Discrete Time Extremum Seeking using Stochastic Perturbations

Chris Manzie and Miroslav Krstic

Abstract—Extremum seeking using periodic, deterministic perturbations has been an effective method for non-model based real time optimisation when only limited knowledge of the system is available. However, periodicity can naturally lead to predictability which is undesirable in some applications and unrepresentative of some biological optimisation processes such as bacterial chemotaxis. With this in mind, it is useful to investigate the benefit of employing stochastic perturbations in the context of a typical extremum seeking architecture, and to compare the approach with existing stochastic optimisation techniques. In this work, we show that the convergence towards the extremum of a static map can be guaranteed with the stochastic extremum seeking algorithm, and quantify the behaviour of the system at the extremum in terms of the extremum seeking constants and map parameters. Finally simulation results are used to demonstrate the stochastic closed loop system convergence and behaviour about the extremum. For the sake of analogy with the classical methods of stochastic approximation, stochastic extremum seeking in this paper is pursued in discrete time.

I. INTRODUCTION

Although sinusoid-based extremum seeking (ES) was initially used in the 1950s, it was not until the proof of its local stability results in 2000 [17] that interest in the technique was rekindled within the control community. A range of further developments have followed including different classes of extremum seeking schemes, [1], extensions to discrete time [7] and semi-global stability results [16]. As a consequence of these theoretical developments, numerous applications including engine calibration [11], drag reduction [2], and plasma control in tokamaks [5], [10] have been recently proposed.

Extremum seeking is a non-model based, real-time optimisation algorithm that uses zero-mean perturbations to develop a gradient estimate. The gradient estimate is then used to shift the state variable closer towards the extremum. A common thread in the published literature on ES to date is its reliance on periodic perturbations. The predictability associated with periodic perturbations can be disadvantageous for some tracking applications (where a vehicle navigated by a periodic extremum seeking algorithm can become a predictable target), while biological systems (such as bacterial chemotaxis) rarely utilize periodicity in locating optima. Furthermore, if the system has high dimensionality, the orthogonality requirements on the elements of the periodic perturbation vector pose an implementation challenge. Thus there is merit in investigating the use of stochastic perturbations within the ES architecture.

The simplest form of sinusoidal ES architecture was recognised by Tan et al in [16] and denoted the first order extremum seeking scheme. This structure differed from the earlier structures used in [17] by the removal of the washout filter prior to the state update, and is illustrated in discrete time form in Fig 1 for a minimisation problem with \( \beta > 0 \) (changing the sign of \( \beta \) leads to a maximum seeking algorithm). This first order scheme will be used as the base ES architecture in the discussions of this paper, although the inclusion of a washout filter on the output is known to significantly improve the transient performance when actuators are present in the closed loop system.

Fig. 1. First order ES system where \( x_k \) and \( v_k \) are the state of the system and perturbation at time \( k \), and \( y(\cdot) \) is a static map.

Of course, the concept of using stochastic perturbations in optimisation is not new and a rich and mature field of stochastic approximation already exists. Within this field there are some basic commonalities, namely the optimisation is implicitly or explicitly derived from the classical gradient descent law whereby updates to the state variable, \( x_k \), of the form \( x_{k+1} = x_k - a_k \hat{g}(x_k) \), are used to find the state, \( x^* \) that minimizes the input-output mapping \( y(x_k) \). However, since direct measurements of the gradient are usually unavailable for applications using most real systems, and the optimization algorithms take the form of approximated gradient descent:

\[
x_{k+1} = x_k - a_k \hat{g}_k(x_k)
\]

where \( \hat{g}_k(x_k) \) is the gradient estimate determined by the specific algorithm.

The existing stochastic approximation methods for estimating the gradient are often grouped into three basic classes - Finite Difference Stochastic Approximation (FDSA) [8], [4]; Random Direction Stochastic Approximation (RDSA) [9], [6] and Simultaneous Perturbation Stochastic Approximation (SPSA) [12], [14], [15], [6].

In the FDSA algorithm, each element of the state vector is perturbed one at a time, and each element of the gradient vector is estimated according to \( \hat{g}_{i,k}(x_k) = \frac{y(x_k + v_k) - y(x_k - v_k)}{2v_k} \). While giving the best estimate of the gradient direction at the current point, this algorithm requires \( 2N \) perturbations at every state update in an \( N \) dimensional system.

In the RDSA algorithm, all the elements of the state vector experience a perturbation at the same time but the effect on
the output is averaged across all the elements of the gradient vector, i.e.
\[ \hat{g}_k(x_k) = \Delta_k \frac{y(x_k + c_k \Delta_k) - y(x_k - c_k \Delta_k)}{2c_k} \] (2)
where the elements of the perturbation vector \( \Delta_k = [\Delta_{1,k} \cdots \Delta_{n,k}]^T \) may have separate distributions. This approach guarantees a correct gradient traversal of the static surface being optimised, and requires only two measurements per update.

Similarly, in the SPSA algorithm only two perturbations and measurements are used per update, however in this case the magnitude of the perturbation is used to normalize the gradient estimation in the direction of the perturbation, i.e.
\[ \hat{g}_{i,k}(x_k) = \frac{y(x_k + c_k \Delta_k) - y(x_k - c_k \Delta_k)}{2c_k \Delta_k} \] (3)
In RDSA and SPSA the random variables must satisfy certain conditions, although these are not identical in general and typically more limiting for the SPSA algorithm - for example, while RDSA can tolerate a perturbation containing \( \Delta_{i,k} = 0 \) whereas this is clearly not allowed in SPSA. In spite of these limitations, Spall demonstrated in [13] that the efficiency of SPSA was superior to FDSA, while Chin [6] argued that the relative efficiency of SPSA with a Bernoulli \([-1,1]\) distribution was superior to RDSA with a N(0,1) distribution of perturbations for certain static maps.

The stochastic optimisation algorithms as listed above are all two measurement algorithms, i.e. a perturbation on either side of the current state is used to estimate the gradient. In [13], a one-measurement version of SPSA was investigated and shown to be capable of locating the optimum of a static map, albeit typically not as quickly or as smoothly as the two measurement version of the algorithm. The one-measurement SPSA algorithm represents the simplest form of stochastic approximation algorithm we are aware of, and represents a more suitable implementation for a dynamic system where the state may not remain constant between two successive perturbations. It is not surprising then that this shares some commonalities with the extremum seeking scheme of Fig 1. It is important however to distinguish the two algorithms, and this is the goal of the following section.

II. COMPARISON OF SA AND ES FOR STATIC MAPS

From Fig. 1, the closed loop system representing the first order ES structure is given by the following vector difference equation
\[ x_{k+1} = x_k - \beta v_k y(x_k + v_k) \] (4)
The following assumption is placed on the distribution of the perturbations in the discrete ES scheme:

Assumption 2.1: The elements of the perturbation vector, \( v_k \), are stochastic and independent such that \( E(v_{i,k}) = 0 \), \( E(v_{i,k}^2) = \sigma_i^2 \) and \( E(v_{i,k} v_{j,k}) = 0 \) \( \forall i \neq j \). Further, it is also assumed that the p.d.f. of the perturbation vector is symmetric about its mean.

In comparison to (4), the one-measurement SPSA algorithm presented in [14] uses a single output measurement following a perturbation \( v_k = c_k \Delta_k \) to estimate the gradient, i.e.
\[ x_{k+1} = x_k - \frac{a_k y(x_k + c_k \Delta_k)}{c_k} \left[ \Delta_{1,k}^{-1} \cdots \Delta_{n,k}^{-1} \right]^T \] (5)
It is important to note that \( a_k \) and \( c_k \) are positive scalars and expected to approach zero asymptotically (the full conditions on these sequences are listed in [14]).

The primary difference in the structures of the one-measurement SPSA and the ES algorithms is the position of the perturbation on the right hand side of equations (4) and (5). The one-measurement SPSA algorithm is (naturally) a simplification of the two-measurement algorithm described in Section I, where explicit gradient approximation is attempted and thus the perturbation appears on the denominator of the state update. On the other hand, the ES approach has its background in deterministic averaging, and by implicit assumption there will be state updates in all directions, however their net effect should be towards the extremum, thus only the sign of the perturbation is important as a multiplier in (4). Thus, equations (4) and (5) are not equivalent except, just formally, in two cases: 1) when the state vector is a scalar and \( a_k = \beta (c_k \Delta_k)^2 \) (which is not a choice that would normally be made in the stochastic approximation algorithms because \( a_k \) should be deterministic and go to zero as \( k \) goes to infinity); and 2) in the vector case when each element of \( v_k \) is an unbiased Bernoulli-type variable from the set \([-1,1]\).

On the other hand, the RDSA algorithm described by (1) and (2) has a lot in common with how the ES algorithm utilises the stochastic perturbation, to the point whereby the simple ES structure of Fig 1 could be considered a one-measurement version of the RDSA algorithm. However, we do not consider classifying the stochastic ES as simply a one-measurement RDSA algorithm due to the common presence of a washout filter in the feedback loop when implementing ES. This type of modification has not been used with a stochastic RDSA algorithm previously.

A further important difference between the SA and ES algorithms is the assumptions in (5) that enforce \( c_k \to 0 \). Spall shows the bias in the gradient estimate is \( O(c_k^2) \) in Lemma 1 of [14], thus bias in the gradient estimate can be totally eliminated by this assumption. As inferred from [15], this may be an acceptable assumption for large dimensional static problems, however it is too constraining for use in online situations where the plant is subject to state disturbances, set point changes or constant drift in the optimum and hence the stochastic ES approach discussed in this work will utilize a constant step size, \( \beta \). A consequence of the time decreasing nature of step size in the one-measurement SPSA is that Proposition 1 of [14] cannot be used to show convergence of \( x_k \) to the extremum \( x^* \) for the stochastic ES algorithm.

III. CONVERGENCE OF FIRST ORDER STOCHASTIC ES

The convergence of the first order stochastic ES algorithm illustrated in Fig. 1 is addressed by firstly finding the
conditions for convergence of the averaged system, before showing that the difference between the averaged and true systems are bounded almost surely in the limit \( k \to \infty \). We begin by identifying the averaged discrete stochastic ES system under the following assumption.

**Assumption 3.1:** The output mapping is quadratic (at least locally), i.e.

\[
y(x) = (x - x^*)^T A(x - x^*) + y_{\text{min}}
\]

where \( x^*, y_{\text{min}} \) are unknown and \( A = \text{diag}(A_1, \ldots, A_n) \in \mathbb{R}^{n \times n} \). Without loss of generality we let \( x^* = 0 \) as a simple change of variable can be used to achieve this, thereby absorbing all uncertainty into the initial condition of the integrator, \( x_0 \).

**A. The averaged system**

The averaged system is obtained by taking the expectation of the closed loop system (4) and substituting (6),

\[
\mathbb{E}(x_{k+1}) = \mathbb{E}(x_k) - \beta \mathbb{E}(v_k x_k^T A x_k) - \beta y_{\text{min}}, \mathbb{E}(v_k) - \beta \mathbb{E}(2v_k x_k^T A x_k) - \beta \mathbb{E}(v_k v_k^T A v_k)
\]

Since \( v_k \) and \( x_k \) are independent, from Assumption 2.1 it follows that \( \mathbb{E}(v_k x_k^T A x_k) = 0; \mathbb{E}(v_k v_k^T A v_k) = 0; \) and \( \mathbb{E}(v_k v_k^T A x_k) = \Sigma_d \mathbb{E}(x_k) \) where \( \Sigma_d = A \times \text{diag} \left[ \sigma_1^2, \ldots, \sigma_n^2 \right] \). Thus (7) reduces to

\[
\mathbb{E}(x_{k+1}) = (I - 2\beta \Sigma_d) \mathbb{E}(x_k)
\]

Hence since \( \beta \) and \( \sigma_i^2 \) are all positive, the state \( \mathbb{E}(x_k) \) of the averaged system is exponentially convergent to the origin if \( \beta < 0.5/ \max_i (A_i \sigma_i^2) \).

**Theorem 1:** Given an initial condition \( x_0 \), \( \beta < 0.5/ \max_i (A_i \sigma_i^2) \) can be chosen sufficiently small that for all \( \gamma > 0 \) there exists a constant \( C(\beta) \) such that the difference between the solution of the true system and the equilibrium of the averaged system obeys

\[
\limsup_{k \to \infty} \mathbb{P}(\|x_k\| > \gamma) \leq C(\beta)
\]

where \( C(\beta) \to 0 \) as \( \beta \to 0 \).

**Proof:** The result follows directly from (6) and Theorem 3, p44, in [3].

**IV. Prediction of Behaviour at Steady State for Gaussian Perturbations**

Since the step size, \( \beta \), used in the ES algorithm is non-decreasing, unlike in the SPSA approach, there will be variations about the extremum of the mapping as \( k \to \infty \). To investigate this behaviour of the system, we consider the special case where \( v_{i,k} \) is a Gaussian sequence satisfying Assumption 2.1. Gaussian perturbations are not chosen to represent optimal perturbations for all mappings, but to instead demonstrate quantitative results; and naturally, similar analysis may be conducted for other perturbation distributions. With this in mind, the following properties are readily derived:

\[
\mathbb{E}(v_{i,k}^{2n+1}) = 0
\]

\[
\mathbb{E}(v_{i,k}^4) = \frac{1}{\sqrt{2\pi} \sigma_i^2} \int_{-\infty}^{\infty} x^4 e^{-x^2/2} \, dx = 3\sigma_i^4
\]

\[
\mathbb{E}(v_{i,k}^6) = \frac{1}{\sqrt{2\pi} \sigma_i^2} \int_{-\infty}^{\infty} x^6 e^{-x^2/2} \, dx = 15\sigma_i^6
\]

**A. Scalar case**

We begin with the state update equation (4), which for a scalar system is given by:

\[
x_{k+1} = x_k - \beta f_k
\]

where \( f_k = v_k y_k = A(v_k^2 + 2x_k v_k^2 + x_k v_k) + v_k y_{\text{min}} = f_{0,k} + v_k y_{\text{min}}\)

Note that \( f_{0,k} \) is the value of \( f_k \) if \( y_{\text{min}} = 0 \), and is defined only to simplify some of the notational detail. We now make the following assumption:

**Assumption 4.1:** We assume the inequality \( \mathbb{E}(x_k^{2n+1}) \geq \mathbb{E}(x_k^n) \) is true for all time \( k \) and \( n > 2 \).

**Remark 1:** This assumption is clearly restrictive and unverifiable, though it is not unreasonable when the initial condition \( x_0 \) and \( \beta \) are small.

We now define the following quantities and make a further assumption before moving to the main results of this section:

\[
\eta(\beta, \sigma, y_{\text{min}}) = 4A^2 \beta^2 - 18A^2 \beta^2 \sigma^4 - 2Ay_{\text{min}}^2 \beta^2 \sigma^2
\]

\[
\kappa(\beta, \sigma) = A^2 \beta^2 \sigma^2
\]

\[
\delta(\beta, \sigma, y_{\text{min}}) = 15A^2 \beta^2 \sigma^6 + 6A^2 \beta^2 y_{\text{min}}^4 + \beta^2 y_{\text{min}}^2 \sigma^2
\]

\[
\lambda(\beta, \sigma, y_{\text{min}}) = 90A^2 \beta^2 \sigma^6 + 36A^2 y_{\text{min}}^2 \sigma^4 + 6y_{\text{min}}^2 \beta^2 \sigma^2
\]

\[
\gamma(\beta, \sigma, y_{\text{min}}) = 8A\beta^2 - 108A^2 \beta^2 \sigma^4 - 12y_{\text{min}}^2 \beta^2 \sigma^2
\]

**Assumption 4.2:** The eigenvalues of the matrix

\[
\begin{bmatrix}
1 - \eta & \kappa \\
\lambda & 1 - \gamma
\end{bmatrix}
\]

are inside the unit disc.

**Theorem 2:** Given Assumptions 3.1, 4.1 and 4.2, with Gaussian perturbations satisfying Assumption 2.1 the variance in the state variable, \( \mathbb{E}(x_k^2) \) approaches the following limit as \( k \to \infty \):

\[
\mathbb{E}(x_k^2) \to \frac{\delta \gamma}{\eta \gamma - \kappa \lambda} + O(\beta^2)
\]

**Remark 2:** For small \( \beta \), we note that \( \delta \gamma = O(\beta^3), \kappa \lambda = O(\beta^3) \) and \( \eta \gamma = O(\beta^3) \). Hence it follows from (19) that \( \mathbb{E}(x_k^2) = O(\beta) \to 0 \) as \( \beta \to 0 \).

**Proof:** [Proof of Theorem 2] By squaring the updated variable given by (13) and taking the expectation it follows:

\[
\mathbb{E}(x_{k+1}^2) = \mathbb{E}(x_k^2 - 2\beta f_{0,k} x_k + \beta^2 f_{0,k}^2)
\]

\[
+ 2\mathbb{E}(\beta^2 y_{\text{min}} f_{0,k} v_k)
\]

\[
+ \mathbb{E}(\beta^2 y_{\text{min}} v_k^2 x_k) - 2\mathbb{E}(\beta y_{\text{min}} v_k x_k)
\]

The first expectation term on the RHS of (20) is the variance for \( y_{\text{min}} = 0 \), and is given by:

\[
\mathbb{E}(x_k^2 - 2\beta f_{0,k} x_k + \beta^2 f_{0,k}^2)
\]

\[
= (1 - 4A^2 \beta^2 + 18A^2 \beta^2 \sigma^4)\mathbb{E}(x_k^2)
\]

\[
+ A^2 \beta^2 \sigma^2 \mathbb{E}(v_k^2) + 15A^2 \beta^2 \sigma^6
\]
The remaining terms in (20) are:
\[2E(\beta^2 y_{min} f_k v_k) = 2A\beta^2 y_{min}(3\sigma^4 + \sigma^2 E(x_k^2)) \] (22)
\[E(\beta^2 y_{min}^2 v_k^2) = \beta^2 y_{min}^2 \sigma^2 \] (23)
\[2E(\beta y_{min} v_k x_k) = 0 \] (24)

Consequently, after substituting (14)-(16) into (21)-(24) and combining into (20), the variance in \( x_k \) is given by:
\[E(x_k^2) = (1 - \eta)E(x_k^2) + nE(x_k^4) + \delta \] (25)

A similar approach to deriving \( E(x_k^2) \) can be used to find the sequence \( E(x_k^4) \), but due to space constraints the details are omitted here. Applying Assumption 4.1 it follows:
\[E(x_k^4) = (\lambda + O(\beta^3))(E(x_k^2) + O(\beta^4)) \] (26)

where \( \lambda(\beta, \sigma, y_{min}) \) and \( \gamma(\beta, \sigma, y_{min}) \) are defined in (17)-(18).

Neglecting the higher order terms in (26), in equations (25)-(26) we then have a (weakly) coupled set of equations which can be represented by the system:
\[
\begin{bmatrix}
E(x_{k+1}^2) \\
\end{bmatrix} = 
\begin{bmatrix}
1 - \eta & \alpha_2 & \ldots & \alpha_N \\
\lambda + O(\beta^3) & 1 - \gamma + O(\beta^3) & 0 & 0 \\
\delta & \delta & \ldots & \delta \\
O(\beta^4) & O(\beta^4) & \ldots & O(\beta^4)
\end{bmatrix}
\begin{bmatrix}
E(x_k^2) \\
\end{bmatrix}
\]

As \( k \to \infty \), given Assumption 4.2 the \( E(x_k^2) \) sequence will approach the limit (19).

We now consider the bias and variance at the output of the static map for large \( k \).

**Theorem 3:** Given the system satisfies Assumptions 3.1, 4.1 and 4.2 and uses Gaussian perturbations satisfying Assumption 2.1, the mean and variance of the steady state output approach the following limits as \( k \to \infty \):
\[E(y - y_{min}) \to A \left( \frac{\delta_1}{\eta_1} + \sigma^2 \right) \] (28)
\[A \sigma^2 \text{ as } \beta \to 0^+ \] (29)
\[\text{var}(y_k - y_{min}) \to A^2 \left( \frac{\lambda + 6\sigma^2\gamma}{\eta_1} + 3\sigma^4 \right) \] (30)
\[\to 2A^2\sigma^2 \text{ as } \beta \to 0^+ \] (31)

**Proof:** The proof follows directly from Theorem 2.

**B. Vector case**

We now derive the asymptotic value of the covariance matrix of \( x_k \) for the non-scalar case. We begin by adjusting Assumption 4.1 as follows:

**Assumption 4.3:** \( E(x_{i,k}^n) \geq E(x_{i,k}^2) \) and \( E(x_{i,k}^n) \geq E(x_{i,k}^2) \) \( \forall n > 0, \forall k \)

**Remark 3:** Assumption 4.3 is more restrictive on \( \beta \) than the equivalent assumption for the scalar case (Assumption 4.1). This is not strictly necessary but will be used to provide a more concise formulation of the results in this section.

We now make the following definitions and a further assumption before stating the main results of this section:

\[\begin{pmatrix}
\mu_1 \\
\mu_2 \\
\vdots \\
\mu_N
\end{pmatrix} = \Gamma^{-1}
\begin{pmatrix}
\delta_1 \\
\delta_2 \\
\vdots \\
\delta_N
\end{pmatrix} \] (32)

where,
\[
\Gamma = 
\begin{bmatrix}
\bar{\eta}_1 & -\alpha_{12} & \ldots & -\alpha_{1N} \\
-\alpha_{21} & \bar{\eta}_2 & \ldots & \vdots \\
\vdots & \ddots & \ddots & \tilde{\alpha}_{N-1,N} \\
-\alpha_{N1} & \ldots & -\tilde{\alpha}_{NN-1} & \bar{\eta}_N
\end{bmatrix}
\] (33)
\[\bar{\eta}_i = 4\beta_i A_i\sigma_i^2 - 18\beta_i^2 A_i\sigma_i^4 - 2\beta_i^2 \sigma_i^2 \sum_{j=1}^{N} A_j \sigma_j^2 \] (34)
\[
\tilde{\alpha}_{ij} = 2\beta_i y_{min} A_j \sigma_j^2 + 5\beta_i^2 A_j^2 \sigma_i^2 \sigma_j^2 + 6\beta_i^2 A_i \sigma_i^2 \] (35)
\[\delta_i = 15\beta_i^2 A_i^2 \sigma_i^6 + 6\beta_i A_i y_{min} \sigma_i^4 + \beta_i^2 \sigma_i^2 y_{min} \] (36)

**Assumption 4.4:** The eigenvalues of the matrix \( (I - \Gamma) \) are inside the unit disc.

**Theorem 4:** Given Assumptions 3.1, 4.3 and 4.4, using Gaussian perturbations satisfying Assumption 2.1 the covariance matrix of the (vector) state approaches the following limit as \( k \to \infty \):
\[\text{cov}(x_k) = \text{diag}[\mu_1, \ldots, \mu_N] + O(\beta)I \] (37)

**Remark 4:** Note that in the case \( N = 1 \) the covariance matrix limit reduces to
\[\text{cov}(x) = \frac{15\beta^2 A^2 \sigma^6 + 6\beta A y_{min} \sigma^4 + \beta^2 \sigma^2 y_{min}}{4\beta A \sigma^2 - 18\beta^2 A^2 \sigma^4 - 2\beta^2 \sigma^2 A y_{min}} + O(\beta) \] (38)

as expected from Theorem 2 with the tightening of Assumption 4.2 to 4.3.

**Proof:** The following identity holds for each element of the covariance matrix:
\[E(x_{i,k+1}x_{j,k+1}) = E[(x_{i,k} - \beta v_{i,k} y_k)(x_{j,k} - \beta v_{j,k} y_k)] \] (39)
For the off-diagonal terms, i.e. $i \neq j$ in (39), Assumption 3.1 means that (39) reduces to
\[
E(x_{i,k+1}x_{j,k}) = E(x_{i,k}x_{j,k}) - \beta E(x_{i,k}v_{j,k}y_{k}) - \beta E(x_{j,k}v_{i,k}y_{k})
\]
(40)

We also note for each $i$ and $j$ the following holds:
\[
E(x_{i,k}v_{j,k}y_{k}) = 2\sigma_{i}^{2}A_{j}E(x_{i,k}x_{j,k})
\]
(41)

Subsequently, the off-diagonal terms in the covariance matrix follow the sequence
\[
E(x_{i,k+1}x_{j,k+1}) = E(x_{i,k}x_{j,k})[1 - 2A_{j}\beta\sigma_{i}^{2} - 2A_{i}\beta\sigma_{j}^{2}]
\]
\[
\rightarrow 0 \quad \text{as} \quad k \rightarrow \infty \quad \text{for} \quad \beta < \frac{1}{2A_{i}\sigma_{i}^{2} + 2A_{j}\sigma_{j}^{2}}
\]
(42)

Therefore, for small $\beta$ (which is implicit from Assumptions 4.3 - 4.4) it follows that all the off-diagonal terms in the covariance matrix are convergent toward zero. We now address the diagonal terms in the covariance matrix. Substituting $i = j$ into (39) leads to:
\[
E(x_{i,k+1}^{2}) = E(x_{i,k}^{2}) - 2\beta E(x_{i,k}v_{i,k}y_{k}) + \beta^{2}E(v_{i,k}y_{k}^{2})
\]
(43)

The second term on the right hand side of (43) is covered by (41), so we now focus on the final term:
\[
E(v_{i,k}y_{k}^{2}) = E(v_{i,k}^{2})(x_{k} + v_{k})^{2} + y_{\min}^{2})
\]
\[
= E(v_{i,k}^{2})(\sum_{j=1}^{N} A_{j}(x_{j,k}^{2} + v_{j,k}^{2})^{2})
\]
\[
+ E(4v_{i,k}^{2}(\sum_{j=1}^{N} A_{j}x_{j,k}v_{j,k})^{2}) + E(v_{i,k}y_{k}^{2})
\]
\[
+ E(2v_{i,k}y_{k}^{2}\sum_{j=1}^{N} A_{j}(x_{j,k}^{2} + v_{j,k}^{2}))
\]
(44)

Space constraints prevent a detailed expansion at this point, however the methodology is similar to the scalar case. Applying (42) and Assumption 4.3 allows the expanded form of the first two terms on the RHS of (44) to be simplified. These simplified equations, along with the expansion of the final two terms of (44) lead to the coupled system:
\[
E\begin{bmatrix} x_{1}^{2} \\ x_{2}^{2} \\ \vdots \\ x_{N}^{2} \end{bmatrix} = \Gamma E\begin{bmatrix} x_{1}^{2} \\ x_{2}^{2} \\ \vdots \\ x_{N}^{2} \end{bmatrix} + \delta_{1} \\ \delta_{2} \\ \vdots \\ \delta_{N} \end{bmatrix}
\]
(45)

The result of Theorem 4 now follows directly when Assumption 4.4 is satisfied.

We now also extend Theorem 3 to the vector case.

**Theorem 5:** Given Assumptions 3.1, 4.3 and 4.4, using Gaussian perturbations satisfying Assumption 2.1 the output bias from $y_{\min}$ approaches the following limit as $\beta \rightarrow 0^{+}$:
\[
E(y_{k} - y_{\min}) \rightarrow \sum_{i=1}^{N} A_{i}\sigma_{i}^{2} \text{ as } \beta \rightarrow 0^{+}
\]
(46)

**Proof:** The proof is straightforward using the result of Theorem 4 and the properties of the stochastic signal outlined in Assumption 2.1.

V. SIMULATION RESULTS

A. Scalar states and perturbations

We consider a scalar state and output mapping $y = 2(x + 2)^{2}$. The initial value of the state is set to $x_{0} = 20$, the ES parameters are $\beta = 10^{-2}$ and Gaussian perturbations with $\sigma^{2} = 1$ are used. Figure 2 and 3 illustrate the convergence to the extremum at the origin using two different realizations of the perturbation sequence, $v_{k}$.

![Figure 2](image1.png)

**Fig. 2.** Convergence of $x_{k}$ in the mapping $y = 2(x + 2)^{2}$ for two different realisations of $v_{k}$. ES parameters are set to $\sigma^{2} = 1, \beta = 10^{-2}$

![Figure 3](image2.png)

**Fig. 3.** Convergence of $y_{k}$ for the mapping $y = 2(x + 2)^{2}$ for two different realisations of $v_{k}$. ES parameters are set to $\sigma^{2} = 1, \beta = 10^{-2}$

From Fig 2 we see that although the extremum at $x = -2$ is attained, there is a bias in $y$ as expected from Theorem 3. It is also important to note that although the averaged system will have monotonic convergence, there are no guarantees of this for the true system. The sample variances (after a large number of iterations) for $x$ and $y$ were found for single runs using different ES parameter values and compared to those predicted by Theorem 2 and 3. The results shown in Table I exhibit good correlation between the theory and simulations.

B. Vector states and perturbations

We now present results for a two dimensional state and perturbation. The input output mapping is given by $y = $
TABLE I
PREDICTED AND SIMULATED RESULTS FOR SCALAR SYSTEMS

<table>
<thead>
<tr>
<th>(A, β, σ^2, y_{min})</th>
<th>Predicted Sample E(x^2)</th>
<th>Predicted Sample E(x^2)</th>
<th>Predicted var(y)</th>
<th>Predicted var(y)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(2,10^{-2},1,0)</td>
<td>0.082</td>
<td>0.077</td>
<td>9.4</td>
<td>9.4</td>
</tr>
<tr>
<td>(1,10^{-2},1,0)</td>
<td>0.039</td>
<td>0.042</td>
<td>2.2</td>
<td>2.2</td>
</tr>
<tr>
<td>(1,10^{-3},1,0)</td>
<td>3.8 x 10^{-3}</td>
<td>3.2 x 10^{-3}</td>
<td>2.02</td>
<td>2.04</td>
</tr>
<tr>
<td>(1,10^{-3},0.25,0)</td>
<td>3.6 x 10^{-6}</td>
<td>2.4 x 10^{-6}</td>
<td>7.8 x 10^{-3}</td>
<td>7.7 x 10^{-3}</td>
</tr>
<tr>
<td>(2,10^{-2},0.5,2)</td>
<td>0.016</td>
<td>0.017</td>
<td>0.57</td>
<td>0.56</td>
</tr>
</tbody>
</table>

TABLE II
PREDICTED AND SIMULATED COVARIANCES WITH 2D SYSTEMS

<table>
<thead>
<tr>
<th>Map constants A, y_{min}</th>
<th>ES constants β x 10^3, σ^2</th>
<th>Predicted cov(x) x 10^{-2}</th>
<th>Sample cov(x) x 10^{-2}</th>
</tr>
</thead>
<tbody>
<tr>
<td>[ 3 0 ] , 10 1, 1</td>
<td>4.3 0</td>
<td>5.1 -0.1</td>
<td>0.1 8.1</td>
</tr>
<tr>
<td>[ 0 2 ] , 0 1, 2</td>
<td>0.82</td>
<td>0.1 -1.1</td>
<td>1.1 1.1</td>
</tr>
<tr>
<td>[ 1 0 ] , 5 2</td>
<td>1.8 0.79</td>
<td>2.7 0.1</td>
<td>0.1 8.7</td>
</tr>
</tbody>
</table>

Using Theorem 4, the predicted covariance matrix can be calculated and compared to the sample covariance matrix. The first row of Table II shows this comparison, while good agreement is observed for other ES parameters and maps in the remainder of the table.

As in the scalar case, we notice that the convergence of the states is non-monotonic, but nonetheless convergence to the extremum occurs provided the perturbations do not cause an exit from the domain of attraction of the closed loop system.

VI. Conclusions

In the absence of actuator dynamics, it was shown that if the parameters of the stochastic ES algorithm are chosen sufficiently small then convergence of the averaged closed loop system towards the extremum of the input-output mapping could be guaranteed, and the difference between the averaged and true systems was bounded in the steady state limit. Furthermore, when independent Gaussian distributed perturbations are used, the behaviour of the system at the extremum was quantified in terms of (co)variance of the state and plant output given small updates. Future areas for research include the effect of plant dynamics and washout filters, continuous time formulations, and time varying input-output maps.

REFERENCES