin{aligned}
\phi_L(k) &= \frac{1}{L} \sum_{m=k_0}^{k_0+L-1} \sum_{l=k_0}^{k_0+k+m-1} H(a_l, x_l) \\
&= \frac{1}{L} \sum_{m=k_0}^{k_0+L-1} \sum_{l=k_0+m}^{k_0+k+m-1} H(a_l, x_l) + R_1 \\
&= \frac{1}{L} \sum_{m=k_0}^{k_0+L-1} \sum_{l=k_0}^{k_0+k-1} H(a_{l+m}, x_{l+m}) + R_1 \\
&= \frac{1}{L} \sum_{m=k_0}^{k_0+L-1} \sum_{l=k_0}^{k_0+k-1} H(a_l, x_{l+m}) + R_1 + R_2 \\
&= \sum_{l=k_0}^{k_0+k-1} H_L(a_l, l) + R_1 + R_2.
\end{aligned}$$

The terms R_1 and R_2 have been defined implicitly. They satisfy

$$\begin{aligned}
\|R_1\| &= \left\| \frac{1}{L} \sum_{m=k_0}^{k_0+L-1} \sum_{l=k_0}^{k_0+m-1} H(a_l, x_l) \right\| \\
&\leq \frac{1}{L} \sum_{m=k_0}^{k_0+L-1} mM \\
&\leq \frac{M(L-1)(L-2)}{2L} \\
&= O_\mu(L)
\end{aligned}$$

and

$$\begin{aligned}
\|R_2\| &= \frac{1}{L}\left\|\sum_{m=k_0}^{k_0+L-1}\sum_{l=k_0}^{k_0+k-1}(H(a_{l+m},x_{l+m})-H(a_l,x_{l+m}))\right\| \\
&\leq \frac{1}{L}\sum_{m=k_0}^{k_0+L-1}\sum_{l=k_0}^{k_0+k-1}\lambda\|a_{l+m}-a_l\| \\
&\leq \frac{\lambda}{L}\sum_{m=k_0}^{k_0+L-1}\sum_{l=k_0}^{k_0+k-1}\mu Mm \\
&\leq \frac{\lambda\mu M}{L}\sum_{l=k_0}^{k_0+k-1}\frac{(L-1)(L-2)}{2} \\
&\leq \frac{\lambda\mu MkL}{2} \\
&= O_\mu(L)
\end{aligned}$$

for k on the time scale $\frac{1}{\mu}$. The result follows.

Result 3: From the definition of a_k,

$$\begin{aligned}
a_k &= \hat{a}-\mu\sum_{l=k_0}^{k_0+k-1}(H(a_l,x_l)+\mu\beta(\mu)h_l(a_l,x_l)) \\
&= \hat{a}-\mu\phi(k)+O_\mu(\mu\beta(\mu)k)
\end{aligned}$$

using Assumption *A5* and the fact that a_k is bounded for k on the time scale $\frac{1}{\mu}$. Hence

$$\begin{aligned}
a_k &= \hat{a}-\mu\phi_L(k)+O_\mu(\mu L)+O_\mu(\beta(\mu)) \\
&= \hat{a}-\mu\sum_{l=k_0}^{k_0+k-1}H_L(a_l,l)+O_\mu(\mu L+\beta(\mu)),
\end{aligned}$$

using result 1 and then result 2. Combining this with (A.2) gives

$$\begin{aligned}
\|a_k-b_k\| &= \mu\left\|\sum_{l=k_0}^{k_0+k-1}(H_L(a_l,l)-H_L(b_l,l))\right\|+O_\mu(\mu L+\beta(\mu)), \\
&\leq \mu\lambda\sum_{l=k_0}^{k_0+k-1}\|a_l-b_l\|+O_\mu(\mu L+\beta(\mu)).
\end{aligned}$$

Lemma A.1 applies, so

$$\|a_k-b_k\| \leq O_\mu(\mu L+\beta(\mu))(1+\mu\lambda)^{k-1}.$$

The result follows since $(1+\mu\lambda)^k \leq e^{\mu\lambda k} = O_\mu(1)$ for k on the time scale $\frac{1}{\mu}$, and $\beta(\mu) = o_\mu(1)$.

Result 4:

$$H_L(a,k) - H^{av}(a) = \frac{1}{L}\sum_{l=k_0}^{k_0+L-1}(H(a,x_{k+l}) - H^{av}(a))$$
$$= \frac{1}{L}\sum_{l=k_0+k}^{k_0+L+k-1}(H(a,x_l) - H^{av}(a))$$

which, by the definition of $\delta(\mu)$, gives

$$\|H_L(a,k) - H^{av}(a)\| \leq \frac{\delta(\mu)}{\mu L}.$$

Result 5:

$$b_k - a_{av}(k) = -\mu\sum_{l=k_0}^{k_0+k-1}\int_l^{l+1}(H_L(b_l,l) - H^{av}(a_{av}(t)))\,dt$$
$$= -\mu\sum_{l=k_0}^{k_0+k-1}\int_l^{l+1}(H^{av}(b_l) - H^{av}(a_{av}(t)))\,dt + O_\mu\left(\frac{\delta(\mu)k}{L}\right)$$

according to result 4. Using Lipschitz continuity of $H(a,x)$ in a, and hence of H^{av}, gives

$$\|b_k - a_{av}(k)\| \leq \mu\lambda\sum_{l=k_0}^{k_0+k-1}\int_l^{l-1}\|b_l - a_{av}(t)\|dt + O_\mu\left(\frac{\delta(\mu)k}{L}\right)$$
$$\leq \mu\lambda\sum_{l=k_0}^{k_0+k-1}\|b_l - a_{av}(l)\| + \mu\lambda k\mu M + O_\mu\left(\frac{\delta(\mu)k}{L}\right)$$

using the definition of c

$$\leq \left(\mu^2\lambda kM + O_\mu\left(\frac{\delta(\mu)k}{L}\right)\right)(1+\mu\lambda)^{k-1}$$

using Lemma A.1

$$\leq O_\mu\left(\mu + \frac{\delta(\mu)}{\mu L}\right)$$

for k on the time scale $\frac{1}{\mu}$.

Result 6: Using results 3 and 5,

$$\|a_k - a_{av}(k)\| = O_\mu(\mu L) + O_\mu(\mu) + O_\mu\left(\frac{\delta(\mu)}{\mu L}\right) \tag{A.4}$$

on the time scale $\frac{1}{\mu}$. At this stage, the length of the short term averaging window $L \in \mathbb{N}$ is arbitrary. Thus L can be chosen in order to make all of the terms in the right hand side of (A.4) go to zero as μ goes to zero. The choice of L depends on the rate of convergence of $\delta(\mu)$ to 0.

Case 1: $\delta(\mu) = \Omega_\mu(\mu^2)$.

Choose $L = \left\lceil c\frac{\delta(\mu)^{\frac{1}{2}}}{\mu} \right\rceil$ for some c independent of μ. Then $L \geq 1$ for all sufficiently small μ, since $\delta(\mu)^{\frac{1}{2}} = \Omega_\mu(\mu)$. Thus the first and last terms in A.4 become $O_\mu\left(\delta(\mu)^{\frac{1}{2}}\right) = o_\mu(1)$.

Case 2: $\delta(\mu) = O_\mu(\mu^2)$.

Choose any value for L, independent of μ. Again all terms are $o_\mu(1)$, and the result follows. ∎

The following is a variant of Theorem 11.4 in [50]. It is simpler than the result in [50] because the ODE is assumed to be autonomous.

Theorem A.5 *Let $f : \mathbb{R}^m \to \mathbb{R}^m$ be Lipschitz continuous on some compact set $A \subset \mathbb{R}^m$. If $a^* \in \overset{\circ}{A}$ is a uniformly asymptotically stable critical point of the ODE $\dot{a} = f(a)$, with basin of attraction $A^0 \subset A$, then there exists a Lyapunov function $V(a) : A^0 \to \mathbb{R}$ and an open neighbourhood $N \subset A^0$ of a^* such that, for all $a, b \in N$,*

1. *$\alpha(\|a - a^*\|) \leq V(a) \leq \beta(\|a - a^*\|)$, where $\alpha(\cdot)$, $\beta(\cdot)$ are continuous, increasing, positive definite, $\alpha(r) \to \infty$ as $r \to \infty$, and $\beta(0) = 0$;*
2. *$|V(a) - V(b)| \leq \lambda_V \|a - b\|$, for some $\lambda_V > 0$;*
3. *$\dot{V}(a) \leq -cV(a)$, for some $c > 0$, where $a(t)$ is a solution of $\dot{a} = f(a)$.*

Outline of the proof of Theorem 2.6 Theorem A.5 applies to equation 2.3 (equivalently equation A.3). The neighbourhood N where the Lyapunov function satisfies properties 1, 2, and 3 contains an open ball centred at a^* with some radius $\delta > 0$.

Since a^* is asymptotically stable in equation 2.3, all solutions of (2.3) that originate in B^0 enter $B(a^*, \delta)$ in some finite time K. The finite time averaging result in Theorem A.4 can be applied for $k \in \{0, \ldots, K\}$, so that all solutions of (2.2) that originate in B^0 enter N in time K.

Once the solution of (2.2) enters N, the contraction properties of the Lyapunov function can be employed. A new solution of the average equation is initialised at time K. Since V is decreasing in N, the new solution of the average equation will be moving closer to a^*. Again, Theorem A.4 can be applied for $k \in \{0, \ldots, K\}$, to show that the solution of (2.2) has moved closer to a^* at time $2K$. This process is repeated until $\|a_k - a^*\| = o_\mu(1)$.

Proof of Theorem 2.6

Let $B^0 \subset A^0$ be compact and let $\hat{\delta} := \sup\{r > 0 : B(a^*, \delta) \subset N\}$, where N is defined in Theorem A.5. Then

$$K := \max\left\{\frac{\ln 2}{c}, \max_{a_{av}(0)\in B^0 \cup B(a^*,\hat{\delta})} \min_{k\in\mathbb{N}}\left\{k : \|a_{av}(k) - a^*\| \le \frac{\hat{\delta}}{2}\right\}\right\}$$

exists, where a_{av} is defined in (2.3). From the definition of a_{av}, it is clear that μK is independent of μ, so k is on the time scale $\frac{1}{\mu}$ if $0 \le k \le K$.

For each $n \in \mathbb{N}_0$, define b_n as the solution of (A.3) with initial value $b_n(nK) = a_{nK}$ (so $b_0(t) = a_{av}(t)$). Result 6 of Theorem A.4 implies that for each $\hat{a} \in B^0$ there exists an $o_\mu(1)$ function $l_{\hat{a}}(\mu)$ and a constant $\mu_{\hat{a}}$ such that if $\mu \le \mu_{\hat{a}}$ then

$$\|a_{nK+j} - b_n(nK + j)\| \le l_{\hat{a}}(\mu)$$

for all $j \in \{0, \ldots, K\}$. Let $\mu = \min_{\hat{a}\in B^0} \mu_{\hat{a}}$ and for each $\mu \le \mu_1$, let $l_1(\mu) = \min_{\hat{a}\in B^0} l_{\hat{a}}(\mu)$. Then for all $a_{nK} \in B^0$, if $\mu \le \mu_1$,

$$\|a_{nK+j} - b_n(nK + j)\| \le l_1(\mu) \tag{A.5}$$

for all $j \in \{0, \ldots, K\}$.

From the definitions of K and b_n, if $a_{nK} \in B^0 \cup B(a^*, \hat{\delta})$, then

$$\|b_n((n + 1)K) - a^*\| \le \frac{\hat{\delta}}{2}.$$

If $\mu \le \mu_1$ is sufficiently small, $l_1(\mu) \le \frac{\hat{\delta}}{2}$, so (A.5) implies that

$$\|a_{(n+1)K} - a^*\| \le \|a_{(n+1)K} - b_n((n + 1)K)\| + \|b_n((n + 1)K) - a^*\| \le \hat{\delta},$$

i.e. $a_{(n+1)K} \in B(a^*, \hat{\delta})$. Thus there exists $\mu_0 \le \mu_1$ such that if $\mu < \mu_0$ then $a_0 \in B^0$ implies $a_{nK} \in B(a^*, \hat{\delta})$ for all $n \in \mathbb{N}$. Thus the properties of the Lyapunov function hold for all a_{nK+j} and $b_n(nK + j)$ where $n \in \mathbb{N}$ and $j \in \{0, \ldots, K\}$.

Let $a_{nK} \in B(a^*, \hat{\delta})$. Combining property 3 of V with the Comparison Principle shows that, for $j \in \{0, \ldots, K\}$,

$$V(b_n(nK + j)) \le V(a_{nK})e^{-cj}. \tag{A.6}$$

Using the definition of K, (A.6) gives

$$V(b_n((n + 1)K)) \le \frac{1}{2}V(a_{nK}). \tag{A.7}$$

Lipschitz continuity of V implies that

$$\begin{aligned} V(a_{(n+1)K}) &\le V(b_n((n + 1)K)) + \lambda_V\|a_{(n+1)K} - b_n((n + 1)K)\| \\ &\le \frac{1}{2}V(a_{nK}) + \lambda_V l_1(\mu) \end{aligned}$$

using (A.5) and (A.7). Since $a_0 \in B^0$, this recursion yields

$$\begin{aligned} V(a_{nK}) &\leq 2^{1-n} V(a_K) + \lambda_V l_1(\mu) \sum_{i=0}^{n-2} \left(\frac{1}{2}\right)^i \\ &\leq 2^{1-n}\alpha(\hat{\delta}) + l_1(\mu). \end{aligned}$$

For any $k \in \mathbb{N}_0$,

$$\|a_k - a^*\| \leq \|a_k - b_n(k)\| + \|b_n(k) - a^*\|$$

where $n = \left\lfloor \frac{k}{K} \right\rfloor$. Using (A.5), property 1 of V, and (A.6), this gives

$$\begin{aligned} \|a_k - a^*\| &\leq l_1(\mu) + \alpha^{-1}(V(a_{nK})) \\ &\leq l_1(\mu) + \alpha^{-1}(2^{1-n}\alpha(\delta)). \end{aligned}$$

Choose k_μ such that $2^{1-\lfloor k_\mu/K \rfloor}\alpha(\hat{\delta}) \leq \alpha(l_1(\mu))$. Now $l(\mu) = 2l_1(\mu)$ is an $o_\mu(1)$ function, and $\|a_k - a^*\| \leq l(\mu)$ for all $k \geq k_\mu$. ∎

B Technical Appendix

In this appendix, we prove the inequality

$$\begin{aligned} &(1-\gamma)^{N-1}\left[1 - \left(\frac{1-\sigma}{1-\sigma\gamma}\right)^N\right]\left[\frac{(1-\gamma)^2(1-\sigma)}{1-\sigma\gamma}\right]^{(N-1)(T-1)} \\ &\quad + \frac{(1-\gamma)^{N-1}\left[(1-\sigma\gamma)^{N-1} - (1-\sigma)^{N-1}\right]}{(1-\sigma\gamma)^{N-1} - (1-\gamma)^{2(N-1)}(1-\sigma)^{N-1}}\left(1 - \left[\frac{(1-\gamma)^2(1-\sigma)}{1-\sigma\gamma}\right]^{(N-1)(T-1)}\right) \\ &\quad < 1 - (1-\sigma)^{N+(N-1)(T-1)} \end{aligned} \tag{B.1}$$

that appears in the discussion immediately following Theorem 5.1. It is assumed that $\sigma,\ \gamma \in (0,1)$, $T \in \mathbb{N}$, and $N \in \{2, 3, \dots\}$. Thus

$$0 < (1-\gamma) < 1 \tag{B.2}$$

$$0 < (1-\sigma) < 1$$

$$(1-\sigma) < (1-\sigma\gamma) < 1$$

$$(1-\sigma) < \left(\frac{1-\sigma}{1-\sigma\gamma}\right) < 1$$

$$(1-\gamma) < \left(\frac{1-\gamma}{1-\sigma\gamma}\right) < 1 \tag{B.3}$$

$$0 < \left[1 - \left(\frac{1-\sigma}{1-\sigma\gamma}\right)^N\right] < \left(1 - (1-\sigma)^N\right) \tag{B.4}$$

From relation (B.4) above, the first term in the left hand side of (B.1) is less than

$$(1-\gamma)^{(N-1)T}\left(1-(1-\sigma)^N\right)\left(\frac{1-\gamma}{1-\sigma\gamma}\right)^{(N-1)(T-1)}(1-\sigma)^{(N-1)(T-1)}$$
$$\leq \left(1-(1-\sigma)^N\right)(1-\sigma)^{(N-1)(T-1)} \tag{B.5}$$

where relations (B.2) and (B.3) above has been employed. Again using (B.2), the second term in the left hand side of (B.1) is less than

$$\left[1-\left(\frac{1-\sigma}{1-\sigma\gamma}\right)^{N-1}\right]\frac{\left[1-\left(\frac{(1-\gamma)^2(1-\sigma)}{1-\sigma\gamma}\right)^{(N-1)(T-1)}\right]}{\left[1-\left(\frac{(1-\gamma)^2(1-\sigma)}{1-\sigma\gamma}\right)^{N-1}\right]} \tag{B.6}$$

Expression B.6 is of the form $(1-b)\frac{1-(ab)^t}{1-ab}$. Since $a,\ b \in (0,1)$,

$$\frac{1-(ab)^t}{1-ab} = \sum_{i=0}^{t-1}(ab)^i < \sum_{i=0}^{t-1} b^i = \frac{1-b^t}{1-b}.$$

The new denominator cancels the first factor in (B.6), so (B.6) is less than

$$\left[1-\left(\frac{1-\sigma}{1-\sigma\gamma}\right)^{(N-1)(T-1)}\right] < 1-(1-\sigma)^{(N-1)(T-1)} \tag{B.7}$$

Combining (B.5) with (B.7) gives the result.