Particle Swarm Optimization: A Stochastic Approximation Approach

Quan Yuan† G. Yin‡

Abstract

Recently, much progress has been made on particle swarm optimization (PSO). A number of works have been devoted to analyzing the convergence of the underlying algorithms. Nevertheless, in most cases, certain rather simplified hypotheses are used. For example, it often assumes that the swarm has only one particle. In addition, more often than not, the variables and the points of attraction are assumed to remain constant throughout the optimization process. In reality, such assumptions are often violated. Moreover, up to now, not much is known regarding the convergence rates of particle swarm. In this paper, we develop a class of PSO algorithms, and analyze asymptotic properties of the algorithms using stochastic approximation methods. We introduce four coefficients and rewrite the PSO procedure as a stochastic approximation type recursive algorithm. Then we analyze its convergence using weak convergence method. It is proved that a suitably scaled sequence of swarms converge to the solution of an ordinary differential equation. We also establish certain stability results. Moreover, convergence rates are ascertained by using weak convergence method. A centered and scaled sequence of the estimation errors is shown to have a diffusion limit. Furthermore, we demonstrate that our PSO algorithms perform much better than the traditional PSOs on some test functions for optimization problems.

Key Words. Particle swarm optimization, stochastic approximation, weak convergence, rate of convergence.

†Department of Mathematics, Wayne State University, Detroit, MI 48202. Email: quanyuan@wayne.edu
‡Department of Mathematics, Wayne State University, Detroit, MI 48202. Email: gyin@math.wayne.edu
1 Introduction

Recently, optimization using particle swarms have received considerable attention owing to the wide range of applications from networked systems, to multi-agent systems, and to autonomous systems. Swarm intelligence from bio-cooperation within groups of individuals can often provide efficient solutions for certain optimization problems. When birds are searching food, they exchange and share information. Each member benefits from all other members owing to their discovery and experience based on the information acquired locally. Then each participating member adjusts the next search direction in accordance with the individual’s best position currently and the information communicated to this individual by its neighbors. When food sources scattered unpredictably, advantages of such collaboration was decisive. Inspired by this, Kennedy and Eberhart proposed a particle swarm optimization (PSO) algorithm in 1995 [1]. A PSO procedure is a stochastic optimization algorithm that mimics the foraging behavior of birds. The search space of the optimization problem is analogous to the flight space of birds. Using an abstract setup, each bird is modeled as a particle (a point in the space of interest). Finding the optimum is the counterpart of searching for food. A PSO can be carried out effectively by using a recursive scheme. As a recursive algorithm, the PSO algorithm simulates social behavior among individuals (particles) “flying” through a multidimensional search space, where each particle represents a point at the intersection of all search dimensions. The particles evaluate their positions according to certain fitness functions at each iteration. The particles share memories of their “best” positions locally, and use the memories to adjust their own velocities and positions. Motivated by this scenario, a model is proposed to represent the traditional dynamics of particles.

To put this in a mathematical form, let $F : \mathbb{R}^D \to \mathbb{R}$ be the cost function to be minimized. If we let $M$ denote the size of the swarm, the current position of particle $i$ is denoted by $X^i$ ($i = 1, 2, \ldots, M$), and its current velocity is denoted by $v^i$. Then, the updating principle can be expressed as

$$
\begin{align*}
    v_{n+1}^{i,d} &= v_n^{i,d} + c_1 r_{1,n}^{i,d} [P_{r}^{i,d} - X_n^{i,d}] + c_2 r_{2,n}^{i,d} [P_{g}^{i,d} - X_n^{i,d}], \\
    X_{n+1}^{i,d} &= X_n^{i,d} + v_{n+1}^{i,d},
\end{align*}
$$

where $d = 1, \ldots, D$; $r_{1}^{i,d} \sim U(0,1)$ and $r_{2}^{i,d} \sim U(0,1)$ represent two random variables uniformly distributed in $[0,1]$; $c_1$ and $c_2$ represent the acceleration coefficients; $P_{r}^{i,d}$ represents
the best position found by particle \( i \) up to “time” \( n \), and \( P^i_n \) represents the “global” best position found by particle \( i \)’s neighborhood \( \Pi_j \), i.e.,

\[
\begin{align*}
Pr^i_n &= \arg\min_{1 \leq k \leq n} F(X^i_k), \\
Pg^i_n &= \arg\min_{j \in \Pi_j} F(Pr^j_n).
\end{align*}
\]

In artificial life and social psychology, \( v^i_n \) in (1) is the velocity of particle \( i \) at time \( n \), which provides the momentum for particles to pass through the search space. The \( c_1 r_{1,n}^{i,d}[Pr^i_n - X^i_n] \) is named as the “cognitive” component, which represents the personal thinking of each particle. The cognitive component of a particle takes the best position found so far by this particle as the desired input to make the particle move toward its own best positions. \( c_2 r_{2,n}^{i,d}[Pg^i_n - X^i_n] \) is named as the “social” component, which represents the collaborative behavior of the particles to find the global optimal solution. The social component always pulls the particles toward the best position found by its neighbors.

In a nutshell, a PSO algorithm has the following advantages: (1) It has versatility and does not rely on the problem information; (2) it has a memory capacity to retain local and global optimal information; (3) it is easy to implement. Given the versatility and effectiveness of PSO, it is widely used to solve practical problems such as artificial neural networks [2, 3], chemical systems [4], power systems [5, 6], mechanical design [7], communications [8], robotics [9, 10], economy [11, 12], image processing [13], bio-informatics [14, 15], medicine [16], and industrial engineering [17, 18]. Note that swarms have also been used in many engineering applications, for example, in collective robotics where there are teams of robots working together by communicating over a communication network; see [20] for a stability analysis and many related references.

To enable and to enhance further applications, much work has also been devoted to improving the PSO algorithms. Because the original model is similar to a mobile multi-agent system and each parameter describes a special character of natural swarm behavior, one can improve the performance of PSO according to the physical meanings of these parameters [19, 21–24]. The first significant improvement was proposed by Shi and Eberhart in [25]. They suggested to add a new parameter \( w \) as an “inertia constant”, which results in fast convergence. The modified equation of (1) is

\[
\begin{align*}
v^{i,d}_{n+1} &= w v^{i,d}_n + c_1 r_{1,n}^{i,d}[Pr^i_n - X^i_n] + c_2 r_{2,n}^{i,d}[Pg^i_n - X^i_n], \\
X^{i,d}_{n+1} &= X^{i,d}_n + v^{i,d}_{n+1}.
\end{align*}
\]
Another significant improvement was due to Clerc and Kennedy [26]. They introduced a constricted coefficient \( \chi \) and then proposed to modify (1) as

\[
v_{i,d}^{n+1} = \chi \left( v_{i,d}^n + c_1 r_{1,n}^i [P_{r_i,d} - X_{i,d}^n] + c_2 r_{2,n}^i [P_{g_i,d} - X_{i,d}^n] \right),
\]

\[
X_{i,d}^{n+1} = X_{i,d}^n + v_{i,d}^{n+1}.
\]

This constricted coefficient can control the “explosion” of the PSO and ensure the convergence. In fact, almost all improvements about PSO are based on these two basic improvements.

Another significant development is the progress on mathematical analysis for the convergence of PSO algorithms. Although most researchers prefer to use discrete system [26–29], there are some works on continuous-time models [30, 31]. Their work has resulted in guidelines for selecting PSO parameters leading to convergence, divergence, or oscillation of the swarm’s particles, and their work also given rise to several PSO variants. However, as criticized by Pedersen [32], the analysis is often oversimplified that the swarm is assumed to have only one particle, that it does not use stochastic variables (namely, \( r_{1,n}, r_{2,n} \)), and that the points of attraction, i.e., the particle’s best known position \( P_r \) and the swarm’s best known position \( P_g \), remain constant throughout the optimization process. It is widely recognized that purely deterministic approach is inadequate in reflecting the exploration and exploitation aspects brought by stochastic variables.

In this paper, we study convergence properties of PSO by stochastic approximation theory. To the best of our knowledge, the only paper using stochastic approximation methods to analyze the dynamics of the PSO so far is by Chen and Li [33]. They designed a special PSO procedure and assumed

(i) \( P_{r_i}^n \) and \( P_{g_i}^n \) are always within a finite domain;

(ii) with \( P^* \) representing the global optimal positions in the solution space, and \( ||P^*|| < \infty \).

\[
\lim_{n \to \infty} P_{r_i}^n \to P^* \quad \text{and} \quad \lim_{n \to \infty} P_{g_i}^n \to P^*.
\]

Using assumption (i), they proved the convergence of in the sense of with probability one. With additional assumption (ii), they showed that the swarm will converge to \( P^* \).

Despite the interesting development, their assumptions (i) and (ii) appear to be rather strong. Moreover, they added some specific terms in the PSO procedure. So their algorithm is different from the traditional PSOs (1)-(3). In this paper, we develop another class of
stochastic algorithms. We introduce four coefficients \( \varepsilon, \kappa_1, \kappa_2, \) and \( \chi \) and rewrite the PSOs in a stochastic approximation setup. Then we analyze its convergence using weak convergence method. We prove that a suitably interpolated sequence of swarms converge to the solution of an ordinary differential equation. Moreover, convergence rates are derived by using a centered and scaled sequence of the estimation errors. Furthermore, we show that the PSO performs better than traditional PSOs ((1)-(3)) on some test functions for optimization.

The remainder of the paper is arranged as follows. Section 2 presents the setup of our algorithm. Section 3 is devoted to studying the convergence of the algorithm. Section 4 presents the analysis on rate of convergence of the algorithm. Section 5 proceeds with several numerical simulation examples to demonstrate the utility of our algorithms. Finally, Section 6 provides a few further remarks.

## 2 Formulation

First, some descriptions on notation are in order. We use \(| \cdot |\) to denote a Euclidean norm. A point \( \theta \) in a Euclidean space is a column vector; the \( i \)th component of \( \theta \) is denoted by \( \theta^i \); \( \text{diag}(\theta) \) is a diagonal matrix whose diagonal elements are the elements of \( \theta \); \( I \) denotes the identity matrix of appropriate dimension; \( z' \) denotes the transposition of \( z \). \( O(y) \) denotes a function of \( y \) satisfying \( \sup_y |O(y)|/|y| < \infty \), and \( o(y) \) denotes a function of \( y \) satisfying \( |o(y)|/|y| \to 0 \), as \( y \to 0 \). In particular, \( O(1) \) denotes the boundedness and \( o(1) \) indicates convergence to 0. Throughout the paper, we use \( K \) to denote a generic positive constant with the convention \( K + K = K \) and \( K K = K \).

In this paper, without loss of generality, we assume that each particle is a one-dimensional scalar. Note that each particle can be a multi-dimensional vector, which does not introduce essential difficulties in the analysis; only the notation is a bit more complex. We introduce four parameters \( \varepsilon, \kappa_1, \kappa_2, \) and \( \chi \). Suppose there are \( r \) particles, then the PSO algorithm can be expressed as

\[
\begin{bmatrix}
  v_{n+1} \\
  X_{n+1}
\end{bmatrix} = \begin{bmatrix}
  v_n \\
  X_n
\end{bmatrix} + \varepsilon \begin{bmatrix}
  \kappa_1 I & -\chi(c_1\text{diag}(r_{1,n}) + c_2\text{diag}(r_{2,n})) \\
  \kappa_2 I & -\chi(c_1\text{diag}(r_{1,n}) + c_2\text{diag}(r_{2,n}))
\end{bmatrix} \begin{bmatrix}
  v_n \\
  X_n
\end{bmatrix} + \chi \begin{bmatrix}
  c_1\text{diag}(r_{1,n}) & c_2\text{diag}(r_{2,n}) \\
  c_1\text{diag}(r_{1,n}) & c_2\text{diag}(r_{2,n})
\end{bmatrix} \begin{bmatrix}
  \text{Pr}(\theta_n, \eta_n) \\
  \text{Pg}(\theta_n, \eta_n)
\end{bmatrix},
\]

where \( X_n = [X_{n,1}, \ldots, X_{n,r}]' \in \mathbb{R}^r \), \( v_n = [v_{n,1}', \ldots, v_{n,r}'] \in \mathbb{R}^r \), \( \theta_n = (X_n, v_n)' \), \( r_1, r_2 \) are \( r \)-dimensional random vectors in which each component is uniformly distributed in \((0, 1)\), and
Pr(θ, η) and Pg(θ, η) are two non-linear functions depending on θ = (X, v)' as well as on a “noise” η, and ε > 0 is a small parameter representing the stepsize of the iterations.

**Remark 1** If there is no noise term η, let ε = 0.01, χ = 72.9, κ₁ = −27.1, and κ₂ = 72.9, then (4) is equivalent to (2) when w = 0.729 or (3) when χ = 0.729. Thus (4) is a generalization of (1)-(3).

In (4), r₁ and r₂ are used to reflect the exploration of particles. Rearranging terms of (4) and considering that $E[c₁ \text{diag}(r₁,n)] = 0.5c₁I$ and $E[c₂ \text{diag}(r₂,n)] = 0.5c₂I$, it can be rewritten as

$$
\begin{bmatrix}
    v_{n+1} \\
    X_{n+1}
\end{bmatrix} = \begin{bmatrix}
    v_n \\
    X_n
\end{bmatrix} + \varepsilon \left\{ \begin{bmatrix}
    \kappa₁I & -0.5\chi(c₁ + c₂)I \\
    \kappa₂I & -0.5\chi(c₁ + c₂)I
\end{bmatrix} \begin{bmatrix}
    v_n \\
    X_n
\end{bmatrix} + \chi \begin{bmatrix}
    0 & -c₁ \text{diag}(r₁,n) - 0.5c₁I - 0.5c₂I \\
    0 & -c₂ \text{diag}(r₂,n) - 0.5c₁I - 0.5c₂I
\end{bmatrix} \begin{bmatrix}
    v_n \\
    X_n
\end{bmatrix} + \chi \begin{bmatrix}
    c₁ \text{diag}(r₁,n) - 0.5c₁I & c₂ \text{diag}(r₂,n) - 0.5c₂I \\
    c₂ \text{diag}(r₁,n) - 0.5c₁I & c₂ \text{diag}(r₂,n) - 0.5c₂I
\end{bmatrix} \begin{bmatrix}
    \text{Pr}(θₙ, ηₙ) \\
    \text{Pg}(θₙ, ηₙ)
\end{bmatrix} \right\}.
$$

Denote

$$
θₙ = [v_n, X_n]' \in \mathbb{R}^{2r},
$$

$$
M = \begin{bmatrix}
    \kappa₁I & -0.5\chi(c₁ + c₂)I \\
    \kappa₂I & -0.5\chi(c₁ + c₂)I
\end{bmatrix},
$$

$$
P(θₙ, ηₙ) = \chi \begin{bmatrix}
    0.5c₁I & 0.5c₂I \\
    0.5c₁I & 0.5c₂I
\end{bmatrix} \begin{bmatrix}
    \text{Pr}(θₙ, ηₙ) \\
    \text{Pg}(θₙ, ηₙ)
\end{bmatrix},
$$

and $W(θₙ, r₁,n, r₂,n, ηₙ)$ to be the sum of the last two terms in the curly braces of (5). Then (5) can be expressed as a stochastic approximation algorithm

$$
θ_{n+1} = θₙ + \varepsilon[Mθₙ + P(θₙ, ηₙ) + W(θₙ, r₁,n, r₂,n, ηₙ)].
$$

We shall use the following assumptions.

(A1) The $\text{Pr}(·, η)$ and $\text{Pg}(·, η)$ are continuously differentiable for each η. For each bounded θ, $E|P(θ, ηₙ)|² < ∞$ and $E|W(θ, r₁,n, r₂,n, ηₙ)|² < ∞$. There exist continuous functions $\overline{\text{Pr}}(θ)$ and $\overline{\text{Pg}}(θ)$ such that

$$
\frac{1}{n} \sum_{j=m}^{n+m-1} E_m \text{Pr}(θ, ηₖ) \rightarrow \overline{\text{Pr}}(θ) \text{ in probability},
$$

$$
\frac{1}{n} \sum_{j=m}^{n+m-1} E_m \text{Pg}(θ, ηₖ) \rightarrow \overline{\text{Pg}}(θ) \text{ in probability},
$$

(7)
where $E_m$ denotes the conditional expectation on the $\sigma$-algebra $F_m = \{\theta_0, r_{i,j}, i = 1, 2, \eta_j : j < m\}$. Moreover, for each $\theta$ in a bounded set,

$$
\sum_{j=n}^{\infty} |E_n \Pr(\theta, \eta_j) - \overline{\Pr}(\theta)| < \infty,
$$

(8)

$$
\sum_{j=n}^{\infty} |E_n \Pr(\theta, \eta_j) - \overline{\Pr}(\theta)| < \infty.
$$

(A2) Define

$$
\overline{\Pr}(\theta) = \chi \begin{bmatrix} 0.5c_1 I & -0.5c_2 I \\ 0.5c_1 I & -0.5c_2 I \end{bmatrix} \begin{bmatrix} \overline{\Pr}(\theta) \\ \overline{\Pr}(\theta) \end{bmatrix}.
$$

The ordinary differential equation

$$
\frac{d\theta(t)}{dt} = M\theta(t) + \overline{\Pr}(\theta(t))
$$

has a unique solution for each initial condition $\theta(0) = (\theta_1, \ldots, \theta_2)^\prime$. This solution is asymptotically stable.

(A3) $\{r_{1,n}\}$, $\{r_{2,n}\}$, and $\{\eta_n\}$ are mutually independent.

Remark 2 Note that the noise treated here is much more general than that of the i.i.d. sequences; the noise observations are also subject to general non-additive noise. In (A1), if $\Pr(\theta, \eta) = \Pr(\theta) + \eta$, then the condition is mainly on the noise sequence $\{\eta_n\}$. It is verified by a large class of random variables. For example, it is trivially satisfied for i.i.d. zero mean noise. It is also satisfied for a large class of correlated noise.

3 Convergence

This section is devoted to obtaining asymptotic properties of algorithm (6). In relation to PSO the word convergence typically means one of two things, although it is often not clarified which definition is meant and sometimes they are mistakenly thought to be identical.

- Convergence may refer to the swarm’s best known position $Pg$ approaching (converging to) the optimum of the problem, regardless of how the swarm behaves.

- Convergence may refer to a swarm collapse in which all particles have converged to a point in the search space, which may or may not be the optimum.
We use the second one as the definition of convergence in this study.

The first result concerns the property of the algorithm as \( \varepsilon \to 0 \) through an appropriate continuous-time interpolation. We define

\[
\theta^\varepsilon(t) = \theta_n \text{ for } t \in [\varepsilon n, \varepsilon n + \varepsilon).
\]

Then \( \theta^\varepsilon(\cdot) \in D([0, T] : \mathbb{R}^{2r}) \), which is the space of functions that are defined on \([0, T]\) taking values in \( \mathbb{R}^{2r} \), and that are right continuous and have left limits endowed with the Skorohod topology [34, Chapter 7].

**Theorem 3** Under (A1)-(A3), \( \theta^\varepsilon(\cdot) \) is tight in \( D([0, T] : \mathbb{R}^{2r}) \). Moreover, as \( \varepsilon \to 0 \), \( \theta^\varepsilon(\cdot) \) converges weakly to \( \theta(\cdot) \), which is a solution of (9).

**Remark 4** An equivalent way of stating the limit in the ODE limit (9) is to consider its associated martingale problem. Consider the differential operator associated with \( \theta(\cdot) \) given by

\[
\mathcal{L}f(\theta) = (\nabla f(\theta))'(M\theta + \overline{P}(\theta)).
\]

Define

\[
\tilde{M}_f(t) = f(\theta(t)) - f(\theta(0)) - \int_0^t \mathcal{L}f(\theta(s))ds.
\]

If \( \tilde{M}_f(\cdot) \) is a martingale for each \( f(\cdot) \in C^1_0 \) (\( C^1 \) function with compact support), then \( \theta(\cdot) \) is said to solve a martingale problem with operator \( \mathcal{L} \). Thus, an equivalent way to state the theorem is to prove that \( \theta^\varepsilon(\cdot) \) converges weakly to \( \theta(\cdot) \), which is a solution of the martingale problem with operator \( \mathcal{L} \).

**Proof of Theorem 3.** To prove the tightness in \( D([0, T] : \mathbb{R}^{2r}) \), we need to prove

\[
\lim_{K \to \infty} \limsup_{\varepsilon \to 0} P\left\{ \sup_{t \leq T} |\theta^\varepsilon(t)| \geq K \right\} = 0 \tag{10}
\]

To avoid verifying (10), we define a process \( \theta^{\varepsilon,N}(\cdot) \) satisfies \( \theta^{\varepsilon,N}(t) = \theta^\varepsilon(t) \) up until the first exit from \( S_N = \{x \in \mathbb{R}^{2r} : |x| \leq N\} \) and satisfies (10), the \( \theta^{\varepsilon,N}(\cdot) \) is said to be an \( N \)-truncation of \( \theta^\varepsilon(\cdot) \). Introduce a truncation function \( q^N(\cdot) \) that is smooth and that satisfies \( q^N(\theta) = 1 \) for \( |\theta| \leq N \), \( q^N(\theta) = 0 \) for \( |\theta| \geq N + 1 \). Then the discrete system (6) is defined as

\[
\theta_{n+1}^N = \theta_n^N + \varepsilon[M\theta_n^N + P(\theta_n^N, \eta_n) + W(\theta_n^N, r_{1,n}, r_{2,n}, \eta_n)]q^N(\theta_n^N), \tag{11}
\]
using the $N$-truncation. Moreover, the $N$-truncated ODE and the operator $L^N$ of the associated martingale problem can be defined as

$$\frac{d\theta^N(t)}{dt} = [M\theta^N(t) + P(\theta^N(t))]q^N(\theta(t)),\quad (12)$$

and

$$L^N f(\theta) = (\nabla f(\theta))'[M\theta + P(\theta)]q^N(\theta),\quad (13)$$

respectively.

To prove the theorem, we carry out the following steps. First, we verify that

(a) for each $N$, $\{\theta^{\varepsilon,N}(\cdot)\}$ is tight.

By virtue of the Prohorov theorem [34, p.229], we can extract a weakly convergent subsequence. For notation simplicity, we still denote the subsequence by $\{\theta^{\varepsilon,N}(\cdot)\}$ with limit denoted by $\theta^N(\cdot)$. Then we show

(b) $\theta^N(\cdot)$ is a solution of the martingale problem with operator $L^N$.

Using the uniqueness of the limit, passing to the limit as $N \to \infty$, and by the corollary in [35, p.44], $\{\theta^\varepsilon(\cdot)\}$ converges weakly to $\theta(\cdot)$. Now we will prove the claims (a) and (b).

(a) Tightness. For any $\delta > 0$, let $t > 0$ and $s > 0$ such that $s \leq \delta$, and $t, t + \delta \in [0, T]$.

Note that

$$\theta^{\varepsilon,N}(t + s) - \theta^{\varepsilon,N}(t) = \varepsilon \sum_{k=t/\varepsilon}^{(t+s)/\varepsilon-1} (M\theta^N_k + P(\theta^N_k, \eta_k) + W(\theta^N_k, r_{1,k}, r_{2,k}, \eta_k))q^N(\theta^N_k).$$

In the above and hereafter, we use the conventions that $t/\varepsilon$ and $(t + s)/\varepsilon$ denote the corresponding integer parts $[t/\varepsilon]$ and $[(t + s)/\varepsilon]$, respectively. For notational simplicity, in what follows, we will not use the floor function notation unless it is necessary.

Using the Cauchy-Schwarz inequality,

$$\varepsilon^2 \sum_{k=t/\varepsilon}^{(t+s)/\varepsilon-1} M\theta^N_k q^N(\theta^N_k) \leq \varepsilon^2 \left( \sum_{k=t/\varepsilon}^{(t+s)/\varepsilon-1} 1^2 \right)^2 \left( \sum_{k=t/\varepsilon}^{(t+s)/\varepsilon-1} |M|^2 |\theta^N_k q^N(\theta^N_k)|^2 \right),$$

so

$$\varepsilon^2 E_t \left( \sum_{k=t/\varepsilon}^{(t+s)/\varepsilon-1} M\theta^N_k q^N(\theta^N_k) \right)^2 \leq \varepsilon K s \sum_{k=t/\varepsilon}^{(t+s)/\varepsilon-1} E_t \left( |\theta^N_k q^N(\theta^N_k)|^2 \right).$$

15
where $E^\varepsilon_t$ denotes the conditional expectation on $\mathcal{F}^\varepsilon_t$ $\sigma$-algebra. Likewise,

$$
\varepsilon^2 E^\varepsilon_t \left| \sum_{k=t/\varepsilon}^{(t+s)/\varepsilon-1} W(\theta_k^N, r_{1,k}, r_{2,k}, \eta_k) q^N(\theta_k^N) \right|^2 \leq K s^2, \quad (16)
$$

and

$$
\varepsilon^2 E^\varepsilon_t \left| \sum_{k=t/\varepsilon}^{(t+s)/\varepsilon-1} P(\theta_k^N, \eta_k) q^N(\theta_k^N) \right|^2 \leq K s^2. \quad (17)
$$

So we have

$$
E^\varepsilon_t \left| \theta^\varepsilon,N(t + s) - \theta^\varepsilon,N(t) \right|^2 \leq K \varepsilon^2 E^\varepsilon_t \left| \sum_{k=t/\varepsilon}^{(t+s)/\varepsilon-1} M \theta_k^N q^N(\theta_k^N) \right|^2
$$

$$
+ K \varepsilon^2 E^\varepsilon_t \left| \sum_{k=t/\varepsilon}^{(t+s)/\varepsilon-1} P(\theta_k^N, \eta_k) q^N(\theta_k^N) \right|^2 + K \varepsilon^2 E^\varepsilon_t \left| \sum_{k=t/\varepsilon}^{(t+s)/\varepsilon-1} W(\theta_k^N, r_{1,k}, r_{2,k}, \eta_k) q^N(\theta_k^N) \right|^2
$$

$$
\leq K \varepsilon s \sum_{k=t/\varepsilon}^{(t+s)/\varepsilon-1} \sup_{t/\varepsilon \leq k \leq (t+s)/\varepsilon-1} E^\varepsilon_t \left| \theta_k^N q^N(\theta_k^N) \right|^2 + K s^2
$$

$$
\leq K \delta^2. \quad (18)
$$

As a result, there is a $\varsigma^\varepsilon(\delta)$ such that

$$
E^\varepsilon_t \left| \theta^\varepsilon,N(t + s) - \theta^\varepsilon,N(t) \right|^2 \leq E^\varepsilon_t \varsigma(\varepsilon, \delta) \quad \text{for all} \quad 0 \leq s \leq \delta
$$

and that

$$
\lim_{\delta \to 0} \limsup_{\varepsilon \to 0} E \varsigma^\varepsilon(\delta) = 0.
$$

The tightness of $\{\theta^\varepsilon,N(\cdot)\}$ then follows from [35, p.47].

(b) Characterization of the limit. To characterize the limit process, we need to work with a continuously differentiable function with compact support $f(\cdot)$. Choose $m_\varepsilon$ so that
$m_e \to \infty$ but $\delta_e = \varepsilon m_e \to 0$. Using the recursion (11),

$$f(\theta^N_{t+s}(t)) - f(\theta^N_{t}(t)) = \sum_{l=t/\delta_e}^{(t+s)/\delta_e} [f(\theta^N_{lm_e+m_e}) - f(\theta^N_{lm_e})]$$

$$= \varepsilon \sum_{l=t/\delta_e}^{(t+s)/\delta_e} (\nabla f(\theta^N_{lm_e}))' \sum_{k=lm_e}^{lm_e+m_e-1} [M\theta^N_k + \mathcal{P}(\theta^N_k)]q^N(\theta^N_k)$$

$$+ \varepsilon \sum_{l=t/\delta_e}^{(t+s)/\delta_e} (\nabla f(\theta^N_{lm_e}))' \sum_{k=lm_e}^{lm_e+m_e-1} [P(\theta^N_k, \eta_k) - \mathcal{P}(\theta^N_k)]q^N(\theta^N_k)$$

$$+ \varepsilon \sum_{l=t/\delta_e}^{(t+s)/\delta_e} \left\{ (\nabla f(\theta^N_{lm_e}) - \nabla f(\theta^N_{lm_e}))' \sum_{k=lm_e}^{lm_e+m_e-1} [M\theta^N_k + P(\theta^N_k, \eta_k)] \right\},$$

where $\theta^N_{lm_e}$ is a point on the line segment joining $\theta^N_{lm_e}$ and $\theta^N_{lm_e+m_e}$.

Our focus here is to characterize the limit. By the Skorohod representation [34, p.230], with a slight abuse of notation, we may assume that $\theta^N(\cdot)$ converges to $\theta^N(\cdot)$ with probability one and the convergence is uniform on any bounded time interval. To show that $\{\theta^N(\cdot)\}$ is a solution of the martingale problem with operator $\mathcal{L}^N$, it suffices to show that for any $f(\cdot) \in C^1_0$, the class of functions that are continuously differentiable with compact support,

$$\tilde{M}^N_I(t) = f(\theta^N(t)) - f(\theta^N(0)) - \int_0^t \mathcal{L}^N f(\theta^N(u))du$$

is a martingale. To verify the martingale property, we need only show that for any bounded and continuous function $h(\cdot)$, any positive integer $\kappa$, any $t, s > 0$, and $t_i \leq t$ with $i \leq \kappa$,

$$E h(\theta^N(t_i) : i \leq \kappa)[\tilde{M}^N_I(t + s) - \tilde{M}^N_I(t)]$$

$$= E h(\theta^N(t_i) : i \leq \kappa)[f(\theta^N(t + s)) - f(\theta^N(t))] - \int_t^{t+s} \mathcal{L}^N f(\theta^N(u))du$$

$$= 0.$$ (20)

To verify (20), we begin with the process indexed by $\varepsilon$. For notational simplicity, denote

$$\tilde{h} = h(\theta^N(t_i) : i \leq \kappa), \quad \tilde{h}^\varepsilon = h(\theta^\varepsilon N(t_i) : i \leq \kappa).$$ (21)
Then the weak convergence and the Skorohod representation together with the boundedness and the continuity of \(f(\cdot)\) and \(h(\cdot)\) yield that as \(\varepsilon \to 0\),

\[
E\tilde{h}^\varepsilon[f(\theta^\varepsilon_N(t + s)) - f(\theta^\varepsilon_N(t))] \to E\tilde{h}[f(\theta^N(t + s)) - f(\theta^N(t))].
\]

For the last term of (19), as \(\varepsilon \to 0\), since \(f(\cdot) \in C^1_0\),

\[
\varepsilon \sum_{l=t/\delta}^{(t+s)/\delta} \left\{(\nabla f(\theta^N_{lm}))' \sum_{k=lm}^{lm+\varepsilon-1} [M\theta^N_k + P(\theta^N_k, \eta_k)] + W(\theta^N_k, r_{1,k}, r_{2,k}, \eta_k)]q^N(\theta^N_k)\right\}
\leq \varepsilon \sum_{l=t/\delta}^{(t+s)/\delta} \varepsilon \cdot Km_{\varepsilon} = \varepsilon^2 Km_{\varepsilon}(s/\delta + 1) = O(\varepsilon) \to 0.
\]

For the next to the last term,

\[
\lim_{\varepsilon \to 0} E\tilde{h}^\varepsilon \left[ \varepsilon \sum_{l=t/\delta}^{(t+s)/\delta} (\nabla f(\theta^N_{lm}))' \sum_{k=lm}^{lm+\varepsilon-1} W(\theta^N_k, r_{1,k}, r_{2,k}, \eta_k)q^N(\theta^N_k) \right]
= \lim_{\varepsilon \to 0} E\tilde{h}^\varepsilon \left[ \varepsilon \sum_{l=t/\delta}^{(t+s)/\delta} (\nabla f(\theta^N_{lm}))' \sum_{k=lm}^{lm+\varepsilon-1} E_{lm}W(\theta^N_k, r_{1,k}, r_{2,k}, \eta_k)q^N(\theta^N_k) \right]
= \lim_{\varepsilon \to 0} E\tilde{h}^\varepsilon \left[ \varepsilon \sum_{l=t/\delta}^{(t+s)/\delta} (\nabla f(\theta^N_{lm}))' \delta_{\varepsilon}/m_{\varepsilon} \sum_{k=lm}^{lm+\varepsilon-1} E_{lm}W(\theta^N_k, r_{1,k}, r_{2,k}, \eta_k)q^N(\theta^N_k) \right].
\]

Since for all \(lm_{\varepsilon} \leq k \leq lm_{\varepsilon} + m_{\varepsilon} - 1\),

\[
\frac{1}{m_{\varepsilon}} \sum_{j=lm_{\varepsilon}}^{k} E_{lm_{\varepsilon}}W(\theta^N_{lm_{\varepsilon}}, r_{1,j}, r_{2,j}, \eta_j)q^N(\theta^N_{lm_{\varepsilon}}) \to 0 \text{ in probability},
\]

we obtain that

\[
E\tilde{h}^\varepsilon \left[ \varepsilon \sum_{l=t/\delta}^{(t+s)/\delta} (\nabla f(\theta^N_{lm}))' \sum_{k=lm_{\varepsilon}}^{lm_{\varepsilon}+\varepsilon-1} W(\theta^N_k, r_{1,k}, r_{2,k}, \eta_k)q^N(\theta^N_k) \right] \to 0.
\]

Using (A1), we obtain

\[
E\tilde{h}^\varepsilon \left[ \varepsilon \sum_{l=t/\delta}^{(t+s)/\delta} (\nabla f(\theta^N_{lm}))' \sum_{k=lm_{\varepsilon}}^{lm_{\varepsilon}+\varepsilon-1} (P(\theta^N_k, \eta_k) - \overline{P}(\theta^N_k))q^N(\theta^N_k) \right]
= E\tilde{h}^\varepsilon \left[ \varepsilon \sum_{l=t/\delta}^{(t+s)/\delta} (\nabla f(\theta^N_{lm}))' \delta_{\varepsilon}/m_{\varepsilon} \sum_{k=lm_{\varepsilon}}^{lm_{\varepsilon}+\varepsilon-1} E_{lm_{\varepsilon}}(P(\theta^N_k, \eta_k) - \overline{P}(\theta^N_k))q^N(\theta^N_k) \right] \to 0.
\]
At last, we consider the first term. We have

$$
\lim_{\varepsilon \to 0} \tilde{E}h^\varepsilon \left[ \varepsilon \sum_{l=t/\delta_\varepsilon}^{(t+s)/\delta_\varepsilon} (\nabla f(\theta_{lm_\varepsilon}))' \sum_{k=lm_\varepsilon}^{lm_\varepsilon+m_\varepsilon-1} (M\theta_k^N + \overline{P}(\theta_k^N))q^N(\theta_k^N) \right] 
$$

Thus, to get the desired limit, we need only examine the last line above. Let \( \varepsilon, m_\varepsilon \to u \) as \( \varepsilon \to 0 \). Then for all \( k \) satisfying \( lm_\varepsilon \leq k \leq lm_\varepsilon + m_\varepsilon - 1 \), \( \varepsilon k \to u \) since \( \delta_\varepsilon \to 0 \). Thus

$$
\lim_{\varepsilon \to 0} \tilde{E}h^\varepsilon \left[ \varepsilon \sum_{l=t/\delta_\varepsilon}^{(t+s)/\delta_\varepsilon} (\nabla f(\theta_{lm_\varepsilon}))' \sum_{k=lm_\varepsilon}^{lm_\varepsilon+m_\varepsilon-1} (M\theta_k^N + \overline{P}(\theta_k^N))q^N(\theta_k^N) \right] 
$$

The desired result then follows. \( \square \)

We will show the condition the equilibrium should satisfy. Suppose \( \overline{P}(\theta^*) = Pr^* \) and \( \overline{P}_g(\theta^*) = Pg^* \). By the inverse formula of partition matrix [36], the equilibria of (9) satisfy

$$
\theta^* = \begin{bmatrix}
\kappa_1 I & -0.5\chi(c_1 + c_2)I \\
\kappa_2 I & -0.5\chi(c_1 + c_2)I
\end{bmatrix}^{-1} \begin{bmatrix}
-0.5\chi(c_1 Pr^* + c_2 Pg^*) \\
-0.5\chi(c_1 Pr^* + c_2 Pg^*)
\end{bmatrix} = \begin{bmatrix}
0 \\
\frac{c_1 Pr^* + c_2 Pg^*}{c_1 + c_2}
\end{bmatrix}.
\quad(28)
$$

4 Rate of Convergence

Once the convergence of a stochastic approximation algorithm is established, the next task is to ascertain the convergence rate. Since the randomness is taken into consideration, as in the investigation of convergence, the rate of convergence study is very different from any purely deterministic optimization algorithms. To study the convergence rate, we take a suitably scaled sequence

$$
z_n = (\theta_n - \theta^*) / \varepsilon^\alpha,
$$

for some \( \alpha > 0 \). The idea is to choose \( \alpha \) such that \( z_n \) converges (in distribution) to a nontrivial limit. The scaling factor \( \alpha \) together with the asymptotic covariance of the scaled sequence gives us the rate of convergence. That is, the scaling tells us the dependence of the estimation error \( \theta_n - \theta^* \) on the step size, and the asymptotic covariance is a mean of assessing
“goodness” of the approximation. Here the factor $\alpha = 1/2$ is used. To some extent, this is dictated by the well-known central limit theorem. For related work on convergence rate of various stochastic approximation algorithms, see [37, 38].

As mentioned above, by using the definition of the rate of convergence, we are effectively dealing with convergence in the distributional sense. In lieu of examining the discrete iteration directly, we are again taking continuous-time interpolations. Three assumptions are provided in what follows.

(A4) The following conditions hold:

(i) The second derivatives (with respect to $\theta$) of $W(\cdot, r_1, r_2, \eta)$ and $P(\cdot, \eta)$ exist and are continuous.

(ii) for each positive integer $m$, as $n \to \infty$,

\[
\frac{1}{n} \sum_{j=m}^{m+n-1} E_m W_\theta(\theta^*, r_{1,j}, r_{2,j}, \eta_j) \to 0 \quad \text{in probability},
\]

\[
\frac{1}{n} \sum_{j=m}^{m+n-1} E_m P_\theta(\theta^*, \eta_j) \to \bar{P}_\theta(\theta^*) \quad \text{in probability},
\]

\[
\sum_{j=m}^{\infty} |E_m W_\theta(\theta^*, r_{1,j}, r_{2,j}, \eta_j)| < \infty,
\]

\[
\sum_{j=m}^{\infty} |E_m P_\theta(\theta^*, \eta_j) - \bar{P}_\theta(\theta^*)| < \infty,
\]

where $E_m$ denotes the conditional expectation on the $\sigma$-algebra $\mathcal{F}_m = \{\theta_0, r_{1j}, r_{2j}, \eta_j : j \leq m\}$.

(iii) The matrix $M + \bar{P}_\theta(\theta^*)$ is stable in that all of its eigenvalues are on the left half of the complex plane.

(iv) There is a twice continuously differentiable Lyapunov function $V(\cdot) : \mathbb{R}^{2r} \to \mathbb{R}$ such that

- $V(\theta) \to \infty$ as $|\theta| \to \infty$, and $V_\theta(\cdot)$ is uniformly bounded.
- $|V_\theta(\theta)| \leq K(1 + V^{1/2}(\theta))$.
- $|M \theta + \bar{P}(\theta)|^2 \leq K(1 + V(\theta))$ for each $\theta$.
- $V_\theta(\theta)(M \theta + \bar{P}(\theta)) \leq -\lambda V(\theta)$ for some $\lambda > 0$ and each $\theta \neq \theta^*$.
(A5) Denote
\[ \tilde{W}(\theta, r_1, r_2, \eta) = P(\theta, \eta) - \overline{P}(\theta) + W(\theta, r_1, r_2, \eta), \]
we have
\[ \sum_{j=m}^{\infty} |E_m \tilde{W}'(\theta_m, r_{1,m}, r_{2,m}, \eta_m) \tilde{W}(\theta_j, r_{1,j}, r_{2,j}, \eta_j)| < \infty, \]

(A6) The sequence
\[ B^\varepsilon(t) = \sqrt{\varepsilon} \sum_{j=0}^{t/\varepsilon - 1} \tilde{W}(\theta^*, r_{1,j}, r_{2,j}, \eta_j) \]
converges weakly to \( B(\cdot) \), a Brownian motion whose covariance \( \Sigma \) with \( \Sigma \in \mathbb{R}^{2r \times 2r} \) is given by
\[ \Sigma = E \tilde{W}(\theta^*, r_{1,0}, r_{2,0}, \eta_0) \tilde{W}'(\theta^*, r_{1,0}, r_{2,0}, \eta_0) + \sum_{k=1}^{\infty} E \tilde{W}(\theta^*, r_{1,0}, r_{2,0}, \eta_0) \tilde{W}'(\theta^*, r_{1,k}, r_{2,k}, \eta_k) + \sum_{k=1}^{\infty} E \tilde{W}(\theta^*, r_{1,k}, r_{2,k}, \eta_k) \tilde{W}'(\theta^*, r_{1,0}, r_{2,0}, \eta_0). \]

Remark 5 Note that (A4)(ii) is another noise condition. The motivation is similar to Remark 2. (A4)(iv) assumes the existence of a Liapunov function. Only the existence is needed; its precise form need not be known. For simplicity, we have assumed the convergence of the scaled sequence to a Brownian motion in (A6); sufficient conditions are well known; see for example, [34, Section 7.4]. Before proceeding further, we first obtain a moment bound of \( \theta_n \).

Lemma 6 Assume that (A1)-(A6) hold. Then there is an \( N_\varepsilon \) such that for all \( n > N_\varepsilon \),
\[ EV(\theta_n) = O(\varepsilon). \]

Proof. To begin, it can be seen that
\[ E_n V(\theta_{n+1}) - V(\theta_n) \]
\[ = \varepsilon V'_\theta(\theta_n)(M \theta_n + \overline{P}(\theta_n)) + E_n \varepsilon V'_\theta(\theta_n) \tilde{W}(\theta_n, r_{1,n}, r_{2,n}, \eta_n) \]
\[ + E_n \varepsilon^2 \frac{1}{2} (\theta_{n+1} - \theta_n) V''_\theta(\theta_n)(\theta_{n+1} - \theta_n) \]
\[ \leq \varepsilon V'_\theta(\theta_n)(M \theta_n + \overline{P}(\theta_n)) + \varepsilon E_n V'_\theta(\theta_n) \tilde{W}(\theta_n, r_{1,n}, r_{2,n}, \eta_n) \]
\[ + O(\varepsilon^2)(1 + V(\theta_n) + E_n |\tilde{W}(\theta_n, r_{1,n}, r_{2,n}, \eta_n)|^2) \]
\[ \leq -\varepsilon \lambda V(\theta_n) + \varepsilon E_n V'_\theta(\theta_n) \tilde{W}(\theta_n, r_{1,n}, r_{2,n}, \eta_n) + O(\varepsilon^2)(1 + V(\theta_n)), \]

15
where $\theta_n^+$ is on the line segment joining $\theta_n$ and $\theta_{n+1}$. The second inequality in (32) follows from the growth condition in (A4)(iv), the last inequality follows from (A1). To proceed, we use the methods of perturbed Lyapunov functions, which entitles to introduce small perturbations to a Lyapunov function in order to make desired cancelation. Define a perturbation

$$V_1^\varepsilon(\theta, n) = \varepsilon \sum_{j=n}^{\infty} E_n V_\theta'(\theta) \tilde{W}(\theta, r_{1,j}, r_{2,j}, \eta_j).$$

Note that

$$|V_1^\varepsilon(\theta, n)| = K\varepsilon(1 + V(\theta)).$$

Moreover,

$$E_n V_1^\varepsilon(\theta_{n+1}, n + 1) - V_1^\varepsilon(\theta, n) = E_n V_1^\varepsilon(\theta_{n+1}, n + 1) - E_n V_1^\varepsilon(\theta_n, n + 1) + E_n V_1^\varepsilon(\theta_n, n + 1) - V_1^\varepsilon(\theta, n)$$

$$= O(\varepsilon^2)(V(\theta_n) + 1) - \varepsilon E_n V_\theta'(\theta_n) \tilde{W}(\theta_n, r_{1,n}, r_{2,n}, \eta_n).$$

Define

$$V^\varepsilon(\theta, n) = V(\theta) + V_1^\varepsilon(\theta, n).$$

Using (32) and (34), we obtain

$$E_n V^\varepsilon(\theta_{n+1}, n + 1) \leq (1 - \varepsilon \lambda)V^\varepsilon(\theta_n, n) + O(\varepsilon^2)(1 + V^\varepsilon(\theta_n, n)),$$

Choosing $N_\varepsilon$ to be a positive integer such that

$$\left(1 - \frac{\lambda \varepsilon^2}{2}\right)^{N_\varepsilon} \leq K\varepsilon.$$

Iterating on the recursion (35), taking expectation, and using the order of magnitude estimate (33), we can then obtain

$$E V^\varepsilon(\theta_{n+1}, n + 1) \leq (1 - \varepsilon \lambda) E V^\varepsilon(\theta_n, n) + O(\varepsilon^2)(1 + V^\varepsilon(\theta_n, n))$$

$$\leq (1 - \frac{\varepsilon \lambda}{2})^n E V^\varepsilon(\theta_0, 0) + O(\varepsilon)$$

$$= O(\varepsilon).$$

when $n > N_\varepsilon$. The second line of (36) follows from $1 - \lambda \varepsilon + O(\varepsilon^2) \leq 1 - \frac{\lambda \varepsilon^2}{2}$ for sufficiently small $\varepsilon$. Now using (33) again, we also have $E V(\theta_{n+1}) = O(\varepsilon)$. Thus the desired estimate follows.

As in (29) define $z_n = (\theta_n - \theta^*)/\sqrt{\varepsilon}$. Then it is readily verified that

$$z_{n+1} = z_n + \varepsilon(M + \bar{P}_\theta(\theta^*))z_n + \sqrt{\varepsilon}(P(\theta^*, \eta_n) - \bar{P}(\theta^*) + W(\theta^*, r_{1,n}, r_{2,n}, \eta_n))$$

$$+ \varepsilon(P_\theta(\theta^*, \eta_n) - \bar{P}_\theta(\theta^*) + W_\theta(\theta^*, r_{1,n}, r_{2,n}, \eta_n))z_n + o(|z_n|^2).$$

16
Corollary 7 Assume that (A1)-(A6) hold. If the Lyapunov function is locally quadratic, i.e.,

\[ V(\theta) = (\theta - \theta^*)'Q(\theta - \theta^*) + o(|\theta - \theta^*|^2). \]

Then \( EV(z_n) = O(1) \) for all \( n > N_\varepsilon \).

Now we are in a position to study the asymptotic properties through weak convergence of appropriately interpolated sequence of \( z_n \). Define \( z^\varepsilon(t) = z_n \) for \( t \in [(n - N_\varepsilon)\varepsilon, (n - N_\varepsilon)\varepsilon + \varepsilon] \).

Now we can introduce the truncation sequence \( z^\varepsilon,N(\cdot) \) and truncation function \( q^N(\cdot) \) and use similar “piecing together” arguments as in Section 3. Nevertheless, for notational simplicity, we use \( z^\varepsilon(t) \) and suppose that it is bounded. For the rate of convergence, our focus is on the convergence of the sequence \( z^\varepsilon(\cdot) \). We shall show that it converges to a diffusion process whose covariance matrix together with the scaling factor will provide us with the desired convergence rates. Although more complex than Theorem 3, we still use the martingale problem setup. To keep the presentation relatively brief, we shall only outline the main steps needed.

For any \( t, s > 0 \),

\[
z^\varepsilon(t + s) - z^\varepsilon(t) = \varepsilon \sum_{j=t/\varepsilon}^{(t+s)/\varepsilon-1} (M + \overline{\theta}(\theta^*))z_j + \sqrt{\varepsilon} \sum_{j=t/\varepsilon}^{(t+s)/\varepsilon} \overline{W}(\theta^*, r_{1,j}, r_{2,j}, \eta_j)
+ \varepsilon \sum_{j=t/\varepsilon}^{(t+s)/\varepsilon} \overline{W}_\theta(\theta^*, r_{1,j}, r_{2,j}, \eta_j)z_j.
\] (38)

Note that for any \( \delta > 0, t, s > 0 \) with \( s < \delta \),

\[
E_t^\varepsilon \left| \sum_{j=t/\varepsilon}^{(t+s)/\varepsilon-1} \overline{W}(\theta^*, r_{1,j}, r_{2,j}, \eta_j) \right|^2
= \varepsilon \sum_{j=t/\varepsilon}^{(t+s)/\varepsilon-1} \sum_{k=t/\varepsilon}^{(t+s)/\varepsilon-1} \text{tr}[E_{t/\varepsilon}^{\varepsilon} \overline{W}(\theta^*, r_{1,j}, r_{2,j}, \eta_j)\overline{W}'(\theta^*, r_{1,k}, r_{2,k}, \eta_k)]
\leq K\varepsilon \left( \frac{t+s}{\varepsilon} - \frac{t}{\varepsilon} \right) = Ks \leq K\delta.
\]

and

\[
E_t^\varepsilon \left| \sum_{j=t/\varepsilon}^{(t+s)/\varepsilon-1} \overline{W}_\theta(\theta^*, r_{1,j}, r_{2,j}, \eta_j)z_j \right|^2
\leq \varepsilon^2 \sum_{j=t/\varepsilon}^{(t+s)/\varepsilon-1} |\overline{W}_\theta(\theta^*, r_{1,j}, r_{2,j}, \eta_j)|^2 \sum_{j=t/\varepsilon}^{(t+s)/\varepsilon-1} |z_j|^2
\leq K\varepsilon \left( \frac{t+s}{\varepsilon} - \frac{t}{\varepsilon} \right) = Ks \leq K\delta.
\]
Using Corollary 7 and similar argument as that of Theorem 3, we have the following result.

**Lemma 8** Assume conditions of Corollary 7, \( \{z^\varepsilon(\cdot)\} \) is tight on \( D([0, T] : \mathbb{R}^{2r}) \).

Next we can extract a convergent subsequence of \( \{z^\varepsilon(\cdot)\} \). Without loss of generality, still denote the subsequence by \( z^\varepsilon(\cdot) \) with limit \( z(\cdot) \). Using the For any \( t, s > 0 \), (38) holds. The way to derive the limit is similar to that of Theorem 3 using martigale problem formulation although the analysis is more involved. We proceed to show that the limit is the unique solution for the martingale problem with operator

\[
Lf(z) = \frac{1}{2} \text{tr}(\Sigma f_{zz}(z)) + (\nabla f(z))'(M + \mathcal{P}(\theta_*)),
\]

for \( f \in C^2_0, C^2 \) functions with compact support.

Using similar notation as that of Section 3. Redefine

\[
\tilde{h} = h(z(t_i) : i \leq \kappa), \quad \tilde{h}^\varepsilon = h(z^\varepsilon(t_i) : i \leq \kappa).
\]

By (A4)(ii), as \( \varepsilon \to 0 \)

\[
E\tilde{h}^\varepsilon \left[ \sum_{j=t/\varepsilon}^{(t+s)/\varepsilon} \tilde{W}(\theta^*, r_{1,j}, r_{2,j}, \eta_j) z_j \right] = E\tilde{h}^\varepsilon \left[ \sum_{l=t/\delta_\varepsilon}^{(t+s)/\delta_\varepsilon} \sum_{m_{\varepsilon}}^{l m_{\varepsilon} + m_{\varepsilon} - 1} \tilde{W}(\theta^*, r_{1,j}, r_{2,j}, \eta_j) z_j \right]
\]

\[
= E\tilde{h}^\varepsilon \left[ \sum_{l=t/\delta_\varepsilon}^{(t+s)/\delta_\varepsilon} \sum_{m_{\varepsilon}}^{l m_{\varepsilon} + m_{\varepsilon} - 1} \tilde{W}(\theta^*, r_{1,j}, r_{2,j}, \eta_j) z_{l m_{\varepsilon}} \right]
\]

\[
\quad + \sum_{l=t/\delta_\varepsilon}^{(t+s)/\delta_\varepsilon} \sum_{m_{\varepsilon}}^{l m_{\varepsilon} + m_{\varepsilon} - 1} \tilde{W}(\theta^*, r_{1,j}, r_{2,j}, \eta_j) [z_j - z_{l m_{\varepsilon}}]
\]

\[
\to 0.
\]

Using the notation as in Section 3,

\[
E\tilde{h}^\varepsilon \left[ \sum_{l=t/\delta_\varepsilon}^{(t+s)/\delta_\varepsilon} \sum_{m_{\varepsilon}}^{l m_{\varepsilon} + m_{\varepsilon} - 1} \tilde{W}(\theta^*, r_{1,j}, r_{2,j}, \eta_j) [z_j - z_{l m_{\varepsilon}}] \right]
\]

\[
\to 0 \text{ as } \varepsilon \to 0.
\]

Moreover, by (A6) we have

\[
\sqrt{\varepsilon} \sum_{j=t/\varepsilon}^{(t+s)/\varepsilon} \tilde{W}(\theta^*, r_{1,j}, r_{2,j}, \eta_j) \to \int_t^{t+s} dB(u)
\]

as \( \varepsilon \to 0 \). For the first term of (38), we have

\[
E\tilde{h}^\varepsilon \left[ \varepsilon \sum_{j=t/\varepsilon}^{(t+s)/\varepsilon} (M + \mathcal{P}_\theta(\theta^*)) z_j \right] \to E\tilde{h} \left[ \int_t^{t+s} (M + \mathcal{P}_\theta(\theta^*)) z(u) du \right]
\]

as \( \varepsilon \to 0 \). Putting the aforementioned arguments together, we have the following theorem.
Theorem 9 Under conditions (A1)-(A7), \( \{z^\varepsilon(\cdot)\} \) converges to \( z(\cdot) \) such that \( z(\cdot) \) is a solution of the following stochastic differential equation

\[
dz = [M + \bar{P}_\theta(\theta^*)]zdt + \Sigma^{1/2}d\hat{B}(t),
\]

where \( \hat{B}(\cdot) \) is a standard Brownian motion.

5 Numerical Simulation

This section contains two parts. First, we use two examples to verify the theoretical results of Sections 3 and 4. Second, we compare our PSO with traditional PSOs on several test functions for optimization.

5.1 Demonstration of Convergence

We use two simulation examples to demonstrate the convergence properties. It is mainly for demonstration purpose. Using (4), we take \( \varepsilon = 0.01, \chi = 1, \kappa_1 = -0.271, \kappa_2 = 1, c_1 = c_2 = 1.5 \). For simplicity, we take the additive noise \( \text{Pr}(\theta_n, \eta_n) = \text{Pr}(\theta_n) + \eta_n \) and \( \text{Pg}(\theta_n, \eta_n) = \text{Pg}(\theta_n) + \eta_n \), where \( \eta_n \) is a sequence of i.i.d. random variables with a standard normal distribution \( \mathcal{N}(0, 1) \). In addition, we set the number of swarms to be 5.

Example 10 Consider the sphere function:

\[
F_1(X) = \sum_{i=1}^{n} X_i^2,
\]

where \( n \) is the dimension of the variable \( x \). Its global optimum is \( (0, 0, \ldots, 0)' \). First, the dimension of \( X \) is set to be 1. Figures 1 shows the state trajectories (left) and the centered and scaled errors of the first component \( \theta^1_n \) (right).

Next, we consider the 2-dimension case of \( X \). Figures 2 illustrates the state trajectories (left) and the centered and scaled errors of the first component \( \theta^1_n \) (right).

Example 11 Consider the Rastrigin function [39]

\[
F_2(X) = 10n + \sum_{i=1}^{n} [X_i^2 - 10\cos(2\pi X_i)],
\]

where \( n \) is the dimension of the variable \( x \).
This function has many local minima. Its global optimum is given by \((0, 0, \ldots, 0)\)' . Same as Example 10, we set the dimension of \(X\) to be 1 and 2, respectively. The consensus error norm trajectories and the centered and scaled errors of the first component are demonstrated in Figures 3 and 4, respectively.

From these figures, we can conclude that all the swarms converge to a point in the searching space. These results were obtained without assuming that \(r_1, r_2, Pr, \text{ and } Pg\) are fixed. Our numerical results confirm our theoretical findings in Sections 3 and 4.
Table 1: Functions used for comparison

| Function                  | Formula                                                                 |
|---------------------------|--------------------------------------------------------------------------|
| Sphere function (De Jong’s f1) | $f_1(x) = \sum_{i=1}^{n} x_i^2$                                         |
| Rosenbrock variant (De Jong’s f2) | $f_2(x) = 100(x_1^2 - x_2)^2 + (1 - x_1)^2$               |
| De Jong’s f4 - no noise     | $f_4(x) = \sum_{i=1}^{n} i \cdot x_i^4$                                 |
| Foxholes (De Jong’s f5)    | $f_5(x) = (0.002 + \sum_{j=1}^{25} \frac{1}{1+0.001(x_j-0.5)^2})^{-1}$ |
| Shaffer’s f6               | $f_6(x) = 0.5 + \frac{(\sin \sqrt{x^2+y^2})^2 - 0.5}{(1.0+0.001(x^2+y^2))^2}$ |
| Griewank function          | $f_7(x) = 1 + \frac{1}{4000} \sum_{i=1}^{n} (x_i - 100)^2 - \prod_{i=1}^{n} \cos \left( \frac{x_i-100}{\sqrt{i}} \right)$ |
| Ackley’s function          | $f_8(x) = 20+e^{-20 \exp \left( -0.2 \left( \sqrt{\frac{1}{n} \sum_{i=1}^{n} x_i^2} \right) \right)} \exp \left( 0.2 \sum_{i=1}^{n} \cos (2 \pi x_i) \right)$ |
| Rosenbrock function        | $f_9(x) = \sum_{i=1}^{n} (100(x_{i+1} - x_i)^2 + (x_i - 1)^2)$           |
| Rastrigin function         | $f_{10}(x) = \sum_{i=1}^{n} x_i^2 - 10 \cos(2\pi x_i) + 10$             |

5.2 Comparisons of Our PSO and Traditional PSOs

We have done extensive comparisons by comparing the PSO algorithm proposed here with several types of PSOs in a set of unconstrained real-valued benchmark functions. We chose the functions same as in [26], namely, De Jong’s functions [40], Schaffer’s f6 [41], Griewank [42], Rosenbrock [43], Ackley [44], and Rastrigin functions; see Tables 1 and 2. These functions were implemented in 30 dimensions except for f2, f5, and f6, which are given two dimension functions. In all cases except f5, the optimum is 0. For f5, the best known result is 0.998004 when $x = (-32, -32)'$.

As in [26], a population of 20 particles was run for 20 trials per function, with the best performance evaluation recorded after 2000 iterations. Use (4), choose parameters $\varepsilon = 0.01$, $\chi = 1$, $\kappa_1 = -0.6$, $\kappa_2 = 1$, $c_1 = c_2 = 100$, and use noise-free observation $\Pr(\theta_n, \eta_n) = \Pr(\theta_n)$ and $\Pr(\theta_n, \eta_n) = \Pr(\theta_n)$.

Table 3 compares this PSO’s performance to that of the traditional PSOs. All particle swarm populations comprised 20 individuals. Results of columns 2 to 7 are from [26].
| Function         | Dimension | Initial Range |
|-----------------|-----------|---------------|
| 1               | 30        | ±20           |
| 2               | 2         | ±50           |
| 4               | 30        | ±20           |
| 5               | 2         | ±50           |
| Schaffer’s f6   | 2         | ±100          |
| Griewank        | 30        | ±300          |
| Ackley          | 30        | ±32           |
| Rastrigin       | 30        | ±5.12         |
| Rosenbrock      | 30        | ±10           |

Table 2: Function parameters for the test problems

| Function         | $V_{\text{max}} = 2$ | $V_{\text{max}} = 4$ | Type 1' | Type 1 | Exp. Version | E&S | Our PSO (Noise-free) | Entries of columns 2 and 3 comes from using (1). Here $V_{\text{max}}$ means using “velocity clamp” (i.e., if $v^i$ is greater than $V_{\text{max}}$ in iteration, then let $v^i = V_{\text{max}}$). Type 1’, Type 1, and Experimental version (columns 3 to 5) mean using variations of (3) suggested in [26]. E&S (column 7) means using (2) suggested in [25]. The superiority of using our PSO algorithms is clearly pronounced.

Next, we consider the PSO algorithms when noisy observations are taken, a more realistic situation. Keeping other parameters unchanged, we let $\Pr(\theta_n, \eta_n) = \Pr(\theta_n)(1 + \eta_n)$ and $Pg(\theta_n, \eta_n) = Pg(\theta_n)(1+\eta_n)$, where $\eta_n$ is a sequence of i.i.d. standard normal random following the distribution $\mathcal{N}(0,1)$. Table 4 illustrates the results. No comparisons w.r.t. previous results were made since all of the aforementioned references consider only no observation noise case. The computational result shows that the dimension (of the parameter) is low the PSO algorithms still perform fairly well; see $f_2$, $f_5$, and $f_6$, where the dimensions are 2. When we treat high dimensional problems, some of the computational results (e.g., the
| Function       | Noisy Obs. (multiplicative noise) |
|----------------|-----------------------------------|
| 1              | 7.7512e-14                        |
| 2              | 0                                 |
| 4              | 0                                 |
| 5              | 0.998004                          |
| Shaffer’s f6   | 0                                 |
| Griewank       | 1.6378e-12                        |
| Ackley         | 4.9776e-7                         |
| Rastrigin      | 4.0927e-12                        |
| Rosenbrock     | 28.5044                           |

Table 4: Empirical results: the PSO with additional noise

Rosenbrock function) are not as good as the lower dimensional problems.

6 Further Remarks

In this paper, we developed a class of general PSO algorithms using a stochastic approximation setup. Different from the existing results in the literature, we have used more general assumptions and obtained more general convergence results without depending on empirical working. In addition, we obtained rates of convergence for the PSO algorithms for the first time. Numerical simulation also shows that our PSO algorithms performs better than that of the traditional ones in a number of test functions for optimization.

Several research directions may be pursued in the future. How to systematically choose the parameter values $\kappa_1$, $\kappa_2$, $c_1$ and $c_2$ is an interesting and practically challenging problem. One thought is to construct a level two (stochastic) optimization algorithm to select best parameter value in a suitable sense. To proceed in this direction requires careful thoughts and consideration. In addition, we can consider that some parameters such as $\chi$, $\kappa_1$, etc. are not fixed but change randomly during iterations or change owing to some random environment change (for example, see [45]). The problem to study is to analyze the convergence and convergence rates in such a case.

To conclude, this paper demonstrated convergence properties of a class of general PSO algorithms and derived the rates of convergence by using a centered and scaled sequence of its iterates. This study opens new arenas for subsequent studies on determining convergence capabilities of different PSO algorithms and parameters.
References

[1] J. Kennedy, R.C. Eberhart, Particle swarm optimization, in: Proc. IEEE Conf. on Neural Networks, IV, Piscataway, NJ, 1995, pp. 1942-1948.

[2] C.F. Juang, A hybrid of genetic algorithm and particle swarm optimization for recurrent network design, IEEE Trans. Syst., Man, Cybern. B, Cybern., vol. 34, no. 2, pp. 997-1006, Apr. 2004.

[3] L. Messerschmidt and A.P. Engelbrecht, Learning to play games using a PSO-based competitive learning approach, IEEE Trans. Evol. Comput., vol. 8, no. 3, pp. 280-288, Jun. 2004.

[4] Jr. E.F. Costa, P.L.C Lage, and Jr. E.C. Biscaia. On the numerical solution and optimization of styrene polymerization in tubular reactors. Comput. Chem. Eng., 27, pp.1591-1604, 2003.

[5] M. R. AlRashidi, M. E. El-Hawary, A Survey of Particle Swarm Optimization Applications in Electric Power Systems, IEEE Trans. Evolutionary Comp., VOL. 13, NO. 4, AUGUST 2009, pp. 913-918

[6] M.A. Abido, Particle swarm optimization for multimachine power system stabilizer design, in Proc. Power Eng. Soc. Summer Meeting, 2001, pp. 1346-1351.

[7] G. Kovács, A. Groenwold, and K. Jármai, et al., Analysis and optimum design of fibereinforced composite structures, Struct. Multidiscip. Optim., 2004, 28: 170-179.

[8] X. Zhang, L. Yu, and Y. Zheng, et al. Two-stage adaptive PMD compensation in a 10 Gbit/s optical communication system using particle swarm optimization. Opt. Commun., 2004, 231: 233-242.

[9] Y. Li and X. Chen, Mobile robot navigation using particle swarm optimization and adaptive NN, in Proc. 1st Int. Conf. Nat. Comput., Changsha, China, Lecture Notes in Computer Science, vol. 3612. Berlin, Germany: Springer-Verlag, 2005, pp. 554-559.

[10] H. Wu, F. Sun, and Z. Sun, et al. Optimal trajectory planning of a flexible dual-arm space robot with vibration reduction. J. Intell. Robot. Syst., 2004, 40: 147-163

[11] N.G. Pavlidis, K.E. Parsopoulos, and M.N. Vrahatis, Computing Nash equilibria through computational intelligence methods. J. Comput. Appl. Math., 2005, 175: 113-136.

[12] J. Nenoraite, R. Simutis, Stocks’ trading system based on the particle swarm optimization algorithm, International Conference on Computational Science, 2004: 843-850.

[13] K.E. Parsopoulos, E.I. Papageorigiou, and P.P. Groumpos, et al. Evolutionary computation techniques for optimizing fuzzy cognitive maps in radiation therapy systems. In: Proc. of the GECCO. 2004: 402-413.

[14] T.K. Rasmussen, T. Krink, Improved hidden Markov model training for multiple sequence alignment by a particle swarm optimization-evolutionary algorithm hybrid. Biosystems, 2003, 72: 5-17.

[15] X. Xiao, E.R. Dow, R. Eberhart, et al., Hybrid self-organizing maps and particle swarm optimization approach. Concurr. Comp-Pract. E., 2004, 16: 895-915.

[16] Q. Shen, J. Jiang, and C. Jiao, et al., Modified particle swarm optimization algorithm for variable selection in MLR and PLS modeling: QSAR studies of antagonism of angiotensin II: antagonists. Eur. J. Pharm. Sci., 2004, 22: 145-152.

[17] M.F. Tasgetiren, M. Sevkli, and Y. Liang, et al., Partial swarm optimization algorithm for permutation flowshop sequencing problem. Lecture Notes in Computer Sciences, 2004, 3172: 382-389.

[18] B. Liu, L. Wang, Y. Jin, An effective PSO-based memetic algorithm for flowshop scheduling. IEEE Trans. Syst. Man. Cy. B.-Cybernetics, 2007, 37(1): 18-27.

[19] Y. Liu, Z. Qin, and Z. Shi, Hybrid particle swarm optimizer with line search, in Proc. IEEE Int. Conf. Syst., Man, Cybern., 2004, vol. 4, pp. 3751-3755.
[20] Y. Liu and K.M. Passino, Stable social foraging swarms in a noisy environment, *IEEE Trans. Automatic Control*, 49 (2004) No. 1, 30-44.

[21] A. Ratnaweera, S.K. Halgamuge, and H. C. Watson, Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients, *IEEE Trans. Evol. Comput.*, vol. 8, no. 3, pp. 240-255, Jun. 2004.

[22] R. Poli, C.D. Chio, and W.B. Langdon, Exploring extended particle swarms: A genetic programming approach, in Proc. Conf. Genet. and Evol. Comput., Washington, DC, 2005, pp. 169-176.

[23] K.E. Parsopoulos and M.N. Vrahatis, Recent approaches to global optimization problems through particle swarm optimization, *Nat. Comput.*, vol. 1, no. 2/3, pp. 235-306, Jun. 2002.

[24] W.J. Zhang and X.F. Xie, DEPSO: Hybrid particle swarm with differential evolution operator, in Proc. *IEEE Int. Conf. Syst., Man, Cybern.*, Washington, DC, 2003, pp. 3816-3821.

[25] Y. Shi and R. Eberhart, A modified particle swarm optimizer, in Proc. IEEE World Congr. Comput. Intell., May 1998, pp. 69-73.

[26] M. Clerc and J. Kennedy, The particle swarm: explosion, stability, and convergence in a multidimensional complex space. *IEEE Trans. Evolut. Comput.*, 2002, 6: 58-73.

[27] I.C. Trelea, The Particle Swarm Optimization Algorithm: convergence analysis and parameter selection. Information Processing Letters 2003, 85 (6): 317-325.

[28] K. Yasuda, A. Ide, and N. Iwasaki, Adaptive particle swarm optimization, in Proc. *IEEE Int. Conf. Syst., Man, Cybern.*, 2003, pp. 1554-1559.

[29] B. Brandstäder and U. Baumgartner, Particle swarm optimization Mass-spring system analogon, *IEEE Trans. Magn.*, vol. 38, no. 2, pp. 997-1000, Mar. 2002.

[30] H.M. Emara and H.A. Fattah, Continuous swarm optimization technique with stability analysis, in Proc. Amer. Control Conf., 2004, vol. 3, pp. 2811-2817.

[31] J.L. Fernández-Martínez, E. García-Gonzalo, Stochastic Stability Analysis of the Linear Continuous and Discrete PSO Models. *IEEE Trans. Evolutionary Computation*, 15, NO. 3, JUNE 2011.

[32] M.E.H. Pedersen and A.J. Chipperfield, Simplifying Particle Swarm Optimization, Appl. Soft Computing, 2010, 2: 618-628.

[33] X. Chen and Y. Li, A modified PSO structure resulting in high exploration ability with convergence guaranteed. *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics*, 2007, 5: 1271-1289.

[34] H.J. Kushner and G. Yin, Stochastic Approximation and Recursive Algorithms and Applications, 2nd Ed., Springer-Verlag, New York, NY, 2003.

[35] H.J. Kushner, Approximation and Weak Convergence Methods for Random Processes, with Applications to Stochastic Systems Theory, MIT Press, Cambridge, MA, 1984.

[36] Y. Tian and Y. Takane, The inverse of any two-by-two nonsingular partitioned matrix and three matrix inverse completion problems, *Comp. & Math. Appl.*, 57(8), April 2009, 1294-1304.

[37] P. L’Ecuyer and G. Yin, Budget-dependent convergence rate of stochastic approximation, *SIAM J. Optim.*, 8 (1998), 217-247.

[38] G. Yin, Rates of convergence for a class of global stochastic optimization algorithms, *SIAM J. Optim.*, 10 (1999), 99-120.

[39] R. G. Reynolds and C.-J. Chung, Knowledge-based self-adaptation in evolutionary programming using cultural algorithms, in Proc. *IEEE Int. Conf. Evolutionary Computation*, Indianapolis, IN, Apr. 1997, pp. 71-76.
[40] K. De Jong, An analysis of the behavior of a class of genetic adaptive systems, Ph.D. dissertation, Dept. Comput. Sci., Univ. Michigan, Ann Arbor, MI, 1975.

[41] L. Davis, Ed., Handbook of Genetic Algorithms. New York: Van Nostrand Reinhold, 1991.

[42] A. O. Griewank, Generalized Decent for Global Optimization., J. Opt. Th. Appl. 34, 11-39, 1981.

[43] P. Angeline, Evolutionary optimization versus particle swarm optimization: Philosophy and performance differences, in Evolutionary Programming VII, V. W. Porto, N. Saravanan, D. Waagen, and A. E. Eiben, Eds. Berlin, Germany: Springer-Verlag, 1998, pp. 601C610.

[44] D. H. Ackley. A connectionist machine for genetic hillclimbing. Boston: Kluwer Academic Publishers, 1987.

[45] G. Yin and C. Zhu, Hybrid Switching Diffusions: Properties and Applications, Springer, New York, 2010.