2016 IEEE 55th Conference on Decision and Control (CDC)
ARIA Resort & Casino
December 12-14, 2016, Las Vegas, USA

Improved Hessian estimation for adaptive random directions stochastic approximation

D. Sai Koti Reddy[†], Prashanth L.A.[♯], Shalabh Bhatnagar[‡]

***Abstract*— We propose an improved Hessian estimation scheme for the second-order random directions stochastic approximation (2RDSA) algorithm [1]. The proposed scheme, inspired by [2], reduces the error in the Hessian estimate by (i) incorporating a zero-mean feedback term; and (ii) optimizing the step-sizes used in the Hessian recursion of 2RDSA. We prove that 2RDSA with our Hessian improvement scheme (2RDSA-IH) converges asymptotically to the true Hessian. The advantage with 2RDSA-IH is that it requires only 75% of the simulation cost per-iteration for 2SPSA with improved Hessian estimation (2SPSA-IH) [2]. Numerical experiments show that 2RDSA-IH outperforms both 2SPSA-IH and 2RDSA without the improved Hessian estimation scheme.**

I. INTRODUCTION

Optimization problems involving uncertainties are very common in a variety of engineering disciplines such as transportation systems, manufacturing, networks, healthcare and finance. The large number of input variables and the lack of precise system model may prohibit analytical solution approaches and a viable alternative is to employ a simulation-based optimization approach. As illustrated in Figure 1, the idea here is to simulate a few times the stochastic system under consideration until a good enough solution is obtained. Formally, given only noise-corrupted measurements of an objective function f, we want to solve the following problem:

$$\text{Find } x^* = \arg\min_{x\in\mathbb{R}^N} f(x). \tag{1}$$

A natural solution approach is to devise an algorithm that incrementally updates the parameter, say x_n, in the descent direction using the gradient and/or Hessian of the objective f. However, in practice, one can only obtain estimates of the function f through black-box simulation and the challenge is to estimate the gradient and/or Hessian of f from function samples.

Simultaneous perturbation (SP) methods are a popular and efficient approach for estimating gradient/Hessian from function samples, especially in high dimensional problems - see [3] for a comprehensive treatment of this subject matter. Simultaneous perturbation stochastic approximation (SPSA) is a popular SP method. The first-order SPSA algorithm, henceforth referred to as 1SPSA, was proposed in [4]. A closely related algorithm is random directions stochastic approximation (RDSA) [5, pp. 58-60]. The gradient estimate in RDSA differs from that in SPSA, both in the construction as well as in the choice of random perturbations. In [5], the random perturbations for 1RDSA were generated by picking samples uniformly on the surface of a sphere and the resulting 1RDSA scheme was found to be inferior to 1SPSA from an asymptotic convergence rate viewpoint - see [6]. Recent work in [1] attempts to bridge the gap between 1RDSA and 1SPSA by incorporating random perturbations based on an asymmetric Bernoulli distribution as well as those with the i.i.d uniform distribution. However, 1SPSA was found to be still marginally better than 1RDSA.

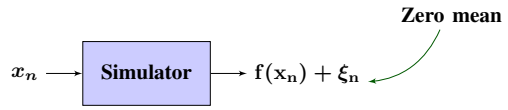

Fig. 1: Simulation optimization

Stochastic Newton methods can counter the ill-conditioning of the objective f as they incorporate second-order information into the update iteration given by

$$x_{n+1} = x_n - a_n(\overline{H}_n)^{-1}\widehat{\nabla} f(x_n), \tag{2}$$

where a_n is the step-size that satisfies standard stochastic approximation conditions (see (C5) in Section III), $\widehat{\nabla} f(x_n)$ and $\overline{H}_n$ are estimates of the gradient and Hessian, respectively. Thus, (2) can be considered as the stochastic version of the well-known Newton method for optimization.

In [7], an estimation scheme for $\overline{H}_n$ that uses $O(N^2)$ function samples per-iteration of (2) was proposed. The number of samples per-iteration was brought down to four, irrespective of dimension N, by the second-order SPSA algorithm (henceforth referred to as 2SPSA). In contrast to the case of 1SPSA vs 1RDSA, the results in [1] show that a second order RDSA approach (referred to as 2RDSA hereafter) can considerably outperform than 2SPSA [8], while requiring only three simulations per iteration of (2).

Our work in this paper is centred on improving the 2RDSA scheme of [1] by

- **I** reducing the error in the Hessian estimate through a feedback term; and
- **II** optimizing the step-sizes used in the Hessian estimation recursion, again with the objective of improving the quality of the Hessian estimate.

Items (I) and (II) are inspired by the corresponding improvements to the Hessian estimation recursion in the

[†] Department of Computer Science and Automation, Indian Institute of Science, Bangalore, E-Mail: danda.reddy@csa.iisc.ernet.in.
[♯] Institute for Systems Research, University of Maryland, College Park, Maryland, E-Mail: prashla@isr.umd.edu.
[‡] Department of Computer Science and Automation, Indian Institute of Science, Bangalore, E-Mail: shalabh@csa.iisc.ernet.in.
∗ Supported by NSF under Grants CMMI-1434419, CNS-1446665, and CMMI-1362303, by AFOSR under Grant FA9550-15-10050, and by the Robert Bosch Centre for Cyber-Physical Systems, IISc.

enhanced 2SPSA from [2]. We shall refer to the latter algorithm as 2SPSA-IH. While item (II) above is a relatively straightforward migration to the 2RDSA setting, item (I) is a non-trivial contribution, primarily because the Hessian estimate in 2RDSA is entirely different from that in 2SPSA and the feedback term that we incorporate in 2RDSA to improve the Hessian estimate neither correlates with that in 2SPSA nor follows from the analysis in [2]. The advantage with 2RDSA scheme along with proposed improvement to Hessian estimation (henceforth referred to as 2RDSA-IH) is that it requires only 75% of the simulation cost per-iteration for 2SPSA-IH.

We establish that the proposed improvements to Hessian estimation in 2RDSA are such that the resulting 2RDSA-IH algorithm is provably convergent, in particular, the Hessian estimate $\overline{H}_n$ of 2RDSA-IH converges almost surely to the true Hessian. Further, we show empirically that 2RDSA-IH outperforms both 2SPSA-IH of [2] and regular 2RDSA of [1]. Our contribution is important because 2RDSA-IH, like 2RDSA, has lower simulation cost per iteration than 2SPSA and unlike 2RDSA, has an improved Hessian estimation scheme.

The rest of the paper is organized as follows: In Section II, we describe the improved Hessian estimation scheme, which is incorporated into the 2RDSA algorithm from [1]. In Section III, we present the theoretical results for the 2RDSA algorithm with improved Hessian estimation. In Section IV, we present the results from numerical experiments and finally, in Section V, provide the concluding remarks.

II. SECOND-ORDER RDSA WITH IMPROVED HESSIAN ESTIMATION (2RDSA-IH)

The second-order RDSA with improved Hessian estimate performs an update iteration as follows:

$$x_{n+1} = x_n - a_n \Upsilon(\overline{H}_n)^{-1} \widehat{\nabla} f(x_n), \tag{3}$$

$$\overline{H}_n = (1-b_n)\overline{H}_{n-1} + b_n(\widehat{H}_n - \widehat{\Psi}_n), \tag{4}$$

where $\widehat{\nabla} f(x_n)$ is the estimate of $\nabla f(x_n)$, $\overline{H}_n$ is an estimate of the true Hessian $\nabla^2 f(\cdot)$, $\Upsilon(\cdot)$ projects any matrix onto the set of positive definite matrices and $\{a_n, n \geq 0\}$ is a step-size sequence that satisfies standard stochastic approximation conditions. There are standard procedures such as Cholesky factorization, see [9], for projecting a given square matrix to set of positive definite matrices. Moreover, in the vicinity of a local minimum, one expects the Hessian to be positive definite. In such a case, Υ will represent the identity operator.

The recursion (3) is identical to that in 2RDSA, while the Hessian estimation recursion (4) differs as follows:
(i) $\widehat{\Psi}_n$ is a zero-mean feedback term that reduces the error in Hessian estimate; and
(ii) b_n is a general step-size that we optimize to improve the Hessian estimate.

On the other hand, $\widehat{H}_n$ is identical to that in 2RDSA, i.e., it estimates the true Hessian in each iteration using 3 function evaluations. For the sake of completeness, we provide below the construction for $\widehat{\nabla} f(x_n)$ and $\widehat{H}_n$ using asymmetric Bernoulli perturbations, before we present the feedback term that reduces the error in $\widehat{H}_n$.

Algorithm 1 Structure of 2RDSA-IH algorithm.

Input: initial parameter $x_0 \in \mathbb{R}^N$, perturbation constants $\delta_n > 0$, step-sizes $\{a_n, b_n\}$, operator Υ.
for $n = 0, 1, 2, \ldots$ **do**
Generate $\{d_n^i, i = 1, \ldots, N\}$, independent of $\{d_m, m = 0, 1, \ldots, n-1\}$.
For any $i = 1, \ldots, N$, d_n^i is distributed either as an asymmetric Bernoulli (see (5)) or Uniform $U[-\eta, \eta]$ for some $\eta > 0$ (see Remark 1).
Function evaluation 1
Obtain $y_n^+ = f(x_n + \delta_n d_n) + \xi_n^+$.
Function evaluation 2
Obtain $y_n^- = f(x_n - \delta_n d_n) + \xi_n^-$.
Function evaluation 3
Obtain $y_n = f(x_n) + \xi_n$.
Newton step
Update the parameter and Hessian as follows:

$$x_{n+1} = x_n - a_n \Upsilon(\overline{H}_n)^{-1} \widehat{\nabla} f(x_n),$$
$$\overline{H}_n = (1-b_n)\overline{H}_{n-1} + b_n(\widehat{H}_n - \widehat{\Psi}_n),$$

where $\widehat{H}_n$ and $\widehat{\Psi}_n$ are chosen according to (7) and (16), respectively.
end for
Return x_n.

Algorithm 1 presents the pseudocode and we describe the individual component of 2RDSA-IH below.

A. Function evaluations

Let $\delta_n, n \geq 0$ denote a sequence of diminishing positive real numbers and $d_n = (d_n^1, \ldots, d_n^N)^\intercal$ denote a random perturbation vector at instant n, where the perturbations $\{d_n^i, i = 1, \ldots, N, n = 1, 2, \ldots\}$ are i.i.d. and distributed as follows:

$$d_n^i = \begin{cases} -1 & \text{w.p. } \dfrac{(1+\epsilon)}{(2+\epsilon)}, \\ 1+\epsilon & \text{w.p. } \dfrac{1}{(2+\epsilon)}, \end{cases} \tag{5}$$

with $\epsilon > 0$ being a constant that can be chosen to be arbitrarily small.

The 2RDSA-IH algorithm obtains three function samples y_n, y_n^+ and y_n^- at x_n, $x_n + \delta_n d_n$ and $x_n - \delta_n d_n$, respectively, i.e., $y_n = f(x_n) + \xi_n$, $y_n^+ = f(x_n + \delta_n d_n) + \xi_n^+$ and $y_n^- = f(x_n - \delta_n d_n) + \xi_n^-$, where the noise terms ξ_n, ξ_n^+, ξ_n^- satisfy $\mathbb{E}\left[\xi_n^+ + \xi_n^- - 2\xi_n | \mathcal{F}_n\right] = 0$ with $\mathcal{F}_n = \sigma(x_m, m \leq n)$ denoting the underlying sigma-field.

B. Gradient estimation

The RDSA estimate of the gradient $\nabla f(x_n)$ is given by

$$\widehat{\nabla} f(x_n) = \frac{1}{1+\epsilon} d_n \left[\frac{y_n^+ - y_n^-}{2\delta_n}\right], \tag{6}$$

C. Hessian estimation

$$\widehat{H}_n = M_n\left(\frac{y_n^+ + y_n^- - 2y_n}{\delta_n^2}\right), \text{ where} \tag{7}$$

$$M_n = \begin{bmatrix} \frac{1}{\kappa}\left((d_n^1)^2 - (1+\epsilon)\right) & \cdots & \frac{1}{2(1+\epsilon)^2} d_n^1 d_n^N \\ \frac{1}{2(1+\epsilon)^2} d_n^2 d_n^1 & \cdots & \frac{1}{2(1+\epsilon)^2} d_n^2 d_n^N \\ \cdots & \cdots & \cdots \\ \frac{1}{2(1+\epsilon)^2} d_n^N d_n^1 & \cdots & \frac{1}{\kappa}\left((d_n^N)^2 - (1+\epsilon)\right) \end{bmatrix},$$

where $\kappa = \tau\left(1 - \frac{(1+\epsilon)^2}{\tau}\right)$ and $\tau = E(d_n^i)^4 = \frac{(1+\epsilon)(1+(1+\epsilon)^3)}{(2+\epsilon)}$, for any $i = 1, \ldots, N$.

D. Improved Hessian estimation

The Hessian estimate $\widehat{H}_n$ can be simplified as follows:

$$\begin{aligned}\widehat{H}_n &= M_n\left(\frac{y_n^+ + y_n^- - 2y_n}{\delta_n^2}\right) \\ &= M_n\left[\left(\frac{f(x_n + \delta_n d_n) + f(x_n - \delta_n d_n) - 2f(x_n)}{\delta_n^2}\right)\right. \\ &\qquad \left. + \left(\frac{\xi_n^+ + \xi_n^- - 2\xi_n}{\delta_n^2}\right)\right] \\ &= M_n\left(d_n^\intercal \nabla^2 f(x_n) d_n + O(\delta_n^2) + \left(\frac{\xi_n^+ + \xi_n^- - 2\xi_n}{\delta_n^2}\right)\right).\end{aligned} \tag{8}$$

For the first term on the RHS above, note that

$$\mathbb{E}[M_n\left(d_n^\intercal \nabla^2 f(x_n) d_n\right) \mid \mathcal{F}_n] = \mathbb{E}\left[M_n \times \left(\sum_{i=1}^{N-1}(d_n^i)^2 \nabla_{ii}^2 f(x_n) + 2\sum_{i=1}^{N}\sum_{j=i+1}^{N} d_n^i d_n^j \nabla_{ij}^2 f(x_n)\right)\middle| \mathcal{F}_n\right]. \tag{9}$$

In analyzing the lth diagonal term of the above expression, the following zero-mean term appears (see the proof of Lemma 4 in [1]):

$$\mathbb{E}\left[([M_n]_D)_{l,l}\left(2\sum_{i=1}^{N-1}\sum_{j=i+1}^{N} d_n^i d_n^j \nabla_{ij}^2 f(x_n)\right)\middle| \mathcal{F}_n\right] = 0, \tag{10}$$

where for any matrix M, $[M]_D$ refers to a matrix that retains only the diagonal entries of M and replaces all the remaining entries with zero, and $([M]_D)_{i,j}$ refers to the (i,j)th entry in $[M]_D$. We shall also use $[M]_N$ to refer to a matrix that retains only the off-diagonal entries of M, while replaces all the diagonal entries with zero.

The term on the LHS in (10), denoted by $\Psi_n^1(\nabla^2 f(x_n))$, can be written in matrix form as follows:

$$\Psi_n^1(\nabla^2 f(x_n)) = [M_n]_D\left(d_n^\intercal [\nabla^2 f(x_n)]_N d_n\right). \tag{11}$$

In analyzing the off-diagonal term ((k,l) where $k < l$) of (9), the following zero-mean term appears:

$$\mathbb{E}\left[([M_n]_N)_{k,l}\left(\sum_{i=1}^{N}(d_n^i)^2 \nabla_{ii}^2 f(x_n)\right)\middle| \mathcal{F}_n\right] = 0. \tag{12}$$

The term on the LHS above, denoted by $\Psi_n^2(\nabla^2 f(x_n))$, can be written in matrix form as follows:

$$\Psi_n^2(\nabla^2 f(x_n)) = [M_n]_N\left(d_n^\intercal [\nabla^2 f(x_n)]_D d_n\right). \tag{13}$$

From the foregoing, the per-iteration Hessian estimate $\widehat{H}_n$ can be re-written as follows:

$$\begin{aligned}\mathbb{E}\left[\widehat{H}_n \middle| \mathcal{F}_n\right] =& \nabla^2 f(x_n) + \mathbb{E}\left[\Psi_n(\nabla^2 f(x_n)) \middle| \mathcal{F}_n\right] + O(\delta_n^2) \\ &+ \mathbb{E}\left[\left(\frac{\xi_n^+ + \xi_n^- - 2\xi_n}{\delta_n^2}\right)\middle| \mathcal{F}_n\right],\end{aligned} \tag{14}$$

where, for any matrix H,

$$\begin{aligned}\Psi_n(H) &= \Psi_n^1(H) + \Psi_n^2(H) \\ &= [M_n]_D\left(d_n^\intercal [H]_N d_n\right) + [M_n]_N\left(d_n^\intercal [H]_D d_n\right).\end{aligned} \tag{15}$$

In the RHS of (14), it is easy to see that the second term involving Ψ_n and the last term involving the noise are zero-mean. Moreover, since the noise is bounded by assumption, the last term in (14) vanishes asymptotically at the rate $O(\delta_n^{-2})$. So, the error in estimating the Hessian is due to the second term, which involves the perturbations d_n. This motivates the term $\widehat{\Psi}_n$ in the update rule (3).

Given that we operate in a simulation optimization setting, which implies $\nabla^2 f$ is not known, we construct the feedback term $\widehat{\Psi}_n$ in (3) by using $\overline{H}_{n-1}$ as a proxy for $\nabla^2 f$, i.e.,

$$\widehat{\Psi}_n = \Psi_n(\overline{H}_{n-1}). \tag{16}$$

E. Step-size optimization

Unlike the feedback term, adapting the idea of optimizing the step-sizes for 2RDSA is relatively straightforward from the corresponding approach for 2SPSA in [2]. The difference here is that there exists only one N-dimensional perturbation vector d_n in our setting, while 2SPSA required two such vectors. This in turn implies that only the perturbation constant δ_n is needed in optimizing b_n.

The optimal choice for b_n in (4) is the following:

$$b_i = \delta_i^4 / \sum_{j=0}^{i} \delta_j^4. \tag{17}$$

The main idea behind the above choice is provided below. From (14), we can infer that

$$\mathbb{E} \parallel \widehat{H}_n \parallel^2 \leq \frac{C}{\delta_n^4} \text{ for some } C < \infty.$$

This is because the third term in (14) vanishes asymptotically, while the fourth term there dominates asymptotically. Moreover, the noise factors in the fourth term in (14) are bounded above due to (C9) and independent of n, leaving the δ_n^2 term in the denominator there.

So, the optimization problem to be solved at instant n is as follows:

$$\min_{\{\tilde{b}_k\}} \sum_{i=0}^{n} (\tilde{b}_k)^2 \delta_i^{-4}, \text{ subject to} \tag{18}$$

$$\tilde{b}_i \geq 0 \ \forall i \text{ and } \sum_{i=0}^{n} \tilde{b}_i = 1. \tag{19}$$

The optimization variable $\tilde{b}_i$ from the above is related to the Hessian recursion (4) as follows:

$$\overline{H}_n = \sum_{i=0}^{n} \tilde{b}_k(\widehat{H}_i - \widehat{\Psi}_i). \tag{20}$$

The solution to (18) is achieved for $\tilde{b}_i^* = \delta_i^4 / \sum_{j=0}^{n} \delta_j^4, i = 1, \ldots, n$. The optimal choice $\tilde{b}_i^*$ can be translated to the step-sizes b_i, leading to (17).

Remark 1. ***(Uniform perturbations)*** *In [1], the authors suggest two alternatives for the distribution of random perturbations d_n: the asymmetric Bernoulli, which we described earlier and uniform that we outline next.*

Choose d_n^i, $i = 1, \ldots, N$ to be i.i.d. $U[-\eta, \eta]$ for some $\eta > 0$, where $U[-\eta, \eta]$ denotes the uniform distribution on the interval $[-\eta, \eta]$. Then, the RDSA estimate of the gradient is given by

$$\widehat{\nabla} f(x_n) = \frac{3}{\eta^2} d_n \left[\frac{y_n^+ - y_n^-}{2\delta_n} \right]. \tag{21}$$

The Hessian estimate in this case is given by

$$\widehat{H}_n = M_n \left(\frac{y_n^+ + y_n^- - 2y_n}{\delta_n^2} \right), \text{ where} \tag{22}$$

$$M_n = \frac{9}{2\eta^4} \begin{bmatrix} \frac{5}{2}\left((d_n^1)^2 - \frac{\eta^2}{3}\right) & \cdots & d_n^1 d_n^N \\ d_n^2 d_n^1 & \cdots & d_n^2 d_n^N \\ \cdots & \cdots & \cdots \\ d_n^N d_n^1 & \cdots & \frac{5}{2}\left((d_n^N)^2 - \frac{\eta^2}{3}\right) \end{bmatrix}.$$

The feedback term in (16) *can be easily extended to the case of uniform perturbations by using the M_n as defined above instead of that for the asymmetric Bernoulli case.*

III. CONVERGENCE ANALYSIS

We make the same assumptions as those used in the analysis of [1], with a few minor alterations. The assumptions are listed below:

(C1) The function f is four-times differentiable[1] with $\left|\nabla^4_{i_1 i_2 i_3 i_4} f(x)\right| < \infty$, for $i_1, i_2, i_3, i_4 = 1, \ldots, N$ and for all $x \in \mathbb{R}^N$.

(C2) For each n and all x, there exists a $\rho > 0$ not dependent on n and x, such that $(x - x^*)^\intercal \bar{f}_n(x) \geq \rho \|x_n - x\|$, where $\bar{f}_n(x) = \Upsilon(\overline{H}_n)^{-1} \nabla f(x)$.

(C3) $\{\xi_n, \xi_n^+, \xi_n^-, n = 1, 2, \ldots\}$ are such that, for all n, $\mathbb{E}[\xi_n^+ + \xi_n^- - 2\xi_n | \mathcal{F}_n] = 0$, where $\mathcal{F}_n = \sigma(x_m, m \leq n)$ denotes the underlying sigma-field..

(C4) $\{d_n^i, i = 1, \ldots, N, n = 1, 2, \ldots\}$ are i.i.d. and independent of $\mathcal{F}_n$.

(C5) The step-sizes a_n and perturbation constants δ_n are positive, for all n and satisfy

$$a_n, \delta_n \to 0 \text{ as } n \to \infty, \sum_n a_n = \infty \text{ and } \sum_n \left(\frac{a_n}{\delta_n}\right)^2 < \infty.$$

[1]Here $\nabla^4 f(x) = \dfrac{\partial^4 f(x)}{\partial x^\intercal \partial x^\intercal \partial x^\intercal \partial x^\intercal}$ denotes the fourth derivate of f at x and $\nabla^4_{i_1 i_2 i_3 i_4} f(x)$ denotes the $(i_1 i_2 i_3 i_4)$th entry of $\nabla^4 f(x)$, for $i_1, i_2, i_3, i_4 = 1, \ldots, N$.

(C6) For each $i = 1, \ldots, N$ and any $\rho > 0$, $P(\{\bar{f}_{ni}(x_n) \geq 0 \text{ i.o}\} \cap \{\bar{f}_{ni}(x_n) < 0 \text{ i.o}\} \mid \{|x_{ni} - x_i^*| \geq \rho \quad \forall n\}) = 0$.

(C7) The operator Υ satisfies $\delta_n^2 \Upsilon(H_n)^{-1} \to 0$ a.s. and $E(\left\|\Upsilon(H_n)^{-1}\right\|^{2+\zeta}) \leq \rho$ for some $\zeta, \rho > 0$.

(C8) For any $\tau > 0$ and nonempty $S \subseteq \{1, \ldots, N\}$, there exists a $\rho'(\tau, S) > \tau$ such that

$$\limsup_{n \to \infty} \left| \frac{\sum_{i \notin S} (x - x^*)_i \bar{f}_{ni}(x)}{\sum_{i \in S} (x - x^*)_i \bar{f}_{ni}(x)} \right| < 1 \text{ a.s.}$$

for all $|(x - x^*)_i| < \tau$ when $i \notin S$ and $|(x - x^*)_i| \geq \rho'(\tau, S)$ when $i \in S$.

(C9) For some $\alpha_0, \alpha_1 > 0$ and for all n, $\mathbb{E}{\xi_n}^2 \leq \alpha_0$, $\mathbb{E}\xi_n^{\pm 2} \leq \alpha_0$, $\mathbb{E} f(x_n)^2 \leq \alpha_1$, $\mathbb{E} f(x_n \pm \delta_n d_n)^2 \leq \alpha_1$ and $\mathbb{E}\left(\left\|\Upsilon(\overline{H}_n)\right\|^2 \mid \mathcal{F}_n\right) \leq \alpha_1$.

(C10) $\delta_n = \frac{\delta_0}{(n+1)^\varsigma}$, where $\delta_0 > 0$ and $0 < \varsigma \leq 1/8$.

The reader is referred to Section II-B of [1] for a detailed discussion of the above assumptions. We remark here that (C1)-(C8) are identical to that in [1], while (C9) and (C10) introduce minor additional requirements on $\left\|\Upsilon(\overline{H}_n)\right\|^2$ and δ_n, respectively and these are inspired by [2].

Lemma 1. ***(Bias in Hessian estimate)*** *Under (C1)-(C10), with $\widehat{H}_n$ defined according to* (7)*, we have a.s. that*[2]*, for $i, j = 1, \ldots, N$,*

$$\left| \mathbb{E}\left[\widehat{H}_n(i,j) \middle| \mathcal{F}_n \right] - \nabla^2_{ij} f(x_n) \right| = O(\delta_n^2). \tag{23}$$

Proof. See Lemma 4 in [1]. □

Theorem 2. ***(Strong Convergence of Hessian)*** *Under (C1)-(C10), we have that*

$$x_n \to x^*, \overline{H}_n \to \nabla^2 f(x^*) \text{ a.s. as } n \to \infty.$$

In the above, x_n and $\overline{H}_n$ are updated according to (3) *and* (4)*, respectively, $\widehat{H}_n$ defined according to* (7) *and the step-sizes b_n are chosen as suggested in* (17)*.*

Proof. The first part of the claim regarding x_n follows in exactly the same fashion as the proof of Theorem 5 in [1]. For proving the claim regarding $\overline{H}_n$, we closely follow the approach used to prove a corresponding result for 2SPSA (see Theorem 1 in [2]). The first step is to prove the following:

$$\sum_{k=0}^{n} \frac{\delta_k^4 \left(\widehat{H}_k - \widehat{\Psi}_k - \mathbb{E}(\widehat{H}_k \mid \mathcal{F}_k) \right)}{\sum_{i=0}^{n} \delta_i^4} \to 0. \tag{24}$$

By a completely parallel argument to that used in the proof of Theorem 1 in [2], we obtain: For any $i, j = 1, \ldots, N$,

$$\mathbb{E}\left[\left((\widehat{H}_k)_{i,j} - (\widehat{\Psi}_k)_{i,j} - \mathbb{E}((\widehat{H}_k)_{i,j} \mid \mathcal{F}_k) \right)^2 \right] = O(\delta_k^{-4}).$$

Now (24) follows by an application of Kronecker's Lemma along with the martingale convergence theorem (see Theorem 6.2.1 of [10]).

[2]Here $\widehat{H}_n(i,j)$ and $\nabla^2_{ij} f(\cdot)$ denote the (i,j)th entry in the Hessian estimate $\widehat{H}_n$ and the true Hessian $\nabla^2 f(\cdot)$, respectively.

From Lemma 1, we have

$$\mathbb{E}[\widehat{H}_k \mid \mathcal{F}_k] = \nabla^2 f(x_n) + O(\delta_n^2) \text{ a.s.}$$

Since the Hessian is continuous near x_n and x_n converges almost surely to x^*, we have

$$\begin{aligned}\sum_{k=0}^{n} \frac{\delta_k^4 \left(\mathbb{E}(\widehat{H}_k \mid \mathcal{F}_k)\right)}{\sum_{i=0}^{n} \delta_i^4} &= \sum_{k=0}^{n} \frac{\delta_k^4 \left(\nabla^2 f(x_n) + O(\delta_n^2)\right)}{\sum_{i=0}^{n} \delta_i^4} \\ &= \sum_{k=0}^{n} \frac{\delta_k^4 \left(\nabla^2 f(x^*) + o(1)\right)}{\sum_{i=0}^{n} \delta_i^4} \\ &\to \nabla^2 f(x^*) \text{ a.s. as } n \to \infty.\end{aligned}$$

The last step above follows from Toeplitz Lemma (see p. 89 of [10]) after observing that $\sum_{i=0}^{n} \delta_i^4 \to \infty$ due to (C10). The main claim now follows since

$$\overline{H}_n = \sum_{k=0}^{n} \frac{\delta_k^4 \left(\widehat{H}_k - \Psi_k\right)}{\sum_{i=0}^{n} \delta_i^4}.$$

□

IV. SIMULATION EXPERIMENTS

A. Implementation

We test the performance of 2RDSA-Unif, 2RDSA-AsymBer and 2SPSA, with/without improved Hessian estimation. 2SPSA algorithm uses Bernoulli ± 1-valued perturbations, while 2RDSA/2RDSA-IH come in two variants - one that uses $U[-1, 1]$ distributed perturbations (referred to as 2RDSA-Unif/2RDSA-IH-Unif) and the other that uses asymmetric Bernoulli perturbations (referred to as 2RDSA-AsymBer/2RDSA-IH-AsymBer)[3].

For the empirical evaluations, we use the following two loss functions in $N = 10$ dimensions:

a) Quadratic loss:

$$f(x) = x^{\mathsf{T}} A x + b^{\mathsf{T}} x. \tag{25}$$

The optimum x^* for the above f is such that each coordinate of x^* is -0.9091, with $f(x^*) = -4.55$.

b) Fourth-order loss:

$$f(x) = x^{\mathsf{T}} A^{\mathsf{T}} A x + 0.1 \sum_{j=1}^{N} (Ax)_j^3 + 0.01 \sum_{j=1}^{N} (Ax)_j^4. \tag{26}$$

The optimum x^* for above f is $x^* = 0$, with $f(x^*) = 0$.

In both functions, A is such that NA is an upper triangular matrix with each entry one, b is the N-dimensional vector of ones and the noise structure is similar to that used in [8]. For any x, the noise is $[x^{\mathsf{T}}, 1]z$, where $z \approx \mathcal{N}(0, \sigma^2 I_{11\times 11})$. We perform experiments for noisy as well as noise-less settings, with $\sigma = 0.1$ for the noisy case.

For all algorithms, we set $\delta_n = 3.8/n^{0.101}$ and $a_n = 1/n^{0.6}$, while b_n are set according to (17). These choices have been used for 2SPSA implementations before (see [8]) and have demonstrated good finite-sample performance empirically, while satisfying the theoretical requirements needed for asymptotic convergence. For all the algorithms, the initial point x_0 is the N-dimensional vector of ones. For both 2SPSA and 2RDSA/2RDSA-IH, an initial 20% of the simulation budget was used up by 1SPSA/1RDSA and the resulting iterate was used to initialize 2SPSA/2RDSA. The distribution parameter ϵ is set to 0.0001 for 2RDSA and to 0.01 for 1RDSA.

[3]The implementation is available at `https://github.com/prashla/RDSA/archive/master.zip`.

TABLE I: Normalized loss values for fourth-order objective (26) with and without noise: standard error from 500 replications shown after $\pm$

Noise parameter $\sigma = 0.1$		
	Regular	**Improved Hessian estimation**
2SPSA	0.132 ± 0.0267	0.104 ± 0.0355
2RDSA-Unif	0.115 ± 0.0214	0.0271 ± 0.0538
2RDSA-AsymBer	0.0471 ± 0.021	$\mathbf{0.0099 \pm 0.0014}$
Noise parameter $\sigma = 0$		
	Regular	**Improved Hessian estimation**
2SPSA	0.0795 ± 0.0234	0.0628 ± 0.0234
2RDSA-Unif	0.0813 ± 0.0275	0.0214 ± 0.00376
2RDSA-AsymBer	0.0199 ± 0.0114	$\mathbf{0.0098 \pm 0.00147}$

B. Results

We use normalized loss and normalized MSE (NMSE) as performance metrics for evaluating the algorithms. NMSE is the ratio $\|x_{n_{\text{end}}} - x^*\|^2 / \|x_0 - x^*\|^2$, while normalized loss is the ratio $f(x_{n_{\text{end}}})/f(x_0)$. Here n_{end} denotes the iteration number when the algorithm stopped updating its parameter. Note that n_{end} is a function of the simulation budget. 2RDSA/2RDSA-IH use only three simulations per-iteration and hence, n_{end} is $1/3$rd of the simulation budget, while it is $1/4$th of the simulation budget for 2SPSA, since the latter algorithm uses four simulations per-iteration.

Tables I–II present the normalized loss values observed for the three algorithms - 2SPSA, 2RDSA-Unif and 2RDSA-AsymBer - with/without improved Hessian estimation scheme and for the fourth-order and quadratic loss functions, respectively. Table III presents the NMSE values obtained for the aforementioned algorithms with the quadratic loss. The results in Tables I–III are obtained after running all the algorithms with a budget of 10000 function evaluations. Figure 2 plots the normalized loss as a function of the simulation budget with the fourth-order loss objective with $\sigma = 0.1$ From the results in Tables I–III and Fig 2, we make the following observations:

Observation 1: *Among 2RDSA schemes, 2RDSA-IH performs better than regular 2RDSA, for both perturbation choices.*

TABLE II: Normalized loss values for quadratic objective (25) with and without noise: standard error from 500 replications shown after $\pm$

Noise parameter $\sigma = 0.1$		
	Regular	**Improved Hessian estimation**
2SPSA	-0.0062 ± 0.1164	-0.1229 ± 0.1374
2RDSA-Unif	0.0485 ± 0.1465	-0.259 ± 0.0398
2RDSA-AsymBer	-0.2564 ± 0.068	$\mathbf{-0.2877 \pm 0.0051}$
Noise parameter $\sigma = 0$		
	Regular	**Improved Hessian estimation**
2SPSA	-0.0785 ± 0.1178	-0.1716 ± 0.1339
2RDSA-Unif	0.0326 ± 0.1599	-0.2672 ± 0.0299
2RDSA-AsymBer	-0.2777 ± 0.0488	$\mathbf{-0.2881 \pm 0.0012}$

TABLE III: NMSE values for quadratic objective (25) with and without noise: standard error from 500 replications shown after $\pm$

Noise parameter $\sigma = 0.1$		
	Regular	**Improved Hessian estimation**
2SPSA	0.9491 ± 0.0131	0.5495 ± 0.0217
2RDSA-Unif	1.0073 ± 0.0140	0.1953 ± 0.0095
2RDSA-AsymBer	0.1667 ± 0.0095	$\mathbf{0.0324 \pm 0.0007}$
Noise parameter $\sigma = 0$		
	Regular	**Improved Hessian estimation**
2SPSA	0.7325 ± 0.0180	0.3939 ± 0.0230
2RDSA-Unif	0.9834 ± 0.0170	0.1623 ± 0.0086
2RDSA-AsymBer	0.0686 ± 0.0078	$\mathbf{0.0316 \pm 0.0006}$

Observation 2: *2RDSA-IH variants outperform both 2SPSA and 2SPSA-IH, with 2RDSA-IH-AsymBer performing the best overall.*

V. CONCLUSIONS

We presented an improved Hessian estimation scheme for the 2RDSA algorithm [1]. The proposed scheme was shown to be provably convergent to the true Hessian. The advantage with 2RDSA-IH is that it requires only 75% of the simulation cost per-iteration for 2SPSA with Hessian estimation improvements (2SPSA-IH) [2]. Numerical experiments demonstrated that 2RDSA-IH outperforms both 2SPSA-IH and 2RDSA without the improved Hessian estimation scheme.

As future work, it would be interesting to derive finite time bounds that show a lower Hessian estimation error for 2RDSA-IH when compared to 2RDSA and 2SPSA.

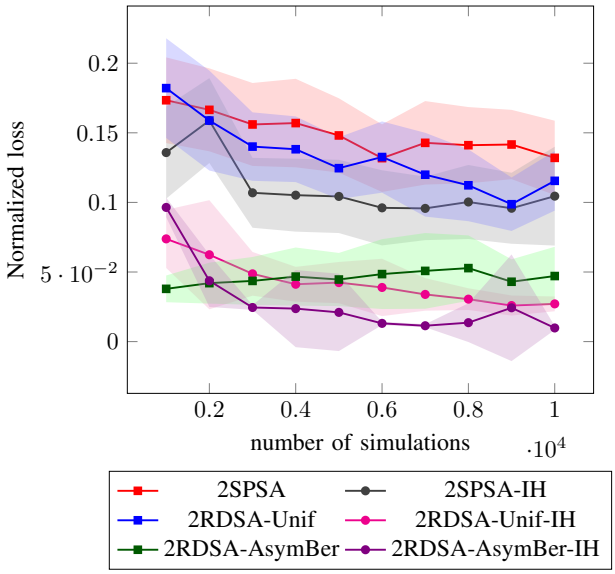

Fig. 2: Normalized loss vs. number of simulations for fourth-order loss (26) with $\sigma = 0.1$ for 2SPSA, 2RDSA-Unif and 2RDSA-AsymBer algorithms with/without improved Hessian estimation: bands around the curves represent standard error from 500 replications.

REFERENCES

[1] L. A. Prashanth, S. Bhatnagar, M. Fu, and S. Marcus, "Adaptive system optimization using random directions stochastic approximation," *IEEE Transactions on Automatic Control (To appear)*, 2017.

[2] J. C. Spall, "Feedback and weighting mechanisms for improving Jacobian estimates in the adaptive simultaneous perturbation algorithm," *IEEE Trans. Autom. Contr.*, vol. 54, no. 6, pp. 1216–1229, 2009.

[3] S. Bhatnagar, H. L. Prasad, and L. A. Prashanth, *Stochastic Recursive Algorithms for Optimization: Simultaneous Perturbation Methods (Lecture Notes in Control and Information Sciences)*. Springer, 2013, vol. 434.

[4] J. C. Spall, "Multivariate stochastic approximation using a simultaneous perturbation gradient approximation," *IEEE Trans. Auto. Cont.*, vol. 37, no. 3, pp. 332–341, 1992.

[5] H. J. Kushner and D. S. Clark, *Stochastic Approximation Methods for Constrained and Unconstrained Systems*. Springer Verlag, 1978.

[6] D. C. Chin, "Comparative study of stochastic algorithms for system optimization based on gradient approximations," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 27, no. 2, pp. 244–249, 1997.

[7] V. Fabian, "Stochastic approximation," in *Optimizing Methods in Statistics (ed. J.J.Rustagi)*. New York: Academic Press, 1971, pp. 439–470.

[8] J. C. Spall, "Adaptive stochastic approximation by the simultaneous perturbation method," *IEEE Trans. Autom. Contr.*, vol. 45, pp. 1839–1853, 2000.

[9] D. P. Bertsekas, *Nonlinear programming*. Athena Scientific, 1999.

[10] R. G. Laha and V. K. Rohatgi, *Probability Theory*. Wiley, New York, 1979.