Proceedings of the 2021 Winter Simulation Conference
S. Kim, B. Feng, K. Smith, S. Masoud, Z. Zheng, C. Szabo, and M. Loper, eds.

STOCHASTIC APPROXIMATION WITH GAUSSIAN PROCESS REGRESSION

Yingcui Yan
Haihui Shen

Sino-US Global Logistics Institute
Antai College of Economics and Management
Shanghai Jiao Tong University
1954 Huashan Road
Shanghai, 200030, CHINA

Zhibin Jiang

Sino-US Global Logistics Institute
and Department of Management Science
Antai College of Economics and Management
Shanghai Jiao Tong University
1954 Huashan Road
Shanghai, 200030, CHINA

ABSTRACT

Stochastic approximation (SA) is attractive due to its curse-of-dimensionality-free convergence rate, but its finite-sample performance is not always satisfying. In addition to improving SA solely, it is also a promising direction to combine SA with other simulation optimization methods together for better performance. In this paper we propose to integrate the original SA with Gaussian process (GP) regression, and call this algorithm SAwGP. The GP regression serves as a surrogate model and it uses all the past sampling information to guide the SA iteration, which tends to be beneficial especially in the early stage. We theoretically prove that integrating the surrogate model does not ruin the local convergence of SA, and numerically demonstrate that the finite-sample performance of SAwGP is better than the original SA while the rate of convergence does not deteriorate and is even enhanced.

1 INTRODUCTION

Stochastic approximation (SA) is one of the most classical and popular algorithms for continuous simulation optimization (or more generally, black-box optimization with noise). It was first proposed by Robbins and Monro (1951) to solve noisy root-finding problems, which is applicable to continuous simulation optimization with unbiased gradient estimator available. Kiefer and Wolfowitz (1952) extend SA to solve simulation optimization problems where the gradient needs to be estimated via finite difference (which is only asymptotically unbiased). SA is a recursive algorithm that can be viewed as the stochastic counterpart to gradient descent/ascent in deterministic optimization, so it is also called stochastic gradient descent/ascent (Newton et al. 2018). Comprehensive reviews on SA can be found in Kushner and Yin (2003) or Chau and Fu (2015).

Under mild conditions, SA has provable local convergence, and more importantly, the rate of convergence is irrelevant to the dimensionality of the problem, which is $O_p(n^{-1/2})$ for Robbins-Monro (RM) type and $O_p(n^{-1/3})$ for Kiefer-Wolfowitz (KW) type (Kushner and Clark 1978). Thanks to such curse-of-dimensionality-free convergence rate, SA and its variants have been widely applied in many complex problems, such as deep learning (Bottou 2010), neural networks (Bottou 2012) and manufacturing system control problems (Chen 2006). However, SA also has its own shortcomings. The finite-time or finite-sample performance of SA may be unsatisfying, and it is quite sensitive to many factors such as the step size, projection and gradient estimator (Chau and Fu 2015). In the past half-century, a large number of studies have tried to overcome these shortcomings, including modifying the iteration step size by averaging, building adaptive constraints for projection, constructing better (e.g., unbiased) gradient estimator and so on (Nemirovski et al. 2009; Kushner and Yin 2003; Fu and Ho 1988; Andradóttir 1996).

978-1-6654-3311-2/21/$31.00 ©2021 IEEE

In addition to improving the SA algorithms solely, it is also a promising direction to combine SA with other simulation optimization methods together to better solve the simulation optimization, with the hope that different methods have different strengths and they may be complementary to each other. The benefit of combing several (simulation) optimization methods has already been revealed in many other papers, including Chang et al. (2013), Sun et al. (2014), Sun et al. (2018), and Fan and Hu (2018), though SA is not involved there. Specifically, Chang et al. (2013) combines the response surface methodology (see Kleijnen (2015) for a review) with the trust-region method developed for deterministic optimization to solve continuous simulation optimization problems. Sun et al. (2014) integrates sampling distribution derived from a fast fitted Gaussian process (GP) into a random research algorithm. It is globally convergent when used to solve discrete simulation optimization problems. Sun et al. (2018) extend this method to continuous simulation optimization problems. Fan and Hu (2018) integrates the response surface methodology with COMPASS (Hong and Nelson 2006) originally developed for discrete simulation optimization. It is locally convergent when used to solve Lipschitz continuous simulation optimization problems. Essentially, all these papers construct some surrogate model and integrate it with other (simulation) optimization method, and the benefit of doing so is demonstrated therein.

Inspired by these works, we consider to integrate SA with some surrogate model to improve the performance. The intuition is as follows. In the early iteration of SA, the step size is usually large, which can easily make the algorithm oscillatory. And the noisy gradient estimator may make it even worse. Besides, original SA does not utilize any past sampling information, which is a huge waste. It should be beneficial if the past sampling information can be used to suggest SA where to go and focus, especially in the early stage. In contrast, the surrogate model is usually good at using previously sampled points to predict response surface in less sampled region and identify promising area. This may bring practical value during the search of optimal solution when the sample size is finite. On the other hand, it is also very important to ensure that, while improving the finite-sample performance, the integrated surrogate model should not ruin the convergence of SA, especially the asymptotic rate of convergence that is curse-of-dimensionality-free. Otherwise, the integration does not fully take advantage of the strength of SA.

Speaking of surrogate models, there are various choices including interpolation, regression, and even neural network model. But for continuous simulation optimization algorithms that only run a single simulation replication at each sampled point (including SA), care needs to be taken to cope with the simulation noises at all sampled points. The algorithm in Fan and Hu (2018) also only requires single replication. They first uses the shrinking ball technique to estimate the response surface at the sampled points (the noises inside the shrinking ball tend to cancel each other out), and then adopts a radial basis function interpolation to construct the surrogate model. The reason why the second step is necessary is that directly using shrinking ball technique to predict the entire response surface will result in a nonsmooth surrogate model. We instead choose to use the GP regression (Rasmussen and Williams 2005), which is also known as stochastic kriging (Ankenman et al. 2010), since it naturally constructs the smooth surrogate model in one step. Moreover, GP regression has been widely used in machine learning, engineering, and other fields, and many nice properties including continuity and convergence have been well established (Adler and Taylor ; Bect et al. 2019).

In summary, in this paper we propose to integrate the original SA with GP regression, and call this algorithm SA with GP regression (SAwGP). To the best of our knowledge, it is the first time in literature to consider such integration and investigate the resulting benefits. In particular, we focus on the case where the unbiased gradient estimator is unavailable, i.e., the KW type SA, while it can be easily adapted to the RM type SA. As the first attempt, we consider extremely simple integration mechanism of SA and GP, which is described in the following. After every certain iterations of SA, a surrogate model for the response surface is built via GP regression using all the previously sampled points. Then, let SA start at the estimated best solution found from the surrogate model. The local convergence of SAwGP is theoretically proved, and numerical experiments shows that SAwGP maintains the same rate of asymptotic convergence as the original SA.

The remainder of the paper is organized as follows. Section 2 introduces the problem formulation and briefly summarize SA and GP regression. Section 3 describes the proposed SAwGP algorithm and proves its local convergence. Numerical experiments are conducted in Section 4 to demonstrate both the finite-sample performance and asymptotic behavior of SAwGP. Finally, in Section 5 we make some conclusions and suggest some directions for future research.

2 PROBLEM FORMULATION

Consider the following continues simulation optimization problem

$$\max_{\boldsymbol{x}\in\mathbb{X}} h(\boldsymbol{x}),$$

where $h(\boldsymbol{x}) = \mathbb{E}[Y(\boldsymbol{x};\boldsymbol{\omega})]$ is a deterministic, real-valued response surface function, $Y(\boldsymbol{x};\boldsymbol{\omega})$ is a sample response, and $\boldsymbol{\omega}$ denotes the randomness. The solution space $\mathbb{X}$ is a convex and compact set in $\mathbb{R}^d$ with nonempty interior, and $h(\boldsymbol{x})$ is assumed to be continuous in $\mathbb{X}$. Besides, $h(\boldsymbol{x})$ cannot be analytically computed and only the random variable $Y(\boldsymbol{x};\boldsymbol{\omega})$ can be observed via simulation experiments (or more generally, sampling). From now on, the argument $\boldsymbol{\omega}$ is omitted for notational convenience when there is no ambiguity. Denote $\sigma^2(\boldsymbol{x}) \coloneqq \mathrm{Var}(Y(\boldsymbol{x}))$.

2.1 KW Type SA

SA constructs an iterative searching sequence $\{\boldsymbol{X}_n\}_{n\geq 1}$ as follows:

$$\boldsymbol{X}_{n+1} \coloneqq \Pi_{\mathbb{X}}\left(\boldsymbol{X}_n + a_n\hat{\nabla}h(\boldsymbol{X}_n)\right), \tag{1}$$

where $\{a_n\}_{n\geq 1}$ is a deterministic positive sequence for step size, $\hat{\nabla}h(\boldsymbol{X}_n)$ is an estimator of the gradient $\nabla h(\boldsymbol{X}_n)$, and $\Pi_{\mathbb{X}}(\boldsymbol{x})$ is a projection mapping $\boldsymbol{x}\notin\mathbb{X}$ back into $\mathbb{X}$. KW type SA constructs the gradient estimator $\hat{\nabla}h(\boldsymbol{X}_n)$ via finite difference, and in this paper we consider the following symmetric difference:

$$\hat{\nabla}h(\boldsymbol{X}_n) \coloneqq (h_1(\boldsymbol{X}_n),\ldots,h_d(\boldsymbol{X}_n))^{\intercal}, \tag{2}$$

where

$$h_i(\boldsymbol{X}_n) \coloneqq \frac{Y(\boldsymbol{X}_n + c_n\boldsymbol{e}_i) - Y(\boldsymbol{X}_n - c_n\boldsymbol{e}_i)}{2c_n},$$

$\boldsymbol{e}_i$ denotes a $d\times 1$ vector whose ith element is one and other elements are all zeros, $i = 1,\ldots,d$, and $\{c_n\}_{n\geq 1}$ is a deterministic positive sequence. It requires $2d$ simulation runs (samples) to compute $\hat{\nabla}h(\boldsymbol{X}_n)$. See more detailed discussion in Kushner and Yin (2003) or Chau and Fu (2015).

KW type SA is proved to be convergent almost surely (a.s.) to the unique local maximizer under certain conditions (Blum 1954), as formally stated in the following assumptions and lemma. Throughout this paper, for any function $f(\boldsymbol{x})$ defined on $\mathbb{X}$, we call a point $\boldsymbol{x}_0\in\mathbb{X}$ local maximizer if $f(\boldsymbol{x}_0)\geq f(\boldsymbol{x})$ for all $\boldsymbol{x}\in\mathbb{X}$ in an arbitrarily small neighborhood of $\boldsymbol{x}_0$.

Assumption 1 Suppose the following conditions all hold:

- (i) $h(\boldsymbol{x})$ has a unique local maximizer $\boldsymbol{\theta}$, which is interior to $\mathbb{X}$.
- (ii) $h(\boldsymbol{x})$ is continuous with continuous first and second derivatives.
- (iii) The second partial derivatives $\partial^2 h(\boldsymbol{x})/\partial x_i\partial x_j$ are bounded for $i,j = 1,\ldots,d$.
- (iv) For every positive number ε, there exists a positive number $\rho(\varepsilon)$ such that $\|\boldsymbol{x}-\boldsymbol{\theta}\|\geq\varepsilon$ implies $h(\boldsymbol{x})-h(\boldsymbol{\theta})\leq-\rho(\varepsilon)$ and $\|\nabla h(\boldsymbol{x})\|\geq\rho(\varepsilon)$, where $\|\cdot\|$ denotes the Euclidean distance.
- (v) $\sigma^2(\boldsymbol{x})\leq\sigma^2<\infty$.

Assumption 2 $\lim_{n\to\infty} c_n = 0$, $\sum_{n=1}^{\infty} a_n = \infty$, $\sum_{n=1}^{\infty} a_n c_n < \infty$, and $\sum_{n=1}^{\infty} a_n^2 c_n^{-2} < \infty$.

Lemma 1 (Blum (1954), theorem 3) Under Assumptions 1 and 2, the sequence $\{\boldsymbol{X}_n\}_{n\geq 1}$ defined in Eq. (1) converge a.s. to $\boldsymbol{\theta}$, i.e., $\boldsymbol{X}_n \xrightarrow{\text{a.s.}} \boldsymbol{\theta}$ as $n \to \infty$.

As pointed by Dupuis and Kushner (1989), such convergence will also hold if $\boldsymbol{\theta}$ is a boundary point. Moreover, when $\boldsymbol{\theta}$ is an interior point, the convergence rate of KW type SA is $O_p\left(n^{-1/3}\right)$ under certain conditions (Sacks 1958; Kushner and Clark 1978). Such rate of convergence is about the solution. Roughly speaking, if one views O_p approximately as O, then $\|\boldsymbol{X}_n - \boldsymbol{\theta}\|$ goes to zero at least as fast as $n^{-1/3}$. It is worth mentioning that the order $O_p\left(n^{-1/3}\right)$ is in terms of the iteration number rather than the sample size. In terms of the sample size, the order is $O_p\left((n/d)^{-1/3}\right)$. It means, roughly speaking, if for a one-dimensional problem we need sample size N to reach certain accuracy of solution, then we will need dN samples to reach the same accuracy for a d-dimensional problem. It is not so bad since the growth is linear to the dimensionality, which does not suffer from the so called curse of dimensionality. Moreover, when simultaneous perturbation stochastic approximation (SPSA) is used to estimate the gradient, instead of using the symmetric finite difference, only two samples are required in each iteration, while the optimal rate of convergence (in terms of iteration number) is still $O_p\left(n^{-1/3}\right)$ (Spall 1992). In such case, even in terms of the sample size, the rate of convergence is still $O_p\left(n^{-1/3}\right)$, which is totally independent of the dimensionality. It is worth emphasizing that while this paper considers the KW type SA as an example, our integrated algorithm proposed later works seamlessly with the RM type SA or SPSA.

2.2 GP Regression

Let f_{GP} be a GP with mean function $\mu : \mathbb{X} \to \mathbb{R}$ and covariance function $k : \mathbb{X} \times \mathbb{X} \to \mathbb{R}$. Let $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m\}$ be the collection of points at which simulation is run, and let $\boldsymbol{Y} = (Y(\boldsymbol{x}_1), \ldots, Y(\boldsymbol{x}_m))^\intercal$ denote the corresponding vector of noisy observations. Then, GP regression will predict $h(\boldsymbol{x})$ at any $\boldsymbol{x} \in \mathbb{X}$ by the conditional mean function of f_{GP} given $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m\}$ and $\boldsymbol{Y}$, which is,

$$S_m(\boldsymbol{x}) \coloneqq \mu(\boldsymbol{x}) + \boldsymbol{k}^\intercal (\boldsymbol{K} + \boldsymbol{\Sigma})^{-1} (\boldsymbol{Y} - \boldsymbol{\mu}), \tag{3}$$

where $\boldsymbol{\mu} \coloneqq (\mu(\boldsymbol{x}_1), \ldots, \mu(\boldsymbol{x}_m))^\intercal$, $\boldsymbol{k} \coloneqq (k(\boldsymbol{x}, \boldsymbol{x}_1), \ldots, k(\boldsymbol{x}, \boldsymbol{x}_m))^\intercal$, $\boldsymbol{K} \coloneqq (k(\boldsymbol{x}_i, \boldsymbol{x}_j))_{1 \leq i,j \leq m}$ is a $m \times m$ matrix, and $\boldsymbol{\Sigma} \coloneqq \text{diag}\{\sigma^2(\boldsymbol{x}_1), \ldots, \sigma^2(\boldsymbol{x}_m)\}$ a $m \times m$ diagonal matrix. In practice, μ is often chosen as a constant, and k is often chosen as the so called squared exponential covariance function (or Gaussian covariance function):

$$k(\boldsymbol{x}, \boldsymbol{y}) = \tau^2 \exp\left\{ -\sum_{i=1}^{d} \alpha_i (x_i - y_i)^2 \right\},$$

where $\left\{\tau^2, \alpha_1, \ldots, \alpha_d\right\}$ are hyper-parameters. We will also adopt such choice throughout this paper, in both theoretical analysis and numerical experiments. See more detailed discussion on GP regression in Rasmussen and Williams (2005).

GP regression has well established property on convergence when the variance of noise, $\sigma^2(\boldsymbol{x})$, is treated as known. Since this result serves as the foundation of asymptotic analysis of GP regression, we will also treat $\sigma^2(\boldsymbol{x})$ as known throughout this paper, while confessing the fact that it usually needs to be estimated in practice. The general convergence of GP regression is formally stated as follows.

Assumption 3 $Y(\boldsymbol{x})$ follows a normal distribution with mean $h(\boldsymbol{x})$ and variance $\sigma^2(\boldsymbol{x})$, for $\boldsymbol{x} \in \mathbb{X}$.

Lemma 2 (Bect et al. (2019), proposition 2.9) Under Assumption 3, $S_m(\boldsymbol{x})$ in Eq. (3) converges to a function, denoted as $S_\infty(\boldsymbol{x})$, uniformly in $\boldsymbol{x} \in \mathbb{X}$ a.s., as $n \to \infty$. That is,

$$\mathbb{P}\left\{ \lim_{m \to \infty} \max_{\boldsymbol{x} \in \mathbb{X}} |S_m(\boldsymbol{x}) - S_\infty(\boldsymbol{x})| = 0 \right\} = 1.$$

Note that Lemma 2 ensures that the regression surface $S_m(\boldsymbol{x})$ will converge to some function in the end (i.e., it is not divergent), but the limit is not necessarily the desired $h(\boldsymbol{x})$. However, as we will show

later, together with the condition that the design points are dense in $\mathbb{A} \subseteq \mathbb{X}$, we can obtain the almost sure uniform convergence of $S_m(\boldsymbol{x})$ to $h(\boldsymbol{x})$ in $\mathbb{A}$. This is critical to ensure that integrating GP regression to SA as a surrogate model will not ruin the convergence of SA.

3 ALGORITHM

We propose the SAwGP algorithm in this paper, which is an integration of the original SA and the GP regression. The complementary strength of GP regression is that it can use previously sampled points to predict response surface in less sampled region and identify promising area. Serving as a surrogate model, GP regression is intended to guide the search process of SA, especially in the early stage. We consider extremely simple integration mechanism of SA and GP: After every η iterations of SA, a surrogate model is built via GP regression, and then let SA start at the estimated best solution found from the surrogate model. The detailed steps will be described shortly. But it is worth mentioning that such integration does not ruin the convergence of SA, and the rate of convergence seems not only undamaged but also enhanced. Before stating the steps of SAwGP, several notations are required. For any $\delta > 0$ and $\boldsymbol{x} \in \mathbb{R}^d$, define $\mathscr{B}(\boldsymbol{x}, \delta) := \{\boldsymbol{y} : \|\boldsymbol{y} - \boldsymbol{x}\| \leq \delta\}$ as the closed ball centered at $\boldsymbol{y}$ with radius δ. Let $\mathscr{X}$ denote the collection of points at which simulation is run, and $\mathscr{Y}$ the collection of corresponding noisy observations.

3.1 SAwGP Algorithm

The proposed SAwGP algorithm consists of the following three main steps.

Step 0. Initialization: Specify a positive integer $\eta \geq 1$ as the period of constructing surrogate model, a small positive number δ, sequence of positive numbers $\{a_n\}_{n\geq 1}$ and $\{c_n\}_{n\geq 1}$ that satisfy Assumption 2. Set $n = 1$, $r = 1$, $\mathscr{X} = \emptyset$, $\mathscr{Y} = \emptyset$. Arbitrarily choose an initial point $\boldsymbol{X}_1 \in \mathbb{X}$.

Step 1. while $n \leq r\eta$ **do**

Output $\boldsymbol{X}_n$ as current best solution.

SA Iteration: Compute $\boldsymbol{X}_{n+1}$ via Eqs. (1) and (2), and update $\mathscr{X} = \mathscr{X} \cup \{\boldsymbol{X}_n, \boldsymbol{X}_n + c_n\boldsymbol{e}_1, \ldots, \boldsymbol{X}_n + c_n\boldsymbol{e}_d, \boldsymbol{X}_n - c_n\boldsymbol{e}_1, \ldots, \boldsymbol{X}_n - c_n\boldsymbol{e}_d\}$, $\mathscr{Y} = \mathscr{Y} \cup \{Y(\boldsymbol{X}_n), Y(\boldsymbol{X}_n + c_n\boldsymbol{e}_1), \ldots, Y(\boldsymbol{X}_n + c_n\boldsymbol{e}_d), Y(\boldsymbol{X}_n - c_n\boldsymbol{e}_1), \ldots, Y(\boldsymbol{X}_n - c_n\boldsymbol{e}_d)\}$.

Set $n = n + 1$.

Step 2. GP Regression: Let $m = (2d+1)r\eta + r - 1$.

Step 2.1. Build surrogate model: Compute $S_m(\boldsymbol{x})$ via Eq. (3) for $\boldsymbol{x} \in \mathbb{X}$, where $\boldsymbol{\mu}$, $\boldsymbol{k}$, $\boldsymbol{K}$, $\boldsymbol{\Sigma}$ and $\boldsymbol{Y}$ in Eq. (3) are all defined corresponding to $\mathscr{X}$ and $\mathscr{Y}$.

Step 2.2. Restart SA iteration: Let $\boldsymbol{X}_m^* := \arg\max_{\boldsymbol{x}\in\mathbb{X}} S_m(\boldsymbol{x})$. Set $\boldsymbol{X}_n = \boldsymbol{X}_m^*$.

Step 2.3. Random sampling: Uniformly and randomly sample a point $\boldsymbol{X}'$ in $\mathscr{B}(\boldsymbol{X}_m^*, \delta) \cap \mathbb{X}$, and update $\mathscr{X} = \mathscr{X} \cup \{\boldsymbol{X}'\}$, $\mathscr{Y} = \mathscr{Y} \cup \{Y(\boldsymbol{X}')\}$.

Set $r = r + 1$. **Go to Step 1.**

In SAwGP, each time when the GP Regression surrogate model S_m is built and the SA iteration is relocated at $\boldsymbol{X}_m^*$, we impose a purely random sampling in the neighborhood of $\boldsymbol{X}_m^*$. This is to ensure that in the limit, the sampled points around the the global maximizer of $S_\infty(\boldsymbol{x})$, say $\boldsymbol{X}_\infty^*$, is dense, which allows us to prove that $S_\infty(\boldsymbol{x})$ agrees with $h(\boldsymbol{x})$ around $\boldsymbol{X}_\infty^*$ and hence $\boldsymbol{X}_\infty^*$ must be one of the local maximizers of $h(\boldsymbol{x})$. Thus the SA iteration will not be misled by the surrogate model. Without the random sampling, the SA iteration alone can only ensure that $\boldsymbol{X}_\infty^*$ is an accumulation point. In such case, it remains a open question whether SAwGP is still locally convergent, as restricted by the theoretical tools available to justify the convergence of GP regression to $h(\boldsymbol{x})$. However, we conjecture that the answer is positive, and the reason is as follows. Although currently the local convergence of SAwGP is established via the convergence of GP regression, we believe that the primary force that drives the solution of SAwGP to a local maximizer of $h(\boldsymbol{x})$ actually comes from the SA iteration. This is supported by the numerical results that: 1) The convergence rate of SAwGP is no slower than SA; 2) the revised SAwGP that omits the random sampling

has similar asymptotic behavior. It is surely a challenging but interesting issue to be theoretically addressed in the future research.

3.2 Convergence of SAwGP

As emphasized before, the purpose of integrating a surrogate model into SA is to better guide the solution search before the sample size goes to infinity, but such surrogate model should not ruin the convergence of SA, and more ideally, the convergence rate of SA. We now present the local convergence of SAwGP with a brief sketch of proofs, while the rate of convergence is investigated by numerical experiments later. Let $\mathscr{M}$ denote the set of all local maximizers of $h(\boldsymbol{x})$, and for $\boldsymbol{x} \in \mathbb{X}$ define $d(\boldsymbol{x}, \mathscr{M}) := \min_{\boldsymbol{y} \in \mathscr{M}} \|\boldsymbol{x} - \boldsymbol{y}\|$. The local convergence of SAwGP is formally stated in the following theorem, with Assumption 1 for original SA being slightly relaxed to Assumption 4.

Assumption 4 Suppose the following conditions all hold:

(i) $h(\boldsymbol{x})$ is continuous with continuous first and second derivatives.
(ii) The second partial derivatives $\partial^2 h(\boldsymbol{x})/\partial x_i \partial x_j$ are bounded for $i, j = 1, \ldots, d$.
(iii) For any $\boldsymbol{x} \in \mathbb{X}$ and positive number ε, if $h(\boldsymbol{x}) = h(\boldsymbol{y})$ for all $\boldsymbol{y} \in \mathscr{B}(\boldsymbol{x}, \varepsilon) \cap \mathbb{X}$, then $\boldsymbol{x} \in \mathscr{M}$.
(iv) $\sigma^2(\boldsymbol{x}) \leq \sigma^2 < \infty$.

Theorem 1 Under Assumptions 2–4, the SAwGP algorithm is locally convergent a.s., i.e.,

$$\mathbb{P}\left\{\lim_{n\to\infty} d(\boldsymbol{X}_n, \mathscr{M}) = 0\right\} = 1,$$

where $\{\boldsymbol{X}_n\}_{n\geq 1}$ is the output of the SAwGP algorithm.

The local convergence of SAwGP is slightly more general than that of the original SA, since the latter requires that there is only one unique local maximizer and no local minimizer and no saddle point in $\mathbb{X}$, while the former allows multiple local maximizers, local minimizers and saddle points. Theorem 1 can be established based on two intermediate results, which are stated in the following two propositions. Note that $m = (2d+1)r\eta + r - 1$, as defined in the SAwGP algorithm, and $m \to \infty$ as $r \to \infty$.

Proposition 1 Suppose Assumptions 3 and 4 (i) hold. Let $S_\infty(\boldsymbol{x})$ denote the limit of $S_m(\boldsymbol{x})$ constructed in the SAwGP algorithm, as $r \to \infty$. Let $\boldsymbol{X}^*_\infty := \arg\max_{\boldsymbol{x}\in\mathbb{X}} S_\infty(\boldsymbol{x})$. For small positive δ specified in the SAwGP algorithm, define $\mathbb{A} := \mathscr{B}(\boldsymbol{X}^*_\infty, \delta) \cap \mathbb{X}$. Then, $S_m(\boldsymbol{x})$ converges to $h(\boldsymbol{x})$ uniformly in $\boldsymbol{x} \in \mathbb{A}$ a.s., as $r \to \infty$. That is,

$$\mathbb{P}\left\{\lim_{r\to\infty} \max_{\boldsymbol{x}\in\mathbb{A}} |S_m(\boldsymbol{x}) - h(\boldsymbol{x})| = 0\right\} = 1.$$

Proposition 1 says that the constructed GP regression surrogate model approaches to the objective function around $\boldsymbol{X}^*_\infty$. It critically relies on Lemma 2 that is about the uniform convergence of GP regression, and the fact that the sampled points are dense in $\mathbb{A}$, and the proof is a combination of the proofs of the lemmas 4 and 5 and theorem 2 in Ding et al. (2021). The second fact above is ensured by the random sampling step in the SAwGP algorithm. We conjecture that similar result may hold even without imposing the random sampling (maybe with some other assumptions), but this question remains open.

Proposition 2 Under Assumptions 3 and 4 (i) (iii),

$$\mathbb{P}\left\{\lim_{r\to\infty} d(\boldsymbol{X}^*_m, \mathscr{M}) = 0\right\} = 1,$$

where $\{\boldsymbol{X}^*_m\}_{r\geq 1}$ is defined in the SAwGP algorithm.

Proposition 2 says that as the sample size goes to infinity, the global maximizer of the GP regression surrogate model converges to one point in $\mathscr{M}$ a.s.. This ensures that, restarting the SA iteration at the global maximizer of the surrogate model every η iterations is indeed not misleading in the limit. Proposition 2 can be proved by contradiction, with the two facts stated in Assumption 4 (iii) and Proposition 1, respectively.

With Proposition 2, it suffices to show that, when restarting from one local maximizer of $h(\boldsymbol{x})$ repeatedly, the SA will converge to that local maximizer. But this is quite clear intuitively, since originally SA is locally convergent and restarting from the local maximizer will not make things worse. We mentioned before that we conjecture SAwGP is still locally convergent without the random sampling step. There are two possible ways to achieve that. The first possible way is to prove that $S_m(\boldsymbol{x})$ converges to $h(\boldsymbol{x})$ uniformly in a neighborhood of $\boldsymbol{X}_\infty^*$ without requiring the sampled points are dense in that neighborhood, then the subsequent analysis remains the same. The second possible way is to directly analyze the sequence $\{\boldsymbol{X}_n\}_{n\geq 1}$ and show that restarting from $\boldsymbol{X}_m^*$ repeatedly does not essentially affect the sequence in the limit. This way is more fundamental but it is necessary if one wants to also analyze the convergence rate of SAwGP. All these issues are worth investigating in the future research.

In practice, solving $\boldsymbol{X}_m^* = \arg\max_{\boldsymbol{x}\in\mathbb{X}} S_m(\boldsymbol{x})$ in SAwGP can be a challenging problem itself, since $S_m(\boldsymbol{x})$ is a continuous function with multiple local maximizers. Solvers may usually return a local maximizer of $S_m(\boldsymbol{x})$, rather than $\boldsymbol{X}_m^*$. However, as long as the returned local maximizer converges (e.g., one always take $\arg\max_{\boldsymbol{x}\in\mathscr{X}} S_m(\boldsymbol{x})$ as the initial value for the solver), the local convergence of SAwGP is stilled guaranteed, although the finally solution obtained by SAwGP may be not as good as the one given $\boldsymbol{X}_m^*$ is always found.

4 NUMERICAL EXPERIMENTS

In this section we investigate the performance of SAwGP numerically. By comparing with the original SA, we demonstrate the finite-sample performance of SAwGP and verify whether SAwGP approaches to the true optimum (like SA or even better than SA).

4.1 A Simple Problem

We first consider a simple problem as follows:

$$h(\boldsymbol{x}) = -0.05\|\boldsymbol{x}-\mathbf{2}\|^2 + 10, \quad \boldsymbol{x}\in\mathbb{X} := [-10,10]^d,$$

where $\mathbf{2}$ denotes a $d\times 1$ vector with all 2's. And clearly, $\boldsymbol{\theta} = \mathbf{2}$ is the unique local maximizer, and the optimal function value is 10. Normally distributed observation noise with mean 0 and variance 1 is added. For both SA and SAwGP, set $a_n = n^{-1}$ and $c_n = n^{-1/6}$, and the iteration points out of $\mathbb{X}$ are projected back by truncating the exceeding coordinates at the boundaries. For SAwGP, set $\eta = 20$, $\mu = 0$, $\tau^2 = 10^2$ and $\alpha_i = 50/d$, $i = 1,\ldots,d$, for GP regression, and $\delta = 1$. The maximization of $S_m(\boldsymbol{x})$ is implemented by the `minimize` function (with method 'L-BFGS-B') in SciPy `optimize`. We let $d = 1,2,3$, respectively. For each d, both SA and SAwGP are repeated independently for 30 times, in which the initial solution is always chosen as $\boldsymbol{X}_1 = (-8,\ldots,-8)^\intercal$ with compatible dimension, and the average performance together with the standard deviation of the average are calculated.

Note that in order to output $\boldsymbol{X}_n$ for a given n, the numbers of samples (i.e., noisy observations) consumed in the original SA algorithm and the SAwGP algorithm are different. For the original SA algorithm the consumed sample size is $2d(n-1)$, while for the SAwGP algorithm it is $(2d+1)(n-1)+\lfloor (n-1)/\eta \rfloor - 1$, where $\lfloor\cdot\rfloor$ is the floor function. So, for fair comparison between SA and SAwGP, we need to ensure that the same number of samples are consumed, rather than the same number of SA iterations. For each d, SAwGP algorithm is run until 1000 SA iterations are completed, and the original SA algorithm is run until the same number of samples are used. We also consider a revised (simplified) version of SAwGP, by omitting the purely random sampling step, and call it SAwGP-simple.

Figure 1 shows the achieved function value $h(\boldsymbol{X}_n)$ versus consumed sample size, for SAwGP, SAwGP-simple and SA. It can be clearly seen that the finite-sample performance of SAwGP outperforms SA for each d, which reveals the benefit by integrating GP regress as surrogate model to guide the SA iteration. Also, from the case of $d = 1$, the asymptotic convergence of SAwGP is observed. For $d = 2,3$, more samples are required to observe the convergence, since $h(\boldsymbol{x})$ is quite flat around $\boldsymbol{\theta}$, making the problem difficult. We can also see that the performance of SAwGP-simple is similar to SAwGP, which suggests that the random sampling does not make much difference.

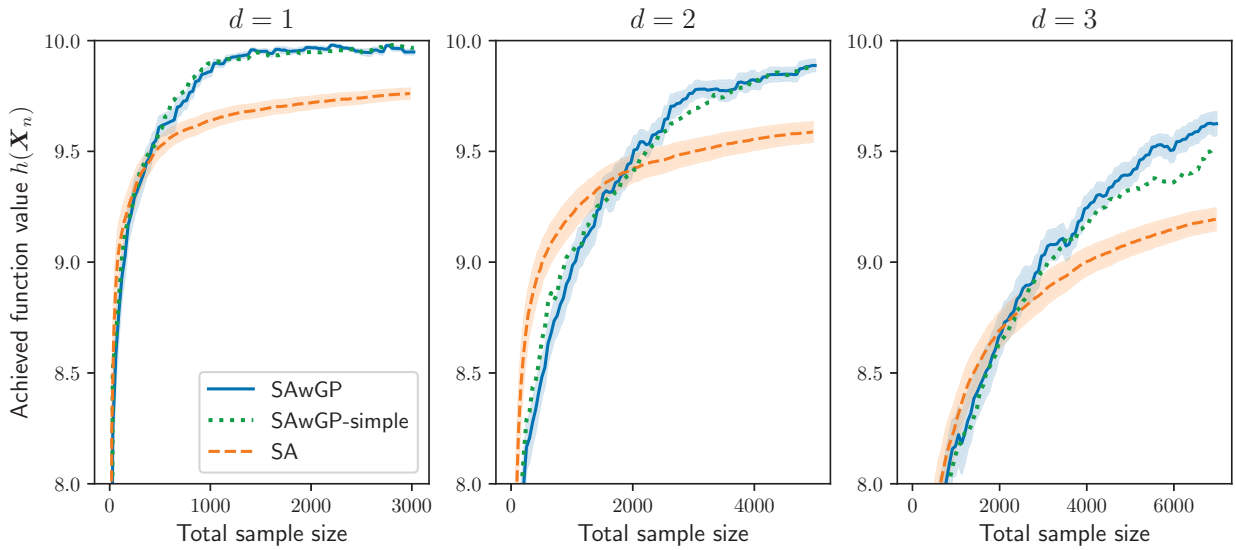


Figure 1: Achieved function value $h(\boldsymbol{X}_n)$ versus consumed sample size. Lines are the average of 30 independent repetitions, and the shaded regions represent ± 1 standard deviation of the average.

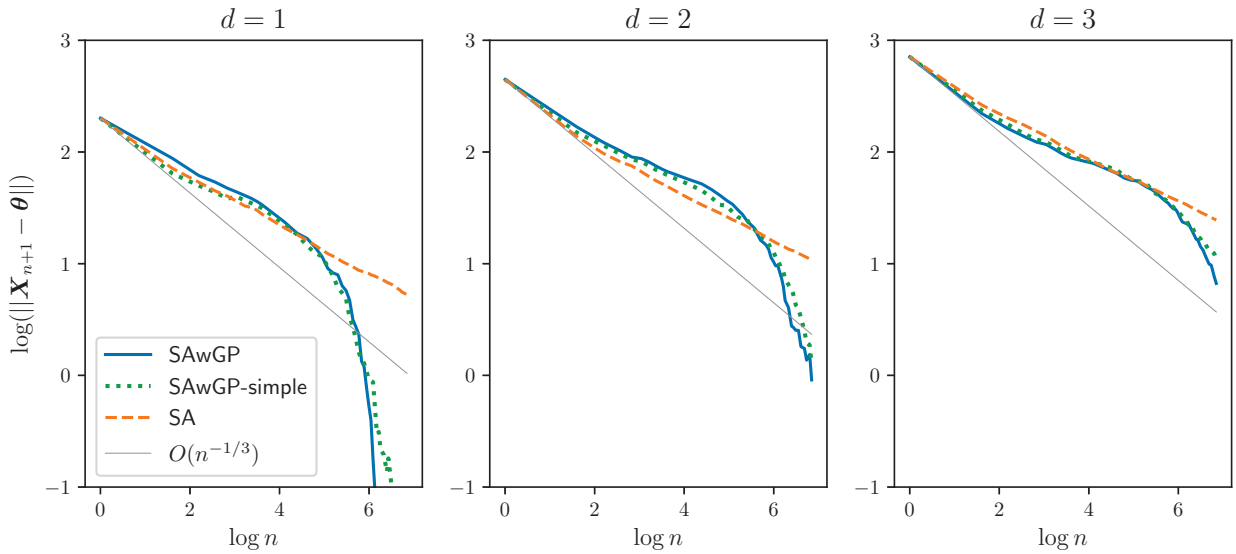


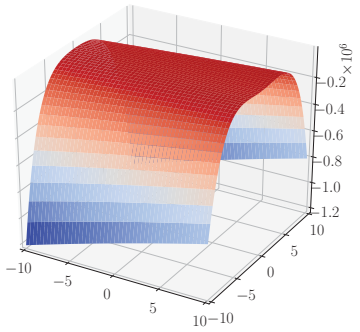
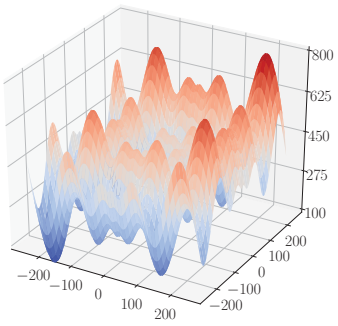
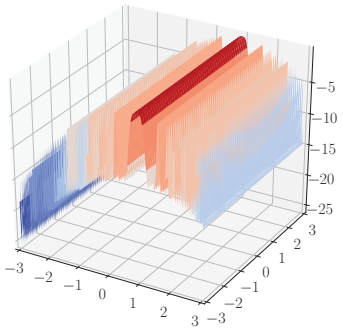
Figure 2: Convergence rate of $\boldsymbol{X}_n$ in terms of the iteration number: $\log(\|\boldsymbol{X}_{n+1} - \boldsymbol{\theta}\|)$ versus $\log n$. Lines are the average of 30 independent repetitions.

Figure 2 shows the convergence rate of $\boldsymbol{X}_n$ in terms of the iteration number. Specifically, the rate at which $\|\boldsymbol{X}_{n+1} - \boldsymbol{\theta}\|$ goes to zero as iteration number n increases is presented to approximately indicate the convergence rate of $\boldsymbol{X}_n$ to $\boldsymbol{\theta}$ (which, strictly speaking, is defined in probability). As mentioned, different algorithms consume different sample sizes in one iteration, but this will make no difference to the slope of the line after taking logarithm (to base e) of both sides. The gray thin line represents the rate $O(n^{-1/3})$, which is (roughly speaking) the theoretical convergence rate of SA. However, we can see that the exhibited rate of SA up to 1000 iterations is slower than that, because the iteration is far from large enough for SA in this problem (as also suggested by Figure 1). On the other hand, SAwGP exhibits mush faster rate within the same number of iterations. It reveals that in this problem the convergence rate of SA does not deteriorate by integrating the surrogate model, and the rate is even further enhanced. Besides, we also see that the performance of SAwGP-simple is similar to SAwGP, which again suggests that the random sampling does not make much difference.

4.2 Well Known Problems

We also consider some well known problems investigated in Fan and Hu (2018), with the minimization tasks converting to maximization tasks. They are described in details in Table 1, together with the values of all hyper-parameters. The results about the convergence of the achieved function value $h(\boldsymbol{X}_n)$ are shown in Figure 3, and the results about the convergence rate of $\boldsymbol{X}_n$ are shown in Figure 4. They are similar to those in the simple problem, thus not repeatedly described here.

Table 1: Details of the tested functions and hyper-parameters.

	Rosenbrock function ($d = 3$)	Schwefel function ($d = 2$)	Trigonometric function ($d = 2$)
$h(\boldsymbol{x})$	$-\sum_{i=2}^{d}\left[100(x_{i-1}-x_i^2)^2 + (x_i-1)^2\right]$	$-201.8432d + \sum_{i=1}^{d} x_i \sin\sqrt{\lvert x_i\rvert}$	$-\sum_{i=1}^{d}\left[8\sin^2(7(x_i-0.9)^2) + 6\sin^2(14(x_i-0.9)^2) + (x_i-0.9)^2\right]$
$\mathbb{X}$	$[-10,10]^d$	$[-250,250]^d$	$[-3,3]^d$
Plot			
Obs. noise	$N(0,10^2)$	$N(0,10^2)$	$N(0,5^2)$
$\boldsymbol{X}_1$	$(-8,-8,-8)^\intercal$	$(50,50)^\intercal$	$(-2,-2)^\intercal$
a_n	$0.01 \times n^{-1}$	n^{-1}	n^{-1}
c_n	$0.001 \times n^{-1/6}$	$n^{-1/6}$	$n^{-1/6}$
μ	-5×10^5	-400	-15
α_i	10	10	10

Note: Hyper-parameters not listed in this table are kept the same as in the simple problem.

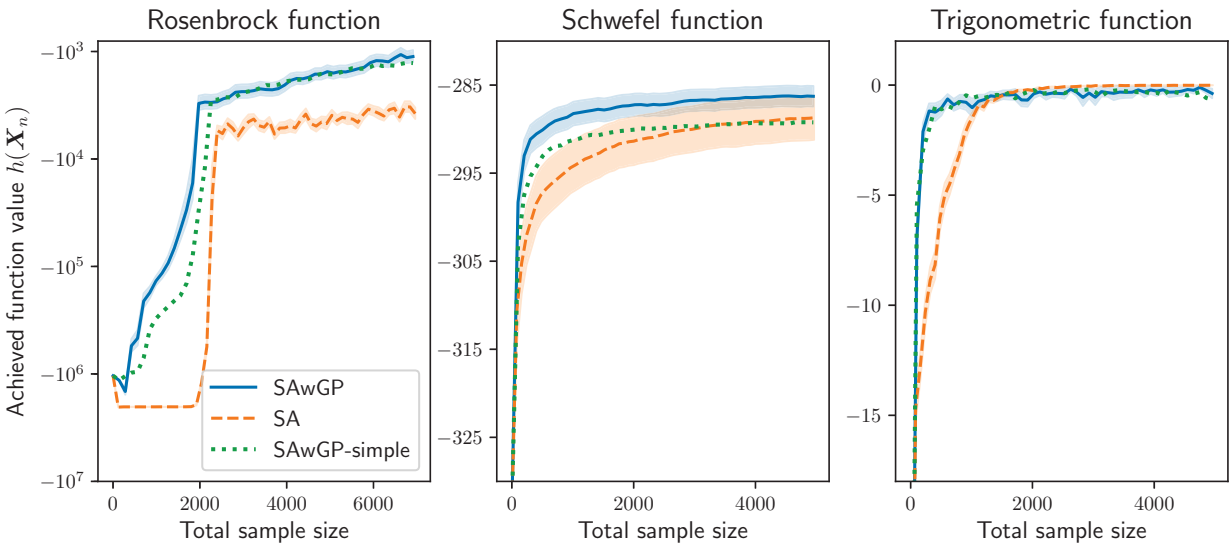


Figure 3: Achieved function value $h(\boldsymbol{X}_n)$ versus consumed sample size. Lines are the average of 100 independent repetitions, and the shaded regions represent ± 1 standard deviation of the average.

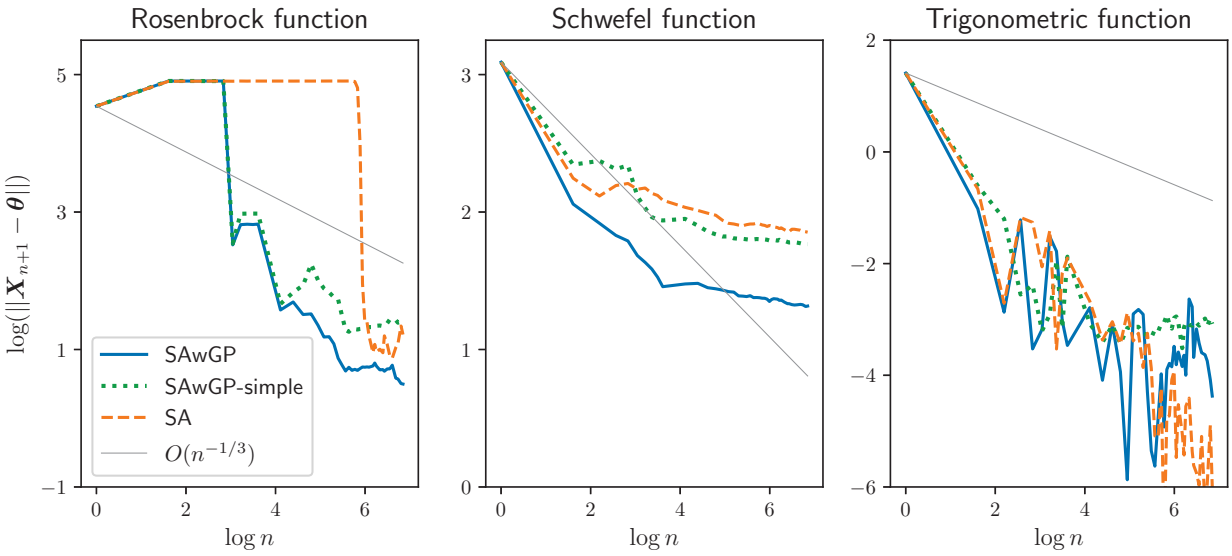


Figure 4: Convergence rate of $\boldsymbol{X}_n$ in terms of the iteration number: $\log(\|\boldsymbol{X}_{n+1} - \boldsymbol{\theta}\|)$ versus $\log n$, where $\boldsymbol{\theta}$ denotes the local maximizer that is closet to $\boldsymbol{X}_n$ when an algorithm stops. Lines are the average of 100 independent repetitions.

5 CONCLUSIONS

This paper proposes to integrate GP regression into SA as surrogate model to guide the search of SA. The local convergence is proved, and the better finite-sample performance and rate of convergence are demonstrated via a simple problem. There are quite a few interesting directions for future research. One is, as mention before, to theoretically analyze if SAwGP is still locally convergent without the random sampling step (i.e., SAwGP-simple). Another direction is to theoretically establish the convergence rate of SAwGP or SAwGP-simple. These two directions are both quite challenging but important. Relatively applied future work includes more extensive numerical studies on the performance of SAwGP and SAwGP-simple.

ACKNOWLEDGMENTS

Haihui Shen and Zhibin Jiang are corresponding authors of the paper. This work is supported by the National Key Research and Development Program of China [Grant No. 2018AAA0101700, Task 5] and National Natural Science Foundation of China [Grant Nos. 72001140 and 72031006].

REFERENCES

Adler, R. J., and J. E. Taylor. *Random Fields and Geometry*. New York: Springer Verlag.

Andradóttir, S. 1996. "A scaled stochastic approximation algorithm". *Management Science* 42(4):475–498.

Ankenman, B., B. L. Nelson, and J. Staum. 2010. "Stochastic kriging for simulation metamodeling". *Operations Research* 58(2):371–382.

Bect, J., F. Bachoc, and D. Ginsbourger. 2019. "A supermartingale approach to Gaussian process based sequential design of experiments". *Bernoulli* 25(4A):2883–2919.

Blum, J. R. 1954. "Multidimensional stochastic approximation methods". *The Annals of Mathematical Statistics* 25(4):737 – 744.

Bottou, L. 2010. "Large-scale machine learning with stochastic gradient descent". In *Proceedings of the 19th International Conference on Computational Statistics*, edited by Y. Lechevallier and G. Saporta, 177–186. Paris, France: Physica Verlag HD.

Bottou, L. 2012. "Stochastic gradient descent tricks". In *Neural Networks: Tricks of the Trade* (2nd ed.)., edited by G. Montavon, G. B. Orr, and K.-R. Müller, 421–436. Berlin, Heidelberg: Springer Verlag.

Chang, K.-H., L. J. Hong, and H. Wan. 2013. "Stochastic trust-region response-surface method (STRONG)–A new response-surface framework for simulation optimization". *INFORMS Journal on Computing* 25(2):230–243.

Chau, M., and M. C. Fu. 2015. "An overview of stochastic approximation". In *Handbook of Simulation Optimization*, edited by M. C. Fu, 149–178. New York: Springer Verlag.

Chen, H. F. 2006. *Stochastic Approximation and Its Applications*, Volume 64. Boston, MA: Springer Verlag.

Ding, L., L. J. Hong, H. Shen, and X. Zhang. 2021. "Knowledge gradient for selection with covariates: Consistency and computation". *arXiv:1906.05098v5*.

Dupuis, P., and H. J. Kushner. 1989. "Stochastic approximation and large deviations: Upper bounds and w.p.1 convergence". *SIAM Journal on Control and Optimization* 27(5):1108–1135.

Fan, Q., and J. Hu. 2018. "Surrogate-based promising area search for Lipschitz continuous simulation optimization". *INFORMS Journal on Computing* 30(4):677–693.

Fu, M. C., and Y. C. Ho. 1988. "Using perturbation analysis for gradient estimation, averaging and updating in a stochastic approximation algorithm". In *Proceedings of the 1988 Winter Simulation Conference*, edited by M. A. Abrams, P. L. Haigh, and J. C. Comfort, 509–517. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Hong, L. J., and B. L. Nelson. 2006. "Discrete optimization via simulation using COMPASS". *Operations Research* 54(1):115–129.

Kiefer, J., and J. Wolfowitz. 1952. "Stochastic estimation of the maximum of a regression function". *The Annals of Mathematical Statistics* 23(3):462–466.

Kleijnen, J. P. C. 2015. "Response surface methodology". In *Handbook of Simulation Optimization*, edited by M. C. Fu, 81–104. New York: Springer Verlag.

Kushner, H. J., and D. S. Clark. 1978. *Stochastic Approximation Methods for Constrained and Unconstrained Systems*. New York: Springer Verlag.

Kushner, H. J., and G. G. Yin. 2003. *Stochastic Approximation and Recursive Algorithms and Applications*. 2nd ed. New York: Springer Verlag.

Nemirovski, A., A. Juditsky, G. Lan, and A. Shapiro. 2009. "Robust stochastic approximation approach to stochastic programming". *SIAM Journal on Optimization* 19(4):1574–1609.

Newton, D., F. Yousefian, and R. Pasupathy. 2018. "Stochastic gradient descent: Recent trends". In *TutORials in Operations Research*, edited by E. Gel and D. Lewis, Volume 7, 193–220. Phoenix, Arizona: Institute for Operations Research and the Management Sciences.

Rasmussen, C. E., and C. K. I. Williams. 2005. *Gaussian Processes for Machine Learning*. Cambridge, MA: The MIT Press.

Robbins, H., and S. Monro. 1951. "A stochastic approximation method". *The Annals of Mathematical Statistics* 22(3):400–407.

Sacks, J. 1958. "Asymptotic distribution of stochastic approximation procedures". *The Annals of Mathematical Statistics* 29(2):373–405.

Spall, J. C. 1992. "Multivariate stochastic approximation using a simultaneous perturbation gradient approximation". *IEEE Transactions on Automatic Control* 37(3):332–341.

Sun, L., L. J. Hong, and Z. Hu. 2014. "Balancing exploitation and exploration in discrete optimization via simulation through a Gaussian process-based search". *Operations Research* 62(6):1416–1438.

Sun, W., Z. Hu, and L. J. Hong. 2018. "Gaussian mixture model-based random search for continuous optimization via simulation". In *Proceedings of the 2018 Winter Simulation Conference*, edited by M. Rabe, A. A. Juan, N. Mustafee, A. Skoogh, S. Jain, and B. Johansson, 2003–2014. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

AUTHOR BIOGRAPHIES

YINGCUI YAN is a master student in the Sino-US Global Logistics Institute, Antai College of Economics and Management at Shanghai Jiao Tong University. Her research interests include continuous simulation optimization and its application in smart factory and unmanned production line. Her email address is yan_yingcui@sjtu.edu.cn.

HAIHUI SHEN is an assistant professor in the Sino-US Global Logistics Institute, Antai College of Economics and Management at Shanghai Jiao Tong University. He earned his Ph.D. in the Department of Management Sciences at City University of Hong Kong. His research interests include ranking and selection, simulation optimization, Gaussian process and their application. His email address is shenhaihui@sjtu.edu.cn. His website is https://shenhaihui.github.io.

ZHIBIN JIANG is a distinguished professor in the Department of Management Science and Sino-US Global Logistics Institute, Antai College of Economics and Management at Shanghai Jiao Tong University. He is the dean of the Sino-US Global Logistics Institute. He is a fellow of Institute of Industrial and Systems Engineers (IISE) and associate editor of *International Journal of Production Research*. His research interests include production and service operations management, and healthcare management. His email address is zbjiang@sjtu.edu.cn.