https://pubsonline.informs.org/journal/opre

OPERATIONS RESEARCH
Vol. 73, No. 1, January–February 2025, pp. 385–407
ISSN 0030-364X (print), ISSN 1526-5463 (online)

Methods

Gaussian Process-Based Random Search for Continuous Optimization via Simulation

Xiuxian Wang,[a] L. Jeff Hong,[b,*] Zhibin Jiang,[a,c] Haihui Shen[a]

[a] Sino-US Global Logistics Institute, Antai College of Economics and Management, Shanghai Jiao Tong University, Shanghai 200030, China; [b] School of Management and School of Data Science, Fudan University, Shanghai 200433, China; [c] Antai College of Economics and Management, Shanghai Jiao Tong University, Shanghai 200030, China

*Corresponding author

Contact: wang_xx@sjtu.edu.cn (XW); hong_liu@fudan.edu.cn, https://orcid.org/0000-0001-7011-4001 (LJH); zbjiang@sjtu.edu.cn (ZJ); shenhaihui@sjtu.edu.cn, https://orcid.org/0000-0002-4157-1278 (HS)

Received: May 14, 2021
Revised: October 1, 2021; September 8, 2022; May 23, 2023
Accepted: June 12, 2023
Published Online in Articles in Advance: August 1, 2023

Area of Review: Optimization

https://doi.org/10.1287/opre.2021.0303

Abstract. Random search is an important category of algorithms to solve continuous optimization via simulation problems. To design an efficient random search algorithm, the handling of the triple "E" (i.e., exploration, exploitation and estimation) is critical. The first two E's refer to the design of sampling distribution to balance explorative and exploitative searches, whereas the third E refers to the estimation of objective function values based on noisy simulation observations. In this paper, we propose a class of Gaussian process-based random search (GPRS) algorithms, which provide a new framework to handle the triple "E." In each iteration, algorithms under the framework build a Gaussian process surrogate model to estimate the objective function based on single observation of each sampled solution and randomly sample solutions from a lower-bounded sampling distribution. Under the assumption of heteroscedastic and known simulation noise, we prove the global convergence of GPRS algorithms. Moreover, for Gaussian processes having continuously differentiable sample paths, we show that the rate of convergence of GPRS algorithms can be no slower than $O_p(n^{-1/(d+2)})$. Then, we introduce a specific GPRS algorithm to show how to design an integrated GPRS algorithm with adaptive sampling distributions and how to implement the algorithm efficiently. Numerical experiments show that the algorithm has good performances, even for problems where the variances of simulation noises are unknown.

Funding: This work was supported by the National Natural Science Foundation of China [Grants 72031007, 72091211, 71931007].
Supplemental Material: The e-companion is available at https://doi.org/10.1287/opre.2021.0303.

Keywords: continuous optimization via simulation • random search algorithms • Gaussian process regression • convergence • rate of convergence

1. Introduction

Stochastic simulation is an important modeling tool for complex systems. It is widely used in the area of operations research and management science to model and to optimize the performances of supply chain networks, healthcare systems, transportation systems, etc. This approach of stochastic optimization is often called optimization via simulation (OvS), where decision variables are typically the design parameters of the simulation models. When the decision variables are continuous, the problem is known as a continuous optimization via simulation (COvS) problem.

Examples of COvS problems include inventory-level optimization to minimize the total expected production cost, appointment time optimization to minimize the total expected patient waiting time, traffic signal optimization to optimize the throughput of a transportation hub, and many others. Readers may refer to Amaran et al. (2016) for a comprehensive introduction to COvS and the related algorithms. Recently, parameter tuning is gaining a lot of research interest, especially in the area of machine learning where complicated stochastic black-box models need to be tuned. It is interesting to note that many of these problems may be viewed as COvS problems as well, where the stochastic black-box model and a call to the model may be treated as a stochastic simulation model and an experiment of the model, respectively. Readers may refer to Yu and Zhu (2020) for a comprehensive introduction to parameter tuning and the related algorithms.

Many types of algorithms have been proposed to solve COvS problems, including stochastic approximation

algorithms (Robbins and Monro 1951, Kiefer and Wolfowitz 1952, Spall 1992, Kushner and Yin 1997), response surface methodologies (Box and Wilson 1951, Kleijnen 1998, Chang et al. 2013), and random search algorithms (Andradóttir 2006, 2015). Different types of algorithms offer different types of convergence guarantees and are applicable to different settings of COvS problems. In this paper, we focus on random search algorithms, which typically do not require gradient information, have global convergence, and work for a wide range of COvS problems.

The key to designing efficient random search algorithms is the handling of the "triple E" (i.e., exploration, exploitation, and estimation) (Andradóttir and Prudius 2009). The first two E's focus on the designing of sampling distributions used in the algorithm iterations to place the search effort so that it can balance global and local searches, also known as explorative and exploitative searches. The third E focuses on the estimation of objective values using noisy simulation outputs. Next, we briefly review the literature on random search COvS algorithms along these two lines (i.e., designing of sampling distributions and estimation of objective values) and position our work relative to the literature.

In terms of sampling distributions, Sun et al. (2014) divide random search discrete optimization via simulation (DOvS) algorithms into four classes (exploration based, exploitation based, combined, and integrated) based on their approaches to handle the exploration and exploitation trade-off. Their classification is also applicable to random search COvS algorithms. *Exploration-based algorithms* include the simple random search algorithm of Chia and Glynn (2013) and the grid search algorithms of Ensor and Glynn (1997) and Yakowitz et al. (2000), which represent the feasible region by a set of either randomly generated solutions or equally spaced grid points and evaluate all of them. *Exploitation-based algorithms* include the surrogate-based promising area search algorithm of Fan and Hu (2018), which samples only from the most promising area in each iteration. Based on Sun et al. (2014, p. 1417), "*combined algorithms* typically focus on exploitative search while either adding a fixed amount of effort in each iteration or assigning a fixed sequence of iterations to conduct explorative search," and "*integrated algorithms* typically have an integrated sampling distribution governing the search effort in each iteration instead of separating the exploitation and exploration as in the combined algorithms." The adaptive sampling and resampling (ASR) algorithm of Andradóttir and Prudius (2010) is an example of the combined algorithms. It samples from the feasible region in each iteration from a predetermined sampling distribution and adds resampling from some of the previously visited solutions. The model reference adaptive search algorithm of Hu et al. (2007) and the Gaussian mixture model-based random search of Sun et al. (2018) are both examples of integrated algorithms. In each iteration, they both build surrogate models based on the simulation observations collected through the iteration and construct a sampling distribution to guide the random search. The use of surrogate models in guiding search is also very common in Bayesian optimization algorithms, and the Gaussian process surrogate model (also known as kriging) is very popular. There are kriging-based Bayesian optimization algorithms for both deterministic and noisy problems, including the P algorithm of Calvin and Žilinskas (1999), the efficient global optimization (EGO) algorithm of Jones et al. (1998), the sequential kriging optimization (SKO) algorithm of Huang et al. (2006), the knowledge gradient for continuous parameters (KGCP) algorithm of Scott et al. (2011), etc. These algorithms use a fixed sampling criterion (e.g., the expected improvement) to sample a solution deterministically in each iteration based on the surrogate model and hence, adopt totally different sampling strategies from the random search-based algorithms considered in this paper. Interested readers may refer to Picheny et al. (2013) for more information on sampling criteria of the kriging-based Bayesian optimization algorithms.

In terms of the estimation of objective values, there are in general two different approaches: the multiobservation approach and the single-observation approach. The multiobservation approach estimates the objective value based on repeatedly sampling the same solution and builds the convergence based on the strong law of large numbers. For instance, the simple random search algorithms of Chia and Glynn (2013) and the grid search algorithms of Ensor and Glynn (1997) and Yakowitz et al. (2000) all use the multiobservation approach. The single-observation approach samples each solution only once but relies on the samples from other solutions to ensure the convergence to the true objective value. To the best of our knowledge, this approach dated back to Devroye (1978), who uses a k-nearest neighbor (KNN) scheme to estimate the objective value of any feasible solution. In recent years, the single-observation approach has become more popular because of its superior empirical performance, and it is typically implemented through a shrinking-ball mechanism that is very similar to the KNN scheme (Baumert and Smith 2002). For instance, the algorithms of Andradóttir and Prudius (2010) and Fan and Hu (2018) all use the shrinking-ball mechanism. Kiatsupaibul et al. (2018) propose a general framework of random search algorithms with the shrinking-ball mechanism and provide conditions under which the algorithms are convergent. Recently, Zhang and Hu (2022) integrate the shrinking-ball mechanism with the model-based annealing random search algorithm of Hu and Hu (2011), and they propose a new single-observation COvS algorithm with adaptive random search. In the algorithm, the

authors build surrogate models based on the shrinking-ball method for constructing sampling distributions and prove a finite-time probability bound on the algorithm's performance under a Lipschitz continuity condition.

In this paper, we propose a class of *Gaussian process-based random search* (*GPRS*) *algorithms*, which provide a new framework to handle the triple "E" for random search COvS algorithms. Algorithms under the GPRS framework have two key components. One is that GPRS algorithms build a Gaussian process surrogate model to estimate the objective function value for every feasible solution in each iteration. The single-replication approach is adopted by GPRS algorithms, which require only one observation from each solution to construct the surrogate model. Another component is that GPRS algorithms sample solutions randomly according to a sampling distribution constructed in each iteration. A wide range of sampling distributions, including adaptive sampling distributions, are allowed in the GPRS framework as long as their sampling densities are lower bounded. Therefore, the combination of these two components enables the design of new *single-observation integrated COvS algorithms*. However, this framework also brings a theoretical challenge in the convergence analysis because the Gaussian process surrogate model is quite different from the shrinking-ball mechanism and the random search strategy is also different from the sampling criteria that are used in Bayesian optimization algorithms. In this paper, we establish the convergence results by exploring the properties of Gaussian process regression, and we prove that the surrogate model converges uniformly to the objective function if the sampling distributions are lower bounded.

Moreover, we also prove an upper bound of the rate of convergence of GPRS algorithms. Although there are many globally convergent random search algorithms for COvS problems, very few of them have rate of convergence results. Indeed, only exploration-based algorithms, such as the simple random search algorithm of Chia and Glynn (2013) and the grid search algorithms of Ensor and Glynn (1997) and Yakowitz et al. (2000), have known rate of convergence. This is because these algorithms have a very simple structure (e.g., determining the set of candidate solutions at the beginning and allocating the same number of observations for all candidate solutions) and are in general easy to analyze. When the sampling distributions are more complicated, rate of convergence results are in general very difficult to establish. In the random search algorithm of Zhang and Hu (2022), to establish the finite-time probability bound, the authors analyze the rate of convergence of the shrinking-ball method on a subset of sampled points under a Lipschitz continuity condition. However, there is still a gap between the rate of convergence of point estimation and the rate of convergence of the shrinking-ball algorithm. In this paper, we prove that if the sample paths of the imposed Gaussian process are continuously differentiable, the rate of convergence of GPRS algorithms can be no slower than $\tilde{O}_p(n^{-1/(d+2)})$, where d is the dimension of the decision variables and $\tilde{O}_p(\cdot)$ is a big O notation ignoring logarithmic factors. Recall that the upper bound of rate of convergence of the EGO algorithm for deterministic continuous black-box optimization problems is $O_p(n^{-1/d})$, when the functions are in the reproducing-kernel Hilbert space of the Gaussian process. Our proved bound is slightly slower possibly because of simulation noises. In the one-dimensional COvS problem, this upper bound may be considered to be tight because the optimal rate of convergence of nonparametric regression is $O_p(n^{-1/3})$ under similar smoothness conditions (Donoho 1994).

Lastly, we propose a specific GPRS algorithm as an example to describe how to design an integrated GPRS algorithm and how to implement the algorithm. This algorithm is an extension of the Gaussian process-based search (GPS) algorithm of Sun et al. (2014) from DOvS to COvS, and therefore, we call it the GPS-C algorithm. By using the Gaussian process surrogate model (i.e., the conditional mean and its uncertainty) both for estimating the objective function value and for constructing adaptive sampling distributions, the GPS-C algorithm integrates exploration, exploitation, and estimation seamlessly. Numerical experiments show that the GPS-C algorithm performs well for solving COvS problems with both known and unknown variances of simulation noises.

To summarize, this paper contributes to existing literature on COvS in a few aspects. First, it establishes the global convergence of GPRS algorithms without the explicit use of the shrinking-ball mechanism. Second, it establishes an upper bound of the rate of convergence for the whole class of GPRS algorithms that (1) use a Gaussian process surrogate model to estimate the objective function and (2) randomly sample solutions from a sequence of density functions bounded from below. Additionally, some intermediate results and techniques used in establishing the convergence and rate of convergence may be extendable to other applications of Gaussian process regression.

There are also some drawbacks in the methods used in establishing the convergence and rate of convergence results of GPRS algorithms. The results depend critically on two assumptions (i.e., the objective function is a sample path from a Gaussian process, and the simulation noises follow normal distributions with known variances). We admit these assumptions are restrictive. However, they are needed because we rely heavily on the properties of Gaussian process regression, which only work under these assumptions at this moment, and they are commonly used in Gaussian process-based optimization algorithms (e.g., the KGCP algorithm

of Scott et al. 2011). In the numerical study, we relax these assumptions and find that the GPS-C algorithm is robust and has good empirical performance even when the assumptions are not satisfied.

The rest of this paper is organized as follows. In Section 2, we describe the problem setting and introduce the GPRS framework. In Section 3, we analyze algorithms under the GPRS framework and show its global convergence. In Section 4, the rate of convergence is established for the Gaussian process having continuously differentiable sample paths. Section 5 uses the GPS-C algorithm as an example to describe the design and implementation of a GPRS algorithm. Illustrative numerical experiments are presented in Section 6. We conclude in Section 7 and include some technical proofs and numerical results in the e-companion.

2. Gaussian Process-Based Random Search Framework

We are interested in solving the COvS problems with the following form

$$\max_{x\in\mathcal{X}} \mathbb{E}[G(\boldsymbol{x};\omega)], \tag{1}$$

where $\boldsymbol{x}$ is a vector of continuous decision variables and ω represents the randomness of simulation experiments, and the expectation is taken with respect to ω. Let $g(\boldsymbol{x}) = \mathbb{E}[G(\boldsymbol{x};\omega)]$, which is a continuous function on $\mathcal{X}$. The functional form of $g(\boldsymbol{x})$ is unknown to us and can only be evaluated via noisy simulation observation $G(\boldsymbol{x};\omega)$, which is denoted as $G(\boldsymbol{x})$ in the sequel for short. We make the following assumption on the feasible region $\mathcal{X}$.

Assumption 1. *The feasible region $\mathcal{X}$ is a compact set in $\mathbb{R}^d$ that satisfies $\mathrm{cl}(\mathrm{int}(\mathcal{X})) = \mathcal{X}$, where $\mathrm{cl}(\mathcal{A})$ and $\mathrm{int}(\mathcal{A})$ denote the closure and interior of a set $\mathcal{A}$, respectively.*

The condition that $\mathrm{cl}(\mathrm{int}(\mathcal{X})) = \mathcal{X}$ in Assumption 1 is a condition on constraint qualifications. It implies that for any boundary point $\overline{\boldsymbol{x}} \in \mathcal{X}$, there exists a sequence of interior points $\{\boldsymbol{x}_i\}$ such that $\boldsymbol{x}_i \to \overline{\boldsymbol{x}}$. This is similar to Slater's condition for convex constrained optimization problems (Boyd et al. 2004), which ensures strict feasibility, although the feasible region $\mathcal{X}$ need not be convex. With Assumption 1, for any $\boldsymbol{x} \in \mathcal{X}$ and any d-dimensional ball centered at $\boldsymbol{x}$ with positive radius, say $\mathcal{S}(\boldsymbol{x})$, the volume of $\mathcal{X} \cap \mathcal{S}(\boldsymbol{x})$ is larger than zero. Based on that, we can ensure that $\mathcal{X} \cap \mathcal{S}(\boldsymbol{x})$ has positive probability to be sampled for any $\boldsymbol{x} \in \mathcal{X}$ if the sampling distribution has positive density on $\mathcal{X}$.

Let $\varepsilon(\boldsymbol{x}) = G(\boldsymbol{x}) - g(\boldsymbol{x})$ be the simulation noise at each point $\boldsymbol{x} \in \mathcal{X}$. We impose Assumption 2 on $\varepsilon(\boldsymbol{x})$.

Assumption 2. *For all $\boldsymbol{x} \in \mathcal{X}$, $\varepsilon(\boldsymbol{x})$ follows a normal distribution with mean 0 and known variance $\lambda^2(\boldsymbol{x})$ (i.e., $\varepsilon(\boldsymbol{x}) \sim \mathcal{N}(0, \lambda^2(\boldsymbol{x}))$). The variance $\lambda^2(\boldsymbol{x})$ is positive and bounded on $\mathcal{X}$ (i.e., $0 < \lambda^2(\boldsymbol{x}) \le \lambda^2_{max}$ for all $\boldsymbol{x} \in \mathcal{X}$).*

The simulation noise $\varepsilon(\boldsymbol{x})$ is assumed to follow a normal distribution with known variance $\lambda^2(\boldsymbol{x})$ in the theoretical analysis because the convergence analysis of GPRS algorithms is based on the properties of the kriging surfaces. In the literature of kriging-based Bayesian optimization algorithms and Gaussian process regression, results are typically derived under the assumption of known (equal or unequal) variances (e.g., the stochastic kriging approach of Ankenman et al. 2010 and the KGCP algorithm of Scott et al. 2011). In the practical COvS setting, variances are typically unknown but can be estimated. Different methods have been proposed to estimate the unknown variances in practice. For instance, under the homoscedastic context, Huang et al. (2006) use the maximum likelihood estimation (MLE) method to estimate the unknown variance with a single observation at each design point; under the heteroscedastic context, Ankenman et al. (2010) directly calculate the sample variance by taking multiple observations at each design point. To keep the single-observation feature of GPRS algorithms, a kernel-based method, which utilizes neighborhood information to estimate the variance of the simulation noise, is proposed in Section 5 of this paper. Additionally, it is reasonable to assume the boundedness of $\lambda^2(\boldsymbol{x})$ on compact $\mathcal{X}$ because it is commonly satisfied in practical applications.

2.1. Gaussian Process Regression

Algorithms under the GPRS framework use Gaussian process regression to build a surrogate model of the objective function $g(\boldsymbol{x})$ in each iteration. It takes a Bayesian viewpoint and assumes that the unknown objective function $g(\boldsymbol{x})$ is a (random) sample path of a Gaussian process $f_{\mathcal{GP}}$ on $\mathcal{X}$, with the *mean function* $\mu_0 : \mathcal{X} \to \mathbb{R}$, defined by $\mu_0(\boldsymbol{x}) = \mathbb{E}[f_{\mathcal{GP}}(\boldsymbol{x})]$, and the *covariance function* $k_0 : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, defined by $k_0(\boldsymbol{x},\boldsymbol{x}') = \mathbb{E}[(f_{\mathcal{GP}}(\boldsymbol{x}) - \mu_0(\boldsymbol{x}))(f_{\mathcal{GP}}(\boldsymbol{x}') - \mu_0(\boldsymbol{x}'))]$. In GPRS algorithms, we require the mean and covariance functions to satisfy the following assumption.

Assumption 3. *The mean function $\mu_0(\boldsymbol{x})$ is continuous on $\mathcal{X}$, and the covariance function $k_0(\boldsymbol{x},\boldsymbol{x}') = \tau^2\rho(\boldsymbol{x}-\boldsymbol{x}')$ for some $\tau > 0$ and some continuous function $\rho : \mathbb{R}^d \to \mathbb{R}$, which further satisfies the following three conditions:*

i. $\rho(|\boldsymbol{\delta}|) = \rho(\boldsymbol{\delta})$, where $|\cdot|$ means taking absolute value component wise;

ii. $\rho(\boldsymbol{\delta})$ is decreasing in $\boldsymbol{\delta}$ component wise for $\boldsymbol{\delta} \ge \mathbf{0}$;

iii. $\rho(\mathbf{0}) = 1$, $\rho(\boldsymbol{\delta}) \to 0$ as $\|\boldsymbol{\delta}\| \to \infty$, and for some $0 < C < \infty$ and some $\epsilon, \delta > 0$, $1 - \rho(\boldsymbol{\delta}) \le C|\log(\|\boldsymbol{\delta}\|)|^{-1-\epsilon}$ for all $\|\boldsymbol{\delta}\| < \delta$, where $\|\cdot\|$ denotes the Euclidean norm.

Assumption 3 is in general a weak assumption, and most covariance functions used in practice satisfy it. Notable examples include the power exponential

covariance function $k_0(\boldsymbol{x},\boldsymbol{x}') = \tau^2\exp\{-\sum_{j=1}^{d}\theta_j|x_j - x_j'|^\eta\}$ with $\theta_j > 0$ and $0 < \eta \le 2$ and the Matérn covariance function; see Rasmussen and Williams (2006, chapter 4) for more types of covariance functions. The mean and covariance functions reflect one's prior belief about the unknown function $g(\boldsymbol{x})$ and are subject to user's choice. When no structural information for $g(\boldsymbol{x})$ is available, it is a convention to set $\mu_0 \equiv 0$. Shen et al. (2018) demonstrate that it is beneficial to embed some stylized models into μ_0 if they are capable of capturing the structure information of $g(\boldsymbol{x})$. Furthermore, Assumption 3 also implies that the correlation function $(1/\tau^2)k_0$ is *stationary* (i.e., it depends on $\boldsymbol{x}$ and $\boldsymbol{x}'$ only through the difference $\boldsymbol{x} - \boldsymbol{x}'$), and the sample paths of $f_{\mathcal{GP}}$ are continuous with probability 1 (Adler and Taylor 2007, theorem 1.4.1).

Suppose that GPRS algorithms have simulated a set of solutions denoted by $\boldsymbol{X}^n = \{\boldsymbol{x}_i\}_{i=1}^n$ with the corresponding simulation observations $\boldsymbol{G}^n = (G(\boldsymbol{x}_1),\dots,G(\boldsymbol{x}_n))^\top \in \mathbb{R}^n$. Then, conditioned on these observations, the conditional Gaussian process is still a Gaussian process whose mean and covariance functions are given by $\mu_n(\boldsymbol{x}) = \mathbb{E}[f_{\mathcal{GP}}(\boldsymbol{x})|\{\boldsymbol{X}^n,\boldsymbol{G}^n\}]$ and $k_n(\boldsymbol{x},\boldsymbol{x}') = \mathbb{E}[(f_{\mathcal{GP}}(\boldsymbol{x}) - \mu_n(\boldsymbol{x}))(f_{\mathcal{GP}}(\boldsymbol{x}') - \mu_n(\boldsymbol{x}'))|\{\boldsymbol{X}^n,\boldsymbol{G}^n\}]$, respectively. By Assumption 3, they can be expressed as

$$\mu_n(\boldsymbol{x}) = \mu_0(\boldsymbol{x}) + k_0(\boldsymbol{x},\boldsymbol{X}^n)[k_0(\boldsymbol{X}^n,\boldsymbol{X}^n) + \boldsymbol{\Sigma}^n]^{-1}[\boldsymbol{G}^n - \mu_0(\boldsymbol{X}^n)], \tag{2}$$

$$k_n(\boldsymbol{x},\boldsymbol{x}') = k_0(\boldsymbol{x},\boldsymbol{x}') - k_0(\boldsymbol{x},\boldsymbol{X}^n)[k_0(\boldsymbol{X}^n,\boldsymbol{X}^n) + \boldsymbol{\Sigma}^n]^{-1}k_0(\boldsymbol{X}^n,\boldsymbol{x}'), \tag{3}$$

where $\boldsymbol{\Sigma}^n$ is an n-dimensional diagonal matrix with simulation noise variance $\lambda^2(\boldsymbol{x}_i)$ being its diagonal elements, $k_0(\boldsymbol{X}^n,\boldsymbol{X}^n) = [k_0(\boldsymbol{x}_i - \boldsymbol{x}_j)]_{1\le i,j\le n} \in \mathbb{R}^{n\times n}$, $k_0(\boldsymbol{x},\boldsymbol{X}^n) = (k_0(\boldsymbol{x} - \boldsymbol{x}_1),\dots,k_0(\boldsymbol{x} - \boldsymbol{x}_n)) \in \mathbb{R}^{1\times n}$, and $k_0(\boldsymbol{X}^n,\boldsymbol{x}') = (k_0(\boldsymbol{x}_1 - \boldsymbol{x}'),\dots,k_0(\boldsymbol{x}_n - \boldsymbol{x}'))^\top \in \mathbb{R}^n$.

Similar to many other kriging-based Bayesian optimization algorithms (e.g., the KGCP algorithm of Scott et al. 2011), $\mu_n(\boldsymbol{x})$ is viewed as our prediction of $g(\boldsymbol{x})$ given the observations $\{\boldsymbol{X}^n,\boldsymbol{G}^n\}$. It serves as the basis for exploitative search. In addition, the Gaussian process regression also provides information on the uncertainty of the prediction measured by the variance of the conditional Gaussian process (i.e., $k_n(\boldsymbol{x},\boldsymbol{x})$ for $\boldsymbol{x} \in \mathcal{X}$). As noted by Sun et al. (2014), it can serve as the basis for explorative search. In this section, we focus on the framework of GPRS algorithms. The discussion on how to use $k_n(\boldsymbol{x},\boldsymbol{x})$ to guide searches will be provided in Section 5.

2.2. Gaussian Process-Based Random Search Algorithms

We propose a framework for GPRS algorithms for solving Problem (1). Let s denote the iteration counter, r denote the number of solutions sampled in each iteration, and n denote the counter of the sampled solutions. Define $\boldsymbol{X}^n$ and $\boldsymbol{G}^n$ as the set of solutions and the vector of observations, respectively. Let $f_n(\boldsymbol{x})$ be the sampling density function constructed based on $\{\boldsymbol{X}^n,\boldsymbol{G}^n\}$. In the course of GPRS algorithms, the unknown objective function $g(\boldsymbol{x})$ is estimated using the conditional mean function $\mu_n(\boldsymbol{x})$, as defined in Equation (2). Design points are sequentially sampled according to the sampling density function $f_n(\boldsymbol{x})$. To ensure the global convergence, the sampling density function $f_n(\boldsymbol{x})$ needs to satisfy the following assumption.

Assumption 4. *There exists a positive constant $\alpha > 0$ such that $f_n(\boldsymbol{x}) \ge \alpha$ for all n and $\boldsymbol{x} \in \mathcal{X}$.*

Assumption 4 essentially ensures that any d-dimensional ball centered at $\boldsymbol{x} \in \mathcal{X}$ with positive radius can be sampled with a positive probability. This assumption is commonly used in random search-based algorithms for both DOvS and COvS problems (Andradóttir and Prudius 2009, 2010; Sun et al. 2014; Kiatsupaibul et al. 2018).

Given the variance of simulation noise $\lambda^2(\boldsymbol{x})$, the framework of GPRS algorithms can be presented as follows.

Step 0 (initialization). Impose a Gaussian process with μ_0 and k_0 that satisfy Assumption 3. Specify a $r > 0$. Set $s = 0$, $n = 0$, $\boldsymbol{X}^0 = \emptyset$, and $\boldsymbol{G}^0 = \emptyset$. Furthermore, set $f_0(\boldsymbol{x})$ as a user-specified sampling distribution over $\mathcal{X}$.

Step 1 (sampling). Set $s = s + 1$. Sample $\boldsymbol{x}_{r(s-1)+1},\dots,\boldsymbol{x}_{rs}$ independently from $f_n(\boldsymbol{x})$, and obtain corresponding simulation observations $G(\boldsymbol{x}_{r(s-1)+1}),\dots,G(\boldsymbol{x}_{rs})$ independently from all previous observations.

Step 2 (calculation). Set $n = rs$. Let $\boldsymbol{X}^n = \boldsymbol{X}^{r(s-1)} \cup \{\boldsymbol{x}_{r(s-1)+1},\dots,\boldsymbol{x}_{rs}\}$ and $\boldsymbol{G}^n = ([\boldsymbol{G}^{r(s-1)}]^\top, G(\boldsymbol{x}_{r(s-1)+1}),\dots,G(\boldsymbol{x}_{rs}))^\top$. Calculate $\mu_n(\boldsymbol{x})$ according to Equation (2). Let $\boldsymbol{x}_n^* = \arg\max_{\boldsymbol{x}\in\mathcal{X}}\mu_n(\boldsymbol{x})$, and break the tie arbitrarily if it exists. Then, construct the sampling distribution $f_n(\boldsymbol{x})$ according to user-specified rules.

Step 3 (stopping). If the stopping condition is not met, go to step 1; otherwise, stop and output $\boldsymbol{x}_n^*$ and $\mu_n(\boldsymbol{x}_n^*)$ as the estimated optimal solution and the estimated optimal objective value.

We summarize some features of GPRS algorithms for COvS problems. First, algorithms under the GPRS framework are single-observation random search algorithms for COvS problems. Compared with those random search algorithms with multiple observations, requiring only a single observation at each solution allows the algorithm to better explore the feasible region given the same simulation budget, which is an appealing property for COvS problems. Second, GPRS algorithms allow a wide range of sampling distributions, as Assumption 4 can easily be satisfied. One simple way is to sample each solution from an exploitative sampling distribution with probability p and from another uniform sampling distribution with probability $1 - p$. Many existing algorithms (e.g., the shrinking-ball

algorithm of Kiatsupaibul et al. 2018) adopt this way. One can also consider constructing some adaptive sampling distributions (e.g., by utilizing the uncertainty estimation of the Gaussian process regression). Third, rather than averaging observations in a shrinking ball centered at a solution, which is commonly used in single-observation random search algorithms to estimate the objective function (Kiatsupaibul et al. 2018), GPRS algorithms use the conditional mean function of the Gaussian process. As a result, the analysis of the asymptotic behavior of GPRS algorithms is different from those of shrinking-ball algorithms, and it relies on the property of the Gaussian process regression.

We note that GPRS algorithms provided in this section are a theoretical framework and cannot be implemented without specifying several key components (e.g., the sampling distribution, the sampling scheme, and the variance estimation method). Before discussing the implementation of GPRS algorithms in Section 5, we first analyze the global convergence and the rate of convergence in next two sections, which are applicable to any algorithm within this framework.

3. Global Convergence

Let $g^* = \max_{x\in\mathcal{X}} g(\boldsymbol{x})$ be the optimal objective function value, and let $\mathcal{X}^* = \arg\max_{x\in\mathcal{X}} g(\boldsymbol{x})$ be the set of optimal solutions. In this section, we establish the almost-sure global convergence of GPRS algorithms (i.e., we prove

$$\mathbb{P}\left\{\lim_{n\to\infty} \mu_n(\boldsymbol{x}_n^*) = g^*\right\} = 1 \quad \text{and}$$
$$\mathbb{P}\left\{\lim_{n\to\infty} d(\boldsymbol{x}_n^*, \mathcal{X}^*) = 0\right\} = 1, \tag{4}$$

where for any set $\mathcal{A} \subset \mathbb{R}^d$ and a point $\boldsymbol{x} \in \mathbb{R}^d$, we define the distance from $\boldsymbol{x}$ to $\mathcal{A}$ by $d(\boldsymbol{x}, \mathcal{A}) = \inf_{\boldsymbol{x}'\in\mathcal{A}} \|\boldsymbol{x} - \boldsymbol{x}'\|$). Notice that Equation (4) implies that the estimated optimal value converges to the true optimal value with probability 1 and the estimated optimal solution converges to the set of true optimal solutions with probability 1.

Even though the probability statements in Equation (4) are common goals for convergence analysis for COvS algorithms, there is a subtle difference between ours and the ones in the literature in terms of the randomness considered in these statements. For most algorithms in the literature (see, for instance, the ASR algorithm of Andradóttir and Prudius 2010 and the shrinking-ball algorithm of Kiatsupaibul et al. 2018), the objective function $g(\boldsymbol{x})$ is considered deterministic (but unknown), and the randomness comes from the sampling and the simulation experiments. In our case, however, the objective function $g(\boldsymbol{x})$ is assumed to be a (random) sample path from the Gaussian process $f_{\mathcal{GP}}$. Therefore, the randomness not only comes from the sampling and the simulation experiments but also from the Gaussian process. This treatment of the objective function is consistent with the Bayesian viewpoint of the Gaussian process regression, and it has also been used to analyze the convergence of the sequential optimization algorithm based on the KGCP policy (Scott et al. 2011).

The convergence analysis of GPRS algorithms contains two major steps. In the first step, we establish the convergence of the conditional variance function $k_n(\boldsymbol{x}, \boldsymbol{x})$. Based on that, in the second step, we prove the uniform convergence of the conditional mean function $\mu_n(\boldsymbol{x})$. Then, the almost-sure convergence of GPRS algorithms can be derived.

3.1. The Convergence of the Conditional Variance

In this subsection, our goal is to show that the conditional variance $k_n(\boldsymbol{x}, \boldsymbol{x})$ converges to zero as $n \to \infty$ for any $\boldsymbol{x} \in \mathcal{X}$. For any $\boldsymbol{x} \in \mathcal{X}$, let $\mathcal{S}(\boldsymbol{x}, \epsilon)$ denote the closed d-dimensional ball centered at $\boldsymbol{x}$ with radius $\epsilon > 0$, and let $s_n(\boldsymbol{x}, \epsilon)$ denote the number of solutions in $\mathcal{S}(\boldsymbol{x}, \epsilon)$ among all n sampled solutions. We first establish the following lemma regarding to the asymptotic behavior of $s_n(\boldsymbol{x}, \epsilon)$, whose proof is an application of the Law of Large Numbers and hence, is omitted.

Lemma 1. *Suppose that Assumption 1 holds and that distributions with density functions ψ_i satisfying $\psi_i \geq \alpha > 0$ on $\mathcal{X}$ are used to generate points $\boldsymbol{x}_i \in \boldsymbol{X}^n$, for $i = 1, \ldots, n$. Then, for any fixed $\epsilon > 0$ and any $\boldsymbol{x} \in \mathcal{X}$, $s_n(\boldsymbol{x}, \epsilon) \to \infty$ almost surely as $n \to \infty$.*

Lemma 1 shows that for any small ball centered at $\boldsymbol{x} \in \mathcal{X}$ with radius ϵ, the number of sampled solutions in that ball goes to infinity as $n \to \infty$. This implies that the sampled solution will eventually be dense on $\mathcal{X}$, and it provides a preliminary result for our convergence analysis.

Another preliminary result is that the conditional variance is upper bounded, which is a direct result of lemma 4 of Ding et al. (2022), and it is provided in Lemma 2.

Lemma 2 (Ding et al. 2022, lemma 4). *Fix a compact set $\mathcal{A} \subset \mathcal{X}$. Suppose $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in \mathcal{A}$. If Assumptions 2 and 3 hold, then for any $\boldsymbol{x} \in \mathcal{A}$,*

$$k_n(\boldsymbol{x}, \boldsymbol{x}) \leq \tau^2 - \frac{n \min_{\boldsymbol{x}'\in\mathcal{A}} [k_0(\boldsymbol{x}, \boldsymbol{x}')]^2}{n\tau^2 + \lambda_{max}^2},$$

where $k_n(\cdot, \cdot)$ is defined in Equation (3).

Based on Lemmas 1 and 2, we have the following lemma on the convergence of the conditional variance function. This is an important result, and its proof is provided in Section EC.1.1 of the e-companion.

Lemma 3. *Suppose that Assumptions 1–3 hold and that distributions with density functions ψ_i satisfying $\psi_i \geq \alpha >$*

0 *on* $\mathcal{X}$ *are used to generate points* $\boldsymbol{x}_i \in \boldsymbol{X}^n$, *for* $i = 1, \ldots, n$. *Then, for any* $\boldsymbol{x} \in \mathcal{X}$, $k_n(\boldsymbol{x}, \boldsymbol{x}) \to 0$ *almost surely as* $n \to \infty$.

Lemma 3 provides the almost-sure pointwise convergence of the conditional variance $k_n(\boldsymbol{x}, \boldsymbol{x})$ to zero as $n \to \infty$. By Chebyshev's inequality, Lemma 3 further implies that the conditional mean function $\mu_n(\boldsymbol{x})$ converges in probability for any $\boldsymbol{x} \in \mathcal{X}$. However, this pointwise convergence is not enough. In the following subsection, we show that the conditional mean function converges uniformly.

3.2. The Global Convergence of GPRS Algorithms

Notice that Lemma 3 only implies the pointwise convergence of $\mu_n(\boldsymbol{x})$ to $\mathbb{E}[\mu_n(\boldsymbol{x})]$. However, this is not enough. In this subsection, our goal is to show that the conditional mean function $\mu_n(\boldsymbol{x})$ converges to $g(\boldsymbol{x})$ uniformly as $n \to \infty$, based on which the global convergence of the algorithm can be easily derived. The following lemma of Bect et al. (2019) is critical to fill the gap. It shows that $\mu_n(\boldsymbol{x})$ converges uniformly to a limiting function.

Lemma 4 (Bect et al. 2019, proposition 2.9). *Suppose that Assumptions 1–3 hold. Then,* $\mu_n(\boldsymbol{x})$ *converges uniformly on* $\mathcal{X}$ *to a function, denoted by* $\mu_\infty(\boldsymbol{x})$, *almost surely as* $n \to \infty$.

Notice that Lemma 4 only ensures that $\mu_n(\cdot)$ converges uniformly to a certain function, which is not necessarily $g(\boldsymbol{x})$. By combining Lemmas 3 and 4, we have the following proposition, which shows that the limit is indeed $g(\boldsymbol{x})$.

Proposition 1. *Suppose that Assumptions 1–3 hold and that distributions with density functions* ψ_i *satisfying* $\psi_i \geq \alpha > 0$ *on* $\mathcal{X}$ *are used to generate points* $\boldsymbol{x}_i \in \boldsymbol{X}^n$, *for* $i = 1, \ldots, n$. *Then,* $\mu_n(\boldsymbol{x}) \to g(\boldsymbol{x})$ *uniformly on* $\mathcal{X}$ *almost surely as* $n \to \infty$.

Proof. Fix any $\boldsymbol{x} \in \mathcal{X}$. First notice that

$$\begin{aligned}\mathbb{E}[k_n(\boldsymbol{x}, \boldsymbol{x})] &= \mathbb{E}[\mathbb{E}[(g(\boldsymbol{x}) - \mu_n(\boldsymbol{x}))^2 | \{\boldsymbol{X}^n, \boldsymbol{G}^n\}]] \\ &= \mathbb{E}[(g(\boldsymbol{x}) - \mu_n(\boldsymbol{x}))^2]. \end{aligned} \tag{5}$$

By Lemma 3, $k_n(\boldsymbol{x}, \boldsymbol{x}) \to 0$ almost surely as $n \to \infty$. Then, together with the fact that $0 \leq k_n(\boldsymbol{x}, \boldsymbol{x}) \leq k_0(\boldsymbol{x}, \boldsymbol{x}) = \tau^2$ from Equation (EC.1) in the proof of Lemma 3 in the e-companion, we can conclude that $\mathbb{E}[k_n(\boldsymbol{x}, \boldsymbol{x})] \to \mathbb{E}[0] = 0$ by the dominated convergence theorem (Durrett 2010, theorem 1.5.6). It implies that $\mathbb{E}[(g(\boldsymbol{x}) - \mu_n(\boldsymbol{x}))^2] \to 0$ (i.e., $\mu_n(\boldsymbol{x}) \to g(\boldsymbol{x})$ in L^2) for any $\boldsymbol{x} \in \mathcal{X}$. Lemma 4 implies that $\mu_n(\boldsymbol{x}) \to \mu_\infty(\boldsymbol{x})$ almost surely on $\mathcal{X}$. Because of the almost-sure uniqueness of convergence in probability (Gut 2013, theorem 2.1 in chapter 5), it can be obtained that $\mathbb{P}\{\mu_\infty(\boldsymbol{x}) = g(\boldsymbol{x})\} = 1$, for any $\boldsymbol{x} \in \mathcal{X}$.

Consider a dense but countable subset $\overline{\mathcal{X}}$ of $\mathcal{X}$ (e.g., the set of all $\boldsymbol{x} \in \mathcal{X}$ such that all elements of $\boldsymbol{x}$ are rational numbers). Then, we have $\mathbb{P}\{\mu_\infty(\boldsymbol{x}) = g(\boldsymbol{x}),$ for all $\boldsymbol{x} \in \overline{\mathcal{X}}\} = 1$, because the probability measure is a nonnegative countably additive set function (Durrett 2010, p. 1). We now focus on one generic sample path such that $\mu_\infty(\boldsymbol{x}) = g(\boldsymbol{x})$ for all $\boldsymbol{x} \in \overline{\mathcal{X}}$ and $\mu_n(\boldsymbol{x}) \to \mu_\infty(\boldsymbol{x})$ uniformly on $\mathcal{X}$ (by Lemma 4). Recall that $\mu_n(\boldsymbol{x})$ is continuous on $\mathcal{X}$ for each n; then, $\mu_\infty(\boldsymbol{x})$ is also continuous on $\mathcal{X}$, as the uniform convergence maintains continuity (Tao 2009, corollary 14.3.2). Additionally, $g(\boldsymbol{x})$ is continuous on $\mathcal{X}$. For any $\boldsymbol{x} \in \mathcal{X}$, because $\overline{\mathcal{X}}$ is a dense subset, we can find a sequence $\boldsymbol{x}_i \in \overline{\mathcal{X}}$, $i = 1, 2, \ldots$, such that $\boldsymbol{x}_i \to \boldsymbol{x}$ as $i \to \infty$. Hence, $\mu_\infty(\boldsymbol{x}_i) \to \mu_\infty(\boldsymbol{x})$ and $g(\boldsymbol{x}_i) \to g(\boldsymbol{x})$, as $i \to \infty$. Because $\mu_\infty(\boldsymbol{x}_i) = g(\boldsymbol{x}_i)$, for $i = 1, 2, \ldots$, it can be concluded that $\mu_\infty(\boldsymbol{x}) = g(\boldsymbol{x})$ because of the uniqueness of limit (Tao 2009, proposition 6.1.7). Therefore, we conclude that $\mathbb{P}\{\mu_\infty(\boldsymbol{x}) = g(\boldsymbol{x}),$ for all $\boldsymbol{x} \in \mathcal{X}\} = 1$. The proof is then completed by combining this result with Lemma 4. □

Proposition 1 shows that $\mu_n(\boldsymbol{x})$ defined in Equation (2) converges uniformly to $g(\boldsymbol{x})$ almost surely as the number of observations goes to infinity. It plays a crucial role in establishing the convergence of GPRS algorithms. With the result of Proposition 1, by theorem 5.3 of Shapiro et al. (2009), it is straightforward to establish the global convergence of GPRS algorithms, which is formally stated in Theorem 1.

Theorem 1. *If Assumptions 1–4 hold and a GPRS algorithm is used to solve Problem* (1), *then* $\mu_n(\boldsymbol{x}_n^*) \to g^*$ *and* $d(\boldsymbol{x}_n^*, \mathcal{X}^*) \to 0$ *almost surely as* $n \to \infty$.

Theorem 1 shows that algorithms under the GPRS framework are globally convergent when solving COvS problems. It is a desirable property for random search algorithms. As discussed at the beginning of this section, the convergence result in Theorem 1 is different from (and weaker than) the typical ones in the literature, which treat the objective function as a deterministic function. To compare with the ones in the literature, we may interpret our result as follows. GPRS algorithms have the global convergence (in the typical sense) for almost all objective functions that are sampled randomly from the Gaussian process that satisfies Assumption 3. In other words, for those objective functions for which GPRS algorithms fail to converge, their combined probability under the Gaussian process assumption is zero.

4. Rate of Convergence

Although almost all random search COvS algorithms in the literature have some sort of convergence guarantees, very few of them have results on the rate of convergence, which provides valuable information on the efficiency and scalability of the algorithm. An important reason for the lack of rate of convergence results is

the difficulty in analyzing them. In this section, we show that by leveraging on the properties of Gaussian process regression, we are able to establish the rate of convergence of GPRS algorithms, and the rate of convergence does provide valuable insights on the performance of the algorithm.

Let $\{\epsilon_n\}_{n\geq 1}$ be a deterministic sequence such that $\epsilon_n > 0$ and $\epsilon_n \to 0$ as $n \to \infty$. In this section, we want to find $\{\epsilon_n\}_{n\geq 1}$ so that we can show that the rate of convergence of GPRS algorithms is $O_p(\epsilon_n)$ (i.e., for every $\delta > 0$, there exist constants $C_\delta \in (0,\infty)$ and $N_\delta \in \mathbb{N}$ such that

$$\mathbb{P}\{|\mu_n(\boldsymbol{x}_n^*) - g^*| > C_\delta \epsilon_n\} < \delta, \tag{6}$$

for all $n > N_\delta$). Similar to that in Equation (4), the probability in Equation (6) also includes the randomness in the Gaussian process as well as the randomness in the sampling and the simulation experiments.

To simplify the presentation, we also adopt the $\tilde{O}_p$ notation. For any constant k, if the rate of convergence of an algorithm is $O_p(\epsilon_n \log^k \epsilon_n)$, it is denoted as $\tilde{O}_p(\epsilon_n)$. The $\tilde{O}_p$ notation is a variant of the O_p notation, which ignores the logarithmic factors in the rate and captures the main effect. Its deterministic version is popular in the area of theoretical computer science.

The rate of convergence analysis of the algorithm also has two steps. In the first step, we establish the rate of convergence of the conditional variance function $k_n(\boldsymbol{x},\boldsymbol{x})$ to zero. Based on that, in the second step, we prove the rate of convergence of the maximum value of conditional mean function $\mu_n(\boldsymbol{x}_n^*)$ to the global optimal value g^*. In the following subsections, we elaborate these two steps.

4.1. The Rate of Convergence of the Conditional Variance

Before we establish the rate of convergence of the conditional variance $k_n(\boldsymbol{x},\boldsymbol{x})$, we first make two stronger assumptions on the Gaussian process and the feasible region, which are needed to establish the rate of convergence of GPRS algorithms. The first one is on the Gaussian process.

Assumption 5. *The first-order derivative surfaces of the Gaussian process $f_{\mathcal{GP}}$ are stationary Gaussian processes and have continuous sample paths almost surely on $\mathcal{X}$.*

Compared with Assumption 3, Assumption 5 imposes stronger regularity conditions on the sample paths of the Gaussian process $f_{\mathcal{GP}}$. The sample paths $g(\boldsymbol{x})$ are not only continuous but also continuously differentiable. This requires that the mean function $\mu_0(\boldsymbol{x})$ is continuously differentiable. The correlation function $(1/\tau^2)k_0(\boldsymbol{x},\boldsymbol{x}')$ needs to have continuous second-order derivatives with finite value at the point $(\boldsymbol{x},\boldsymbol{x})$, and the correlation functions of the derivatives surfaces should satisfy Assumption 3(iii) (Abrahamsen 1997). Many commonly used correlation functions (e.g., the Gaussian correlation function, the Matérn correlation function with the smoothness parameter v ($v > 1$) being a half-integer, and the rational quadratic correlation function) satisfy these properties (see the analysis provided in Section EC.3 of the e-companion).

Because differentiation is a linear operator, the first-order derivative surfaces of Gaussian process are still Gaussian processes (Azaïs and Wschebor 2009, p. 29). Assumption 5 indicates that these derivative surfaces are bounded almost surely on $\mathcal{X}$ (Adler and Taylor 2007, theorem 1.5.4). As we will see later, these boundedness properties are critical in the analysis of the rate of convergence of GPRS algorithms.

Besides the stronger assumption on the Gaussian process, we also need a stronger assumption on the feasible region, which is stated as follows.

Assumption 6. *The feasible region $\mathcal{X} \subset \mathbb{R}^d$ is a bounded convex set with nonempty interior.*

Compared with Assumption 1, Assumption 6 is stronger by requiring the convexity of the feasible region $\mathcal{X}$. With this assumption, we can establish a lower bound for the volume of the intersection of $\mathcal{X}$ and a small ball $\mathcal{S}(\boldsymbol{x},\epsilon)$, which is proved by Baumert and Smith (2002, p. 14) and formally stated in Lemma 5.

Lemma 5 (Baumert and Smith 2002, p. 14). *If Assumption 6 holds, then for any $\boldsymbol{x} \in \mathcal{X}$ and sufficiently small $\epsilon > 0$, there exists some constant $C > 0$, which may depend on $\mathcal{X}$, such that*

$$\nu(\mathcal{S}(\boldsymbol{x},\epsilon) \cap \mathcal{X}) \geq C \cdot \nu(\mathcal{S}(\boldsymbol{x},\epsilon)), \tag{7}$$

where $\nu(\cdot)$ denotes the d-dimensional volume.

For the interior points of $\mathcal{X}$, Equation (7) always holds even without Assumption 6 when ϵ is small enough. However, for $\boldsymbol{x}$ on the boundary of $\mathcal{X}$, it is not necessarily the case. Lemma 5 helps to rule out those situations. It ensures that for any $\boldsymbol{x} \in \mathcal{X}$, the sampling probability of $\mathcal{S}(\boldsymbol{x},\epsilon) \cap \mathcal{X}$ has a lower bound that is proportional to the volume of the ball $\mathcal{S}(\boldsymbol{x},\epsilon)$. This result is a foundation for the convergence analysis of the shrinking-ball algorithms. Inspired by the shrinking-ball idea, in the following analysis, we also construct balls that shrink with the number of sampled points, through which we can investigate the increasing rate of the sampled design points in local areas.

Recall that $s_n(\boldsymbol{x},\epsilon)$ denotes the number of points sampled in the closed d-dimensional ball $\mathcal{S}(\boldsymbol{x},\epsilon)$ with a deterministic radius ϵ. We further let $s_n(\boldsymbol{x},r_n)$ denote the number of points sampled in another closed d-dimensional ball $\mathcal{S}(\boldsymbol{x},r_n)$, centered at $\boldsymbol{x}$ with radius r_n. We have the following lemma, whose proof is provided in Section EC.2.1 of the e-companion.

Lemma 6. *Suppose that Assumption 6 holds and that distributions with density functions ψ_i satisfying $\psi_i \geq \alpha > 0$ on $\mathcal{X}$ are used to generate points $\boldsymbol{x}_i \in \boldsymbol{X}^n$, for $i = 1,\ldots,n$. Let $\varepsilon(n) = (\log\log n)/\log n$, $\gamma_n = \gamma - a\varepsilon(n)$, $p_n = \gamma - b\varepsilon(n)$,*

and $r_n = r_0 n^{-\frac{1-\gamma_n}{d}}$ *with* $r_0 > 0$. *Then, for any* $\gamma \in (0,1)$ *and* $b - 1 > a$, $s_n(\boldsymbol{x}, r_n)$ *is* $\Omega(n^{p_n})$[1] *almost surely (i.e.,* $\mathbb{P}\{s_n(\boldsymbol{x}, r_n)$ *is* $\Omega(n^{p_n})\} = 1$*) for any* $\boldsymbol{x} \in \mathcal{X}$.

The result of Lemma 6 is stronger than that of Lemma 1. In particular, Lemma 1 shows that $s_n(\boldsymbol{x}, \epsilon)$ increases to ∞, whereas Lemma 6 shows the increasing rate of $s_n(\boldsymbol{x}, r_n)$ to ∞. Similar results have also been developed by shrinking-ball algorithms (see, for instance, Baumert and Smith 2002, Andradóttir and Prudius 2010). Based on the work of Andradóttir and Prudius (2010), we prove a slightly sharper result by including the logarithm factors in the increasing rate of $s_n(\boldsymbol{x}, r_n)$. This is used to derive the rate of convergence of GPRS algorithms in the following analysis.

Lemma 6 indicates that the increasing rate of $s_n(\boldsymbol{x}, r_n)$ is close to $\Omega(n^{\gamma})$, which relies on the contracting rate of the radius. Similar to the proof of Lemma 3, by combining the increasing rate of $s_n(\boldsymbol{x}, r_n)$ and the upper bound of $k_n(\boldsymbol{x}, \boldsymbol{x})$ (Lemma 2), we can then establish the rate of convergence of the conditional variance $k_n(\boldsymbol{x}, \boldsymbol{x})$. This result is formally stated in Lemma 7, and its proof is provided in Section EC.1.2 of the e-companion.

Lemma 7. *Suppose that Assumptions 2* and *6 hold and that distributions with density functions* ψ_i *satisfying* $\psi_i \geq \alpha > 0$ *on* $\mathcal{X}$ *are used to generate points* $\boldsymbol{x}_i \in \boldsymbol{X}^n$, *for* $i = 1, \ldots, n$. *If the correlation function satisfies* $\rho(\boldsymbol{x}, \boldsymbol{x}') \geq 1 - C_r \|\boldsymbol{x} - \boldsymbol{x}'\|^{\eta}$ *for* $\boldsymbol{x}' \in \mathcal{S}(\boldsymbol{x}, r_n)$ *with constants* $C_r > 0$ *and* $0 < \eta \leq 2$, *then* $k_n(\boldsymbol{x}, \boldsymbol{x})$ *is* $O(n^{-\kappa(n)})$ *almost surely for any* $\boldsymbol{x} \in \mathcal{X}$ *(i.e.,* $\mathbb{P}\{k_n(\boldsymbol{x}, \boldsymbol{x})$ *is* $O(n^{-\kappa(n)})\} = 1$*, where*

$$\kappa(n) = \frac{\eta}{d + \eta} - b\varepsilon(n) \text{ and } \varepsilon(n) = \frac{\log \log n}{\log n}, \tag{8}$$

for any $b > \eta/(d + \eta)$).

Lemma 7 is the main result of this subsection. It states that the rate of convergence of the conditional variance $k_n(\boldsymbol{x}, \boldsymbol{x})$ is $\tilde{O}(n^{-\eta/(d+\eta)})$ if the correlation function satisfies $\rho(\boldsymbol{x}, \boldsymbol{x}') \geq 1 - C_r \|\boldsymbol{x} - \boldsymbol{x}'\|^{\eta}$ for any $\boldsymbol{x}' \in \mathcal{S}(\boldsymbol{x}, r_n)$. We note that this inequality is met by many correlation functions (e.g., the power exponential correlation function, the rational quadratic correlation function, and the Matérn correlation function with smoothness parameter v being half-integer). According to the analyses of correlation functions in Section EC.3 of the e-companion, the Gaussian correlation function (a special case of the power exponential correlation function), the rational quadratic correlation function, and the Matérn correlation function (with v being half-integers and greater than one) all satisfy Assumption 5 and the inequality with $\eta = 2$. Hence, the rate of convergence of $k_n(\boldsymbol{x}, \boldsymbol{x})$ is $\tilde{O}(n^{-2/(d+2)})$ if Gaussian processes with these correlation functions are used.

Roughly speaking, the rate of convergence of $k_n(\boldsymbol{x}, \boldsymbol{x})$ is achieved by an adequate choice of the radius r_n. By Lemma 2, to make $k_n(\boldsymbol{x}, \boldsymbol{x})$ converge to zero, we need to let radii of the shrinking balls converge to zero while keeping the number of sampled points in these shrinking balls still going to infinity. Because the trends of the contracting rate of the radius and the increasing rate of the number of sampled points within shrinking balls are opposite, to obtain a good rate of convergence of $k_n(\boldsymbol{x}, \boldsymbol{x})$, it requires an adequate balance of these two rates. Under the condition of the unconditional covariance function $k_0(\boldsymbol{x}, \boldsymbol{x}')$, by letting the radius r_n contract at the rate $\tilde{O}(n^{-1/(d+\eta)})$, the proved rate of convergence of $k_n(\boldsymbol{x}, \boldsymbol{x})$ can be obtained, which is the optimal rate under the bound of Lemma 2. Notice that the dimension d is included because the expected number of sampled points in the shrinking balls is proportional to the volumes of the shrinking balls, which are proportional to r_n^d.

4.2. The Rate of Convergence of the Maximum Value of the Conditional Mean Function

In this subsection, our goal is to establish the rate of convergence of the maximum value of the conditional mean function $\mu_n(\boldsymbol{x}_n^*)$ to the global optimal value g^* (i.e., the rate of convergence of GPRS algorithms). To establish the rate, two preliminary results are needed. The first is stated in Lemma 8, which establishes an upper bound for the probability that the estimation error of $\mu_n(\boldsymbol{x})$ is beyond a certain threshold. This result is obtained by applying the Chernoff bound on the conditional mean function $\mu_n(\boldsymbol{x})$, and its proof is provided in Section EC.1.3 of the e-companion.

Lemma 8. *For any* $n = 1, 2, \ldots$ *and any* $\boldsymbol{x} \in \mathcal{X}$, *we have*

$$\mathbb{P}\{|\mu_n(\boldsymbol{x}) - g(\boldsymbol{x})| > \epsilon_n\} \leq 2\mathbb{E}[e^{-\epsilon_n^2/(2k_n(\boldsymbol{x}, \boldsymbol{x}))}],$$

for any deterministic sequence $(\epsilon_n)_{n \geq 1}$ *such that* $\epsilon_n > 0$ *and* $\epsilon_n \to 0$ *as* $n \to \infty$.

Lemma 8 indicates that the probability that the estimation error of $\mu_n(\boldsymbol{x})$ is greater than ϵ_n is bounded by how fast the conditional variance function $k_n(\boldsymbol{x}, \boldsymbol{x})$ converges to zero. If it converges at a faster rate than ϵ_n^2 for any $\boldsymbol{x} \in \mathcal{X}$, then the probability converges to zero for any feasible point.

The following lemma provides another preliminary result, which establishes an upper bound for the probability that no point is ever sampled near the global optima. Such a bound is critical to establishing the rate of convergence of GPRS algorithms. The proof of Lemma 9 is nontrivial, and it utilizes the properties of the derivative surfaces of the Gaussian process and the Borell-TIS inequality to construct a valid bound.

Lemma 9. *Suppose that Assumptions 5* and *6 hold and that distributions with density functions* ψ_i *satisfying* $\psi_i \geq \alpha > 0$ *on* $\mathcal{X}$ *are used to generate points* $\boldsymbol{x}_i \in \boldsymbol{X}^n$, *for* $i = 1, \ldots, n$. *Then, for sufficiently large* n, *there exist some positive*

constants C_j, a_j, and σ_j^2, for $j = 1\ldots,d$, such that

$$\mathbb{P}(\cap_{i=1}^{n}\{g^* - g(\boldsymbol{x}_i) > \epsilon_n\}) \le e^{-\frac{\alpha C\pi^{d/2}\epsilon_n^d n}{\Gamma(d/2+1)\log n}} + 2\sum_{j=1}^{d} e^{C_j\left(\frac{(\log n)^{1/d}}{\sqrt{d}} - a_j\right) - \frac{\left(\frac{(\log n)^{1/d}}{\sqrt{d}} - a_j\right)^2}{2\sigma_j^2}},$$

where C is as defined in Equation (7) and $\Gamma(\cdot)$ is the gamma function.

Proof. Because the proof is long, we only provide a sketch here, and we provide the detailed proof in Section EC.1.4 of the e-companion. The proof contains three major steps.

1. The first step is to bound the probability $\mathbb{P}(\cap_{i=1}^{n}\{g^* - g(\boldsymbol{x}_i) > \epsilon_n\})$ with the probabilities of another two disjoint events.

Suppose $\boldsymbol{x}^*$ is a solution in $\mathcal{X}^*$; by applying the mean value theorem and the Cauchy–Schwarz inequality, we can bound the gap $g^* - g(\boldsymbol{x}_i)$ with the product of the distance $\|\boldsymbol{x}^* - \boldsymbol{x}_i\|$ and the norm of the gradient $\nabla g(\boldsymbol{\xi})$, where $\boldsymbol{\xi}$ is a point in $\mathcal{X}$. Because the derivative surfaces $\dot{f}_{\mathcal{GP}}(\boldsymbol{x})_j$ are almost surely bounded on $\mathcal{X}$ for all $j = 1,2,\ldots,d$, let $\dot{g}(\boldsymbol{x})_j$ be the (random) sample path of $\dot{f}_{\mathcal{GP}}(\boldsymbol{x})_j$ and $\dot{g}^* = \max_{j=1,\ldots,d}\{\sup_{\boldsymbol{x}\in\mathcal{X}}|\dot{g}(\boldsymbol{x})_j|\}$, and we can further bound $\|\nabla g(\boldsymbol{\xi})\|$ with $\sqrt{d}\dot{g}^*$, and hence, $\mathbb{P}(\cap_{i=1}^{n}\{g^* - g(\boldsymbol{x}_i) > \epsilon_n\}) \le \mathbb{P}(\cap_{i=1}^{n}\{\sqrt{d}\dot{g}^*\|\boldsymbol{x}^* - \boldsymbol{x}_i\| > \epsilon_n\})$. For large-enough n, by dividing $\sqrt{d}\dot{g}^*$ at $(\log n)^{1/d}$, we can bound the aforementioned probability with the summation of another two probabilities $\mathbb{P}(\cap_{i=1}^{n}\{\|\boldsymbol{x}^* - \boldsymbol{x}_i\| > (\epsilon_n/(\log n)^{1/d})\})$ and $\mathbb{P}(\sqrt{d}\dot{g}^* \ge (\log n)^{1/d})$. The first probability corresponds to the probability that no point is ever sampled within a ball centered at one global optimal solution given a small-enough radius. The second probability is the tail probability of the supremum of all d derivative surfaces of the Gaussian process.

2. The second step is to bound the first probability by utilizing the fact that the sampling distributions are bounded below by a positive constant α. Specifically, this is achieved by constructing a sequence of Bernoulli random variables $(B_i)_{i\ge1}$ with parameter $\alpha C\cdot\nu(S(\boldsymbol{x},(\epsilon_n/(\log n)^{1/d}))$ and calculating the probability $\mathbb{P}\{\sum_{i=1}^{n}B_i > 0\}$.

3. The third step is to bound the second probability by utilizing the fact that the derivative surfaces are almost surely bounded and then applying the Borell-TIS inequality. The Borell-TIS inequality is frequently used in probability theory, so we think it is ok to use this name directly. To clarify the source of the abbreviation "TIS", we can add a reference (Adler and Taylor 2007, chapter 2)

By combining the bounds in the second and third steps, the proof is completed. □

In the proof of Lemma 9, we devise a novel approach to using the properties of the Gaussian process and its derivative surfaces. It is one of the major technical contributions of this paper, and it may be used in other applications of Gaussian process regression as well.

Now, with Lemmas 7–9, we are ready to establish the rate of convergence of GPRS algorithms. It is formally stated in Theorem 2.

Theorem 2. *Suppose that Assumptions 2 and 4–6 hold and that the correlation function of the imposed Gaussian process satisfies $\rho(\boldsymbol{x},\boldsymbol{x}') \ge 1 - C_r\|\boldsymbol{x} - \boldsymbol{x}'\|^\eta$ for $\boldsymbol{x}' \in \mathcal{S}(\boldsymbol{x}, r_n)$ with constants $C_r > 0$ and $0 < \eta \le 2$. If a GPRS algorithm is used to solve Problem (1), then there exists a constant $C_0 > 0$ such that*

$$\mathbb{P}\left\{|\mu_n(\boldsymbol{x}_n^*) - g^*| > \left(\frac{16C_0\log n}{n^{\kappa(n)}}\right)^{1/2}\right\} \to 0 \tag{9}$$

as $n \to \infty$, where $\kappa(n)$ is defined in Equation (8).

Proof. For any $n \ge 1$, define

$$\begin{aligned}
A_0 &= \{|\mu_n(\boldsymbol{x}_n^*) - g^*| > \epsilon_n\},\\
A_1 &= \{|\mu_n(\boldsymbol{x}_n^*) - g(\boldsymbol{x}_n^*)| > \epsilon_n/2\},\\
A_2 &= \{|\mu_n(\boldsymbol{x}_i) - g(\boldsymbol{x}_i)| > \epsilon_n/2 \text{ for some } \boldsymbol{x}_i \in \{\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n\}\},\\
A_3 &= \{g^* - g(\boldsymbol{x}_i) > \epsilon_n/2 \text{ for all } \boldsymbol{x}_i \in \{\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n\}\}.
\end{aligned}$$

First, it is easy to see that if A_0 happens, then $A_1 \cup A_2 \cup A_3$ must happen. Suppose none of A_1 A_2, A_3 happen, and then, $A_1^c \cap A_2^c \cap A_3^c$ happens. Because both A_0 and A_1^c happen, we must have $g^* - \mu_n(\boldsymbol{x}_n^*) > \epsilon_n$; otherwise, we conclude that $g^* - g(\boldsymbol{x}_n^*) < -\epsilon_n/2$, which contradicts the definition of g^*. Notice that A_2^c implies that $|\mu_n(\boldsymbol{x}_i) - g(\boldsymbol{x}_i)| \le \epsilon_n/2$ for all $\boldsymbol{x}_i$, $i = 1,\ldots,n$, and A_3^c implies that $g^* - g(\boldsymbol{x}_i) \le \epsilon_n/2$ for some $\boldsymbol{x}_i$, say $\boldsymbol{x}_1$. They together imply that $|g^* - \mu_n(\boldsymbol{x}_1)| \le \epsilon_n$. Recall that $g^* - \mu_n(\boldsymbol{x}_n^*) > \epsilon_n$, so we have $\mu_n(\boldsymbol{x}_1) - \mu_n(\boldsymbol{x}_n^*) > 0$, but it contradicts to the definition of $\boldsymbol{x}_n^*$.

Based on the observations, we then have

$$\begin{aligned}
&\mathbb{P}\{|\mu_n(\boldsymbol{x}_n^*) - g^*| > \epsilon_n\}\\
&\le \mathbb{P}\{A_1 \cup A_2 \cup A_3\} \le \mathbb{P}\{A_1\} + \mathbb{P}\{A_2\} + \mathbb{P}\{A_3\}\\
&= \mathbb{P}\{|\mu_n(\boldsymbol{x}_n^*) - g(\boldsymbol{x}_n^*)| > \epsilon_n/2\}\\
&\quad + \mathbb{P}(\cup_{i=1}^{n}\{|\mu_n(\boldsymbol{x}_i) - g(\boldsymbol{x}_i)| > \epsilon_n/2\})\\
&\quad + \mathbb{P}(\cap_{i=1}^{n}\{g^* - g(\boldsymbol{x}_i) > \epsilon_n/2\})\\
&\le \mathbb{P}\{|\mu_n(\boldsymbol{x}_n^*) - g(\boldsymbol{x}_n^*)| > \epsilon_n/2\}\\
&\quad + \sum_{i=1}^{n}\mathbb{P}\{|\mu_n(\boldsymbol{x}_i) - g(\boldsymbol{x}_i)| > \epsilon_n/2\}\\
&\quad + \mathbb{P}(\cap_{i=1}^{n}\{g^* - g(\boldsymbol{x}_i) > \epsilon_n/2\}).
\end{aligned} \tag{10}$$

For any $\boldsymbol{x} \in \mathcal{X}$, by Lemma 7, with probability 1, $k_n(\boldsymbol{x},\boldsymbol{x}) \le C_0 n^{-\kappa(n)}$ for some $C_0 > 0$. Then, by Lemma 8,

$$\begin{aligned}
\mathbb{P}\{|\mu_n(\boldsymbol{x}) - g(\boldsymbol{x})| > \epsilon_n/2\} &\le 2\mathbb{E}[e^{-\epsilon_n^2/(8k_n(\boldsymbol{x},\boldsymbol{x}))}]\\
&\le 2e^{-\frac{1}{8C_0}\epsilon_n^2 n^{\kappa(n)}}.
\end{aligned} \tag{11}$$

Combining Equations (10) and (11) and Lemma 9 yields

$$\mathbb{P}\{|\mu_n(\boldsymbol{x}_n^*) - g^*| > \epsilon_n\}$$

$$\le 2(n+1)e^{-\frac{1}{8C_0}\epsilon_n^2 n^{\kappa(n)}} + e^{-\frac{\alpha C\pi^{d/2}(\epsilon_n/2)^d n}{\Gamma(d/2+1)\log n}} + 2\sum_{j=1}^{d} e^{C_j\left(\frac{(\log n)^{1/d}}{\sqrt{d}}-a_j\right)-\frac{\left(\frac{(\log n)^{1/d}}{\sqrt{d}}-a_j\right)^2}{2\sigma_j^2}}. \tag{12}$$

Now, let $\epsilon_n = \left(\frac{16C_0\log n}{n^{\kappa(n)}}\right)^{1/2}$. Notice that

$$2(n+1)e^{-\frac{1}{8C_0}\epsilon_n^2 n^{\kappa(n)}} = \frac{2(n+1)}{n^2} \to 0 \text{ as } n \to \infty$$

and

$$e^{-\frac{\alpha C\pi^{d/2}(\epsilon_n/2)^d n}{\Gamma(d/2+1)\log n}} = e^{-\frac{\alpha C(4C_0\pi)^{d/2}(\log n)^{d/2}}{\Gamma(d/2+1)\log n}\cdot n^{1-d\kappa(n)/2}} \to 0 \text{ as } n \to \infty$$

because $1 - d\kappa(n)/2 > 0$ by the definition of $\kappa(n)$ with $0 < \eta \le 2$ in Equation (8). Moreover, it is easy to see that

$$e^{C_j\left(\frac{(\log n)^{1/d}}{\sqrt{d}}-a_j\right)-\frac{\left(\frac{(\log n)^{1/d}}{\sqrt{d}}-a_j\right)^2}{2\sigma_j^2}} \to 0 \text{ as } n \to \infty.$$

Therefore, it follows that, as $n \to \infty$,

$$\mathbb{P}\left\{|\mu_n(\boldsymbol{x}_n^*) - g^*| > \left(\frac{16C_0\log n}{n^{\kappa(n)}}\right)^{1/2}\right\} \to 0.$$

This concludes the proof of the theorem. □

Theorem 2 shows that the optimal value of the conditional mean function $\mu_n(\boldsymbol{x}_n^*)$ converges to the global optimal value g^* no slower than $O_p(n^{-\kappa(n)/2})$ when the Gaussian process satisfies Assumption 5 and its correlation function satisfies $\rho(\boldsymbol{x},\boldsymbol{x}') \ge 1 - C_r\|\boldsymbol{x}-\boldsymbol{x}'\|^\eta$ for $\boldsymbol{x}' \in \mathcal{S}(\boldsymbol{x}, r_n)$ with constants $C_r > 0$ and $0 < \eta \le 2$. As discussed in Section 4.1, many commonly used Gaussian processes (e.g., the Gaussian processes having Gaussian correlation function, rational quadratic correlation function, and Matérn correlation function with the smoothness parameter v being a half-integer and greater than one) satisfy these conditions with $\eta = 2$. Hence, the upper bound of the rate of convergence of GPRS algorithms can be $\tilde{O}_p(n^{-1/(d+2)})$. This rate is the maximum allowable decreasing rate of ϵ_n such that both the probability that the prediction error of $\mu_n(\boldsymbol{x})$ is greater than $\epsilon_n/2$ and the probability that no point is ever sampled near one of the global optimal solutions (with radius determined by $\epsilon_n/2$) still tend to zero. Based on the established probability bounds in Lemmas 8 and 9, by letting $\epsilon_n = (16C_0\log n/n^{\kappa(n)})^{1/2}$, the rate of convergence can be obtained. Notice that this rate is an upper bound for the rate of convergence of the whole class of GPRS algorithms, with various kinds of sampling distributions. This implies that the lower bound of the sampling densities (i.e, Assumption 4) plays a critical role in establishing the rate of convergence. For GPRS algorithms that have adaptive sampling distributions, their finite-sample performance may be better than this theoretical rate.

It is worth emphasizing again that the rate here is not for a specific objective function as typically in the literature. Notice that we may interpret the probability statement in Equation (9) as

$$\mathbb{P}\left\{|\mu_n(\boldsymbol{x}_n^*) - g^*| > \left(\frac{16C_0\log n}{n^{\kappa(n)}}\right)^{1/2}\right\} = \mathbb{E}\left[\mathbb{P}\left\{|\mu_n(\boldsymbol{x}_n^*) - g^*| > \left(\frac{16C_0\log n}{n^{\kappa(n)}}\right)^{1/2}\middle| g\right\}\right],$$

where the expectation is taken with respect to the distribution of the random objective function g. Therefore, the rate of convergence in Theorem 2 may be viewed as the average rate of convergence (in the typical sense) of all objective functions that are sampled randomly from the Gaussian process that satisfies Assumption 5.

To understand the upper bound of the rate of convergence, we compare it with other rate of convergence results in the literature. The first is the EGO algorithm of Jones et al. (1998), which similar to GPRS algorithms, uses Gaussian processes to guide the searches to solve deterministic black-box optimization problems. It was proved by Bull (2011) that the rate of convergence (to the global optimum) of the EGO algorithm is at least $O_p(n^{-1/d})$ for functions in the reproducing-kernel Hilbert space of the chosen Gaussian process. Compared with this result, we have two findings. First, the dimension d has a common and significant impact on the rate of convergence of Gaussian process-based algorithms. Second, GPRS algorithms for COvS problems converge slightly slower than the EGO algorithm. However, we think that such difference may be caused by the existence of simulation noises that make the search and estimation more difficult.

Next, we consider simple random search algorithms, such as the ones of Yakowitz et al. (2000) and Chia and Glynn (2013). Even though their practical performances are typically not competitive, they often reveal important insights on the asymptotic properties, such as convergence and rate of convergence. The random search algorithm with low-dispersion point sets of Yakowitz et al. (2000) solves the COvS problems using a multiobservation approach. The upper bound of its rate of convergence is $O_p((n/\log n)^{-q/(d+2q)})$ or $\tilde{O}_p(n^{-q/(d+2q)})$, where the definition of the rate of convergence is similar to ours in this paper and q is a parameter that measures the local Lipschitz condition of the objective function g around the global optimal solution $\boldsymbol{x}^*$. Specifically, q satisfies $\sup_{x\in\mathcal{S}(\boldsymbol{x}^*,t)}(g(\boldsymbol{x}^*) - g(\boldsymbol{x})) \le Kt^q$ for $t \le t_0$ for some positive constants t_0 and K. When $q = 1$, the local Lipschitz condition implies that the first-order derivative of g around $\boldsymbol{x}^*$ is bounded, and the global optimal solution can be a boundary point. When $q = 2$, the local Lipschitz condition implies that g is locally quadratic around $\boldsymbol{x}^*$, and the global optimal solutions

are in the interior of the feasible set. In the analysis of the rate of convergence of GPRS algorithms, we do not differentiate whether the global optimal solution is an interior point or not, and we assume continuous differentiability of the sample paths (of the impose Gaussian process), which corresponds to the Lipschitz condition with $q = 1$. In this situation, the upper bounds of rate of convergence of the both algorithms are $\tilde{O}_p(n^{-1/(d+2)})$. Actually, under a similar regularity condition, Donoho (1994) proves that the optimal rate of convergence for nonparametric regression is $O(n^{-1/3})$ in one-dimensional space, which implies that our proved upper bound is nearly tight in the one-dimensional problem.

When $q = 2$, the rate of convergence of the random search algorithm with low-dispersion point sets is $\tilde{O}_p(n^{-2/(d+4)})$. Chia and Glynn (2013) prove the same rate of convergence of pure random search algorithm under similar regularity conditions (i.e., the objective function is three times continuously differentiable and has an negative definite Hessian matrix at the unique interior maximizer). This rate of convergence is slightly better than ours, which implies that the upper bound of the rate of convergence of GPRS algorithms may be improved if the sample paths are smoother (e.g., the Gaussian process with Gaussian correlation function). However, because of the probability inequalities that are used to deal with the Gaussian processes, we have not yet found an approach to establishing this. We leave this as a topic for future research.

4.3. Revised GPRS Algorithms

In this subsection we propose a slightly revised version of GPRS algorithms, which improve the computational efficiency of the original GPRS algorithms without impacting its rate of convergence. This is achieved by replacing the original step 2 with the following step 2′.

Step 2′ (calculation). Set $n = rs$. Let $\boldsymbol{X}^n = \boldsymbol{X}^{r(s-1)} \cup \{\boldsymbol{x}_{r(s-1)+1}, \ldots, \boldsymbol{x}_{rs}\}$ and $\boldsymbol{G}^n = ([\boldsymbol{G}^{r(s-1)}]^\top, G(\boldsymbol{x}_{r(s-1)+1}), \ldots, G(\boldsymbol{x}_{rs}))^\top$. Calculate $\mu_n(\boldsymbol{x})$ according to Equation (2).

Let $\boldsymbol{x}_n^\dagger = \arg\max_{\boldsymbol{x}\in\boldsymbol{X}^n} \mu_n(\boldsymbol{x})$, and break the tie arbitrarily if it exists. Then, construct the sampling distribution $f_n(\boldsymbol{x})$ according to user-specified rules.

The main difference of this revised algorithm to the original one is to replace the function best solution $\boldsymbol{x}_n^*$ with the sample best solution $\boldsymbol{x}_n^\dagger$. In this way, the revised algorithm can avoid calculating $\boldsymbol{x}_n^* = \arg\max_{\boldsymbol{x}\in\mathcal{X}} \mu_n(\boldsymbol{x})$ repeatedly, which can significantly reduce the computation overhead of the algorithm (the associated optimization problem is often nonconvex and difficult), especially when the dimension d is large. Although the finite-sample performance of the revised algorithm may differ, because of the different way to output the current solution (especially in the early iterations), it can be shown that the rate of convergence is not affected.

The rate of convergence of the revised algorithm can be proved by following the same steps as in this section. Here, we omit the detailed analysis and only provide a sketch of the proof. Let $\boldsymbol{x}_n^\dagger = \arg\max_{\boldsymbol{x}\in\boldsymbol{X}^n} \mu_n(\boldsymbol{x})$. With the same A_2 and A_3 as defined in the proof of Theorem 2, one can have $\mathbb{P}\{|\mu_n(\boldsymbol{x}_n^\dagger) - g^*| > \epsilon_n\} \le \mathbb{P}\{A_2\} + \mathbb{P}\{A_3\}$. Following the same arguments as in the proof of Theorem 2, we can show that the convergence rate of $\mu_n(\boldsymbol{x}_n^\dagger)$ is the same as $\mu_n(\boldsymbol{x}_n^*)$, which completes the proof.

Notice that the revised algorithm requires Assumption 5 to ensure both the convergence and the rate of convergence. When using the sample best solution $\boldsymbol{x}_n^\dagger$, to prove the almost-sure global convergence of the revised algorithm, we need to establish the probability bound of the event that no point is ever sampled near the global optimal solutions and then, use the Borel–Cantelli lemma to prove that such an event does not happen infinitely. According to the proof of Lemma 9, this requires the boundedness of the derivative surfaces of the Gaussian process. However, for general Gaussian processes that satisfy Assumption 3 but not Assumption 5, such a property may not hold, and the revised algorithm may not guarantee the convergence.

5. An Example of GPRS Algorithms: The GPS-C Algorithm

In this section, we introduce a specific GPRS algorithm and use it as an example to discuss the design and implementation of a GPRS algorithm. This algorithm is an extension of the GPS algorithm of Sun et al. (2014), which is originally for DOvS problems, and it is called the GPS-C algorithm. Based on the framework of GPRS algorithms in Section 2.2, the GPS-C algorithm combines the variance of the conditional Gaussian process $k_n(\boldsymbol{x}, \boldsymbol{x})$ with $\mu_n(\boldsymbol{x})$ to construct adaptive sampling distributions, and it achieves a seamless integration of estimation, exploration, and exploitation. Hence, this algorithm is a single-observation integrated COvS algorithm. In the following subsections, we will describe the sampling distribution of the GPS-C algorithm and formally give the GPS-C algorithm. Then, we will discuss several implementation issues of the algorithm briefly.

5.1. The GPS-C Algorithm

Sampling distribution is a key element of a random search algorithm, which is constructed in each iteration to determine where to allocate the simulation effort. As shown by Sun et al. (2014), their sampling distribution constructed based on the Gaussian process can adaptively balance the trade-off between exploration and exploitation when solving DOvS problems. For COvS problems, we can construct the sampling distributions in a similar way.

From Equations (2) and (3), for any $\boldsymbol{x} \in \mathcal{X}$, we have

$$f_{\mathcal{GP}}(\boldsymbol{x}) | \{\boldsymbol{X}^n, \boldsymbol{G}^n\} \sim \mathcal{N}(\mu_n(\boldsymbol{x}), k_n(\boldsymbol{x}, \boldsymbol{x})). \tag{13}$$

Then, we may define a probability density function $f_n(\boldsymbol{x})$ as follows:

$$f_n(\boldsymbol{x}) = \frac{\mathbb{P}\{Z(\boldsymbol{x}) > c\}}{\int_{\mathcal{X}} \mathbb{P}\{Z(\boldsymbol{z}) > c\}\, \mathrm{d}\boldsymbol{z}}, \quad \boldsymbol{x} \in \mathcal{X}, \tag{14}$$

where $Z(\boldsymbol{x}) \sim \mathcal{N}(\mu_n(\boldsymbol{x}), k_n(\boldsymbol{x}, \boldsymbol{x}))$, $c = \max_{\boldsymbol{x} \in \mathcal{X}} \mu_n(\boldsymbol{x})$, and $\boldsymbol{z} \in \mathbb{R}^d$. Notice that c is well defined under Assumptions 1 and 3, which imply that $\mathcal{X}$ is compact and $\mu_n(\boldsymbol{x})$ is continuous. This sampling distribution maintains the desirable properties of the sampling distribution of the GPS algorithm of Sun et al. (2014). (1) It assigns higher probabilities to regions that contain good solutions (because of higher conditional mean). (2) It assigns higher probabilities to less explored regions (because of higher conditional variance). Therefore, it can balance the trade-off between exploration and exploitation adaptively. We note that similar sampling methods that utilize the probabilistic prediction (instead of sole mean prediction) of the Gaussian process surrogate model are also used in the Bayesian optimization algorithm (i.e., the P algorithm (Žilinskas 1985, Calvin and Žilinskas 1999)) and related reliability analysis researches (Dubourg et al. 2011, 2013).

Furthermore, as shown by Sun et al. (2014), it is easy to ensure the density to satisfy the requirement of Assumption 4. Specifically, the user may specify a proper lower bound and upper bound for $\mu_n(\boldsymbol{x})$, say $\underline{M}$ and $\overline{M}$, and a lower bound for $k_n(\boldsymbol{x}, \boldsymbol{x})$, say $\underline{\tau}^2$ with $\underline{\tau} > 0$. Then, we can define $k_n^{\mathrm{cap}}(\boldsymbol{x}, \boldsymbol{x}) = \max\{\underline{\tau}^2, k_n(\boldsymbol{x}, \boldsymbol{x})\}$ and

$$\mu_n^{\mathrm{cap}}(\boldsymbol{x}) = \begin{cases} \underline{M}, & \text{if } \mu_n(\boldsymbol{x}) < \underline{M}, \\ \overline{M}, & \text{if } \mu_n(\boldsymbol{x}) > \overline{M}, \\ \mu_n(\boldsymbol{x}), & \text{otherwise.} \end{cases}$$

The density of the sampling distribution used in the algorithm is given by

$$f_n(\boldsymbol{x}) = \frac{\mathbb{P}\{Z(\boldsymbol{x}) > c\}}{\int_{\mathcal{X}} \mathbb{P}\{Z(\boldsymbol{z}) > c\}\, \mathrm{d}\boldsymbol{z}}, \quad \boldsymbol{x} \in \mathcal{X}, \tag{15}$$

where $Z(\boldsymbol{x}) \sim \mathcal{N}(\mu_n^{\mathrm{cap}}(\boldsymbol{x}), k_n^{\mathrm{cap}}(\boldsymbol{x}, \boldsymbol{x}))$, and $c = \max_{\boldsymbol{x} \in \mathcal{X}} \mu_n^{\mathrm{cap}}(\boldsymbol{x})$. It is not difficult to see that $f_n(\boldsymbol{x})$ has a lower bound on $\mathcal{X}$, which is explicitly stated in Lemma 10, whose proof is provided in Section EC.2.2 of the e-companion.

Lemma 10. *Let* $\alpha = 2[1 - \Phi((\overline{M} - \underline{M})/\underline{\tau})]/\nu(\mathcal{X}) > 0$, *where* Φ *is the cumulative distribution function of the standard normal random variable, and* $\nu(\mathcal{X}) = \int_{\mathcal{X}} \mathrm{d}\boldsymbol{z}$ *for* $\boldsymbol{z} \in \mathbb{R}^d$ *is the volume of* $\mathcal{X}$. *Then,* $f_n(\boldsymbol{x}) \geq \alpha$ *for all* $\boldsymbol{x} \in \mathcal{X}$.

Lemma 10 ensures that the sampling density function defined in Equation (15) is bounded from below on $\mathcal{X}$, which satisfies Assumption 4.

With the adaptive sampling distribution $f_n(\boldsymbol{x})$, we can formally describe the GPS-C algorithm as follows using the parameters of GPRS algorithms defined in Section 2.2.

Step 0 (initialization). Impose a Gaussian process with μ_0 and k_0 that satisfy Assumption 3. Specify a $r > 0$, $\underline{\tau} > 0$, $\underline{M}$, and $\overline{M}$. Set $s = 0$, $n = 0$, $\boldsymbol{X}^0 = \emptyset$, and $\boldsymbol{G}^0 = \emptyset$. Furthermore, set $f_0(\boldsymbol{x})$ as a user-specified sampling distribution over $\mathcal{X}$.

Step 1 (sampling). Set $s = s + 1$. Sample $\boldsymbol{x}_{r(s-1)+1}, \ldots, \boldsymbol{x}_{rs}$ independently from $f_n(\boldsymbol{x})$, and obtain corresponding simulation observations $G(\boldsymbol{x}_{r(s-1)+1}), \ldots, G(\boldsymbol{x}_{rs})$ independently from all previous observations.

Step 2 (calculation). Set $n = rs$. Let $\boldsymbol{X}^n = \boldsymbol{X}^{r(s-1)} \cup \{\boldsymbol{x}_{r(s-1)+1}, \ldots, \boldsymbol{x}_{rs}\}$ and $\boldsymbol{G}^n = ([\boldsymbol{G}^{r(s-1)}]^\top, G(\boldsymbol{x}_{r(s-1)+1}), \ldots, G(\boldsymbol{x}_{rs}))^\top$. Calculate $\mu_n(\boldsymbol{x})$ and $k_n(\boldsymbol{x}, \boldsymbol{x})$ according to Equations (2) and (3). Let $\boldsymbol{x}_n^* = \arg\max_{\boldsymbol{x} \in \mathcal{X}} \mu_n(\boldsymbol{x})$, and break the tie arbitrarily if it exists. Then, construct the sampling distribution $f_n(\boldsymbol{x})$ according to Equation (15).

Step 3 (stopping). If the stopping condition is not met, go to step 1; otherwise, stop and output $\boldsymbol{x}_n^*$ and $\mu_n(\boldsymbol{x}_n^*)$ as the estimated optimal solution and the estimated optimal objective value.

The GPS-C algorithm is an example of an integrated GPRS algorithm. It well utilizes the Gaussian process surrogate model to construct adaptive sampling distributions (to balance exploration and exploitation) and can be expected to have good finite-sample performance. However, the GPS-C algorithm is still a theoretical version. There are still several gaps for us to implement this algorithm in practice. (1) The parameters of the Gaussian process may be difficult to determine in advance. (2) The variances of the simulation noise are unknown. (3) An efficient sampling scheme, which samples design points from the sampling distribution, is required. (4) An efficient method, which can solve the optimization problem $\boldsymbol{x}_n^* = \arg\max_{\boldsymbol{x} \in \mathcal{X}} \mu_n(\boldsymbol{x})$, is required. In the following subsection, we provide some effective approximation methods to address these implementation issues.

5.2. The Implementation of the GPS-C Algorithm

5.2.1. Estimation of the Variances and the Gaussian Process Parameters. The need to estimate the Gaussian process parameters and the unknown variances of the simulation noises is common in kriging-based Bayesian optimization algorithms (e.g., the SKO algorithm of Huang et al. 2006 and the KGCP algorithm of Scott et al. 2011). There are in general two ways to estimate them under different contexts. For homoscedastic simulation noises, one common way is to use the MLE method to estimate the Gaussian parameters μ_0, τ^2, and θ, together with the variance λ^2. The readers can refer to Stein (1999) and Huang et al. (2006) for more information about this method. In a typical kriging-based Bayesian optimization algorithm, the variance and the Gaussian process parameters are updated in each iteration of the algorithm based on current observations. However, in the GPS-C algorithm, we do not update these parameters at each iteration. Instead, the users may use the MLE method to estimate these parameters

at the initialization stage or in the first few iterations. This is because the GPS-C algorithm, as a random search-based algorithm, usually requires large simulation budgets to solve optimization problems. The choice of not updating these parameters repeatedly can alleviate the computational burden because the matrix inversion in the MLE method may become prohibitive with more and more sampled points (Sun et al. 2014).

For heteroscedastic simulation noises, a common way is to estimate the unknown variances first and then use an MLE method to estimate the Gaussian process parameters based on current observations and the estimated variances. The readers can refer to Ankenman et al. (2010) for more information about this approach. We also adopt the two-step approach in this paper. To estimate the unknown variances under the heteroscedastic context, multiple observations are typically required at each sampled point, and the variances are estimated using the conventional sample variance estimator. However, in the GPS-C algorithm, only one observation is taken at each sampled point. To estimate the unknown variances, a kernel-based sample variance estimator is proposed in this paper. Define a uniform kernel function $K(\boldsymbol{u}) = \frac{1}{(2h)^d}\mathbf{1}_{\{\|\boldsymbol{u}/h\|_\infty \le 1\}}$ in $\mathbb{R}^d$, where h is the single bandwidth of the kernel. With the observations $\{\boldsymbol{X}^n, \boldsymbol{G}^n\}$, the unknown variance $\lambda^2(\boldsymbol{x})$ can be estimated using

$$\hat{\lambda}^2(\boldsymbol{x}) = \frac{1}{k}\sum_{i=1}^{k}(G_i - \hat{m}_{NW}(\boldsymbol{x}))^2, \tag{16}$$

where k is the number of sampled points within the kernel $\|(\boldsymbol{x}' - \boldsymbol{x})/h\|_\infty \le 1$ centered at $\boldsymbol{x}$ and $\hat{m}_{NW}(\boldsymbol{x})$ is the classical Nadaraya–Watson (NW) estimator. This NW estimator is used to estimate the unknown objective function and is defined as

$$\hat{m}_{NW}(\boldsymbol{x}) = \frac{\sum_{i=1}^{n} G_i K(\boldsymbol{x} - \boldsymbol{x}_i)}{\sum_{i=1}^{n} K(\boldsymbol{x} - \boldsymbol{x}_i)}.$$

It is not difficult to prove that as the number of sampled points n goes to infinity, this kernel-based sample variance estimator is asymptotically consistent with a decreasing bandwidth h_n to zero. Interested readers can refer to Schimek (2013) for more information.

Based on the estimated variance $\hat{\lambda}^2(\boldsymbol{x}_i)$ at each $\boldsymbol{x}_i \in \boldsymbol{X}^n$, the Gaussian process parameters μ_0, τ^2, and θ can be estimated using the MLE method (Ankenman et al. 2010). Under the heteroscedastic context, we can also estimate and update the Gaussian process parameters only in the initialization stage or early iterations of the GPS-C algorithm. The variances need to be updated in each iteration after obtaining new observations. By replacing the Gaussian process parameters and the variance with their estimated counterparts in Equations (2) and (3), we can use

$$\hat{\mu}_n(\boldsymbol{x}) = \hat{\mu}_0(\boldsymbol{x}) + \hat{k}_0(\boldsymbol{x}, \boldsymbol{X}^n)[\hat{k}_0(\boldsymbol{X}^n, \boldsymbol{X}^n) + \hat{\Sigma}^n]^{-1}[\boldsymbol{G}^n - \hat{\mu}_0(\boldsymbol{X}^n)], \tag{17}$$

$$\hat{k}_n(\boldsymbol{x}, \boldsymbol{x}') = \hat{k}_0(\boldsymbol{x}, \boldsymbol{x}') - \hat{k}_0(\boldsymbol{x}, \boldsymbol{X}^n)[\hat{k}_0(\boldsymbol{X}^n, \boldsymbol{X}^n) + \hat{\Sigma}^n]^{-1}\hat{k}_0(\boldsymbol{X}^n, \boldsymbol{x}') \tag{18}$$

to estimate the conditional mean and the conditional variance at each $\boldsymbol{x} \in \mathcal{X}$. Based on these values, we can construct sampling distributions using the same way as discussed in Section 5.1.

5.2.2. Sampling Scheme. In random search-based algorithms, it is not always straightforward to sample design points from the sampling distributions. Notice that the explicit form of $f_n(\boldsymbol{x})$ of the GPS-C algorithm is typically not applicable because the denominator involves an integration that is computationally expensive to calculate. Therefore, we propose two sampling algorithms to sample design points approximately. These two algorithms are the extensions of the sampling schemes of Sun et al. (2014) to the continuous context. Because of space limitations, we provide these two sampling schemes in Section EC.4 of the e-companion.

5.2.3. Finding the Best Solution. The last implementation issue to address is that the GPS-C algorithm needs to find the optimal solution of the Gaussian surrogate model (i.e., $\hat{\boldsymbol{x}}_n^* = \arg\max_{x\in\mathcal{X}} \hat{\mu}_n(\boldsymbol{x})$) to construct the sampling distribution in each iteration of the algorithm. Such an issue is common for surrogate-based optimization algorithms. For example, the EGO algorithm of Jones et al. (1998), the SKO algorithm of Huang et al. (2006), and the KGCP algorithm of Scott et al. (2011) all need to solve some optimization problem to determine the next sampling decision, and the metamodel-based optimization algorithm of Osorio and Bierlaire (2013) needs to solve the optimal solution of the metamodel to guide the sampling. Additionally, for all these surrogate-based optimization algorithms, the optimal solution of the surrogate model needs to be reported when the algorithm terminates or even during the iterations (for monitoring the performance of the algorithm). Typically, these optimization problems are nonconvex and are computationally difficult to solve. In the GPS-C algorithm, we offer some methods to deal with this issue. If the imposed Gaussian process satisfies Assumption 5, we can use the revised algorithm and solve the sample best solution (i.e., $\hat{\boldsymbol{x}}_n^\dagger = \arg\max_{x\in\boldsymbol{X}^n} \hat{\mu}_n(\boldsymbol{x})$) for both constructing the sampling distribution and reporting the current solution. If Assumption 5 is not satisfied, we can solve $\hat{\boldsymbol{x}}_n^* = \arg\max_{x\in\mathcal{X}} \hat{\mu}_n(\boldsymbol{x})$ approximately using the following two approaches. When the dimension is low (e.g., d is small), one may simply evaluate $\hat{\mu}_n(\boldsymbol{x})$ on a dense grid within $\mathcal{X}$ and find the optimal solution of the grid. When the dimension is high, one may set $\hat{\boldsymbol{x}}_n^\dagger$ as the initial

solution and use some nonlinear optimization solvers (e.g., the `fmincon` in MATLAB) to find an approximate optimal solution.

Before we show the practical performance of the GPS-C algorithm, the computational complexity of the algorithm is briefly discussed. We note that it is difficult to calculate the overall computational complexity of the algorithm for three reasons. (1) The computational complexity depends on the chosen component of the algorithm (e.g., which sampling scheme is used). (2) Some complex optimization problems need to be solved in the algorithm (e.g., the best solution finding in step 2 and the MLE method used to estimate the Gaussian process parameters). (3) The simulation models of COvS problems in practice can be large and complex, and they contribute greatly to the overall computational complexity. Nevertheless, we can come up with some methods to reduce the computational complexity of certain parts of the algorithm. For instance, to build a Gaussian process surrogate model in each iteration, the inverse of a covariance matrix (i.e., $[k_0(\boldsymbol{X}^n,\boldsymbol{X}^n)+\boldsymbol{\Sigma}^n]^{-1}$) must be calculated, which has an $O(n^3)$ complexity and scales poorly with n. To reduce its computation complexity, we can record the inverse matrix in each iteration and use the block matrix inversion (Bernstein 2009) to calculate the covariance matrix inverse for next iteration. In this way, its computational complexity can be reduced to $O(n^2)$. This method is used in the numerical studies of Section 6, where we show the empirical rate of convergence of the GPS-C algorithm.

6. Numerical Experiments

In this section, we conduct numerical experiments to understand the empirical performance of a specific GPRS algorithm (i.e., the GPS-C algorithm). We first test and verify the theoretical properties of the GPS-C algorithm on problems that are generated from Gaussian processes, and we then compare the GPS-C algorithm with other widely used algorithms (including kriging-based Bayesian optimization algorithms and random search based algorithms) on various test problems. Additionally, the influence of dimensionality on the rate of convergence of the GPS-C algorithm is also investigated. Throughout this section, we adopt two performance measures. One is the estimated objective function value at the estimated optimal solution (i.e., $\hat{\mu}_n(\hat{\boldsymbol{x}}_n^*)$), and the other is the true objective function value at the estimated optimal solution (i.e., $g(\hat{\boldsymbol{x}}_n^*)$). All the numerical experiments are conducted in MATLAB, and the GPS-C algorithm is implemented with the Markov chain coordinate sampling scheme. The detailed parameters of algorithms in each problem are provided in Section EC.5 of the e-companion together with some supplementary figures. The codes used in this section can be found at https://github.com/xiuxianwang/GPS-C-algorithm.

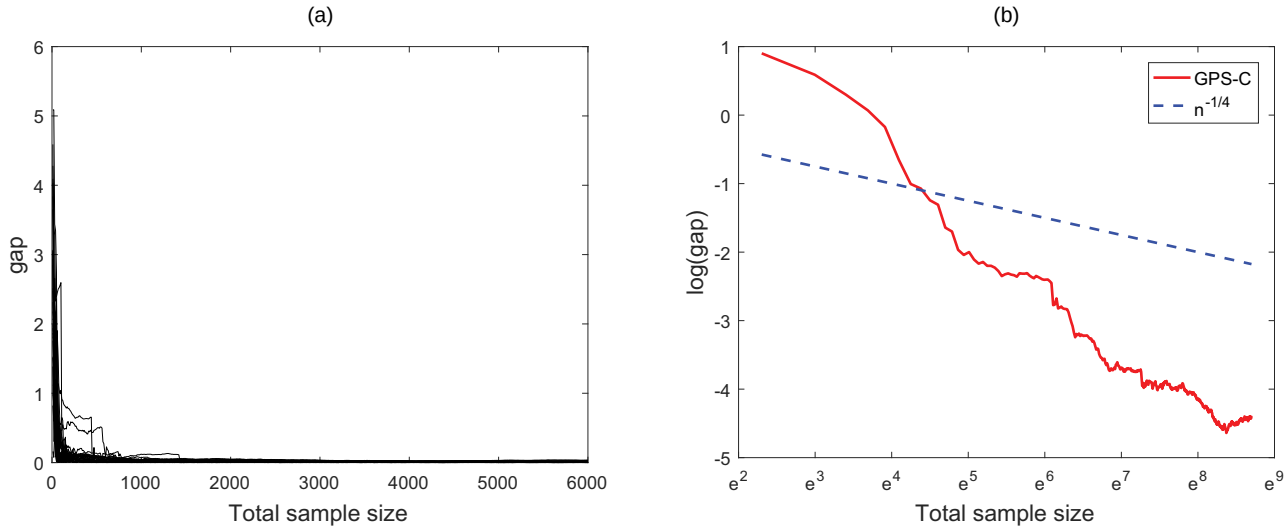
6.1. Empirical Performance on Generated Problems

In this subsection, our goal is to check whether the empirical performances of the GPS-C algorithm are consistent with the theoretical analysis. To fulfill this goal, we do not use a specific objective function but use a number of continuous functions sampled from a Gaussian process (as specified in Assumptions 3 and 5) on the space $[0,1]^d$ to show the empirical convergence and the rate of convergence of the GPS-C algorithm. In addition, the variance of the simulation noise is given as in Assumption 2.

We first let $d=2$. On the two-dimensional space $[0,1]^2$, a Gaussian process having Gaussian correlation function with $\mu_0=1$, $\tau^2=4$, and $\boldsymbol{\theta}=(80,80)$ is used to generate 30 sample paths to represent 30 objective functions, and a random variable following $N(0,0.25)$ is added at any point as the simulation noise. To generate these sample paths efficiently, we take observations of this Gaussian process on a uniform grid containing 400 points and fit the conditional sample path using the stochastic kriging method (Ankenman et al. 2010) with artificial intrinsic noise (normally distributed with mean 0 and variance 0.1) for numerical stability. Such 30 conditional sample paths are used as objective functions, and two examples are shown in Figure EC.1 in the e-companion. Then, we identify the maximum value of each objective function (by evaluating a dense grid with step size 0.01) and run the GPS-C algorithm to search for this maximum value and its corresponding solution. The Gaussian process parameters of the GPS-C algorithm are the same as those used to generate the objective functions, and other parameters are listed in Table EC.1 in the e-companion. In each iteration of the algorithm, we use the same dense grid as before to approximately solve $\hat{\boldsymbol{x}}_n^* = \arg\max_{\boldsymbol{x}\in\mathcal{X}}\hat{\mu}_n(\boldsymbol{x})$ both for constructing the sampling distribution and for reporting the current best solution. Figure 1(a) shows all 30 empirical sample paths of the GPS-C algorithm in terms of the optimality gap $|\hat{\mu}_n(\hat{\boldsymbol{x}}_n^*)-g^*|$ as a function of sample size n. Figure 1(b) shows the average optimality gap with respect to the sample size n on a log-log plot to show the empirical rate of the algorithm. It can be observed that the optimality gaps decrease quickly, and the average gap appears to shrink in a rate faster than $n^{-1/4}$, which is its theoretical rate of convergence.

We also replicate this experiment on a three-dimensional space $[0,1]^3$ with the parameters of the Gaussian process being $\mu_0=1$, $\tau^2=9$, and $\boldsymbol{\theta}=(40,40,40)$. Figure 2 shows similar results. Because of the numerical difficulties in generating sample paths from Gaussian process in higher dimensions, we cannot try problems with $d>3$.

Figure 1. (Color online) Empirical Performance of the GPS-C Algorithm on Two-Dimensional Problems



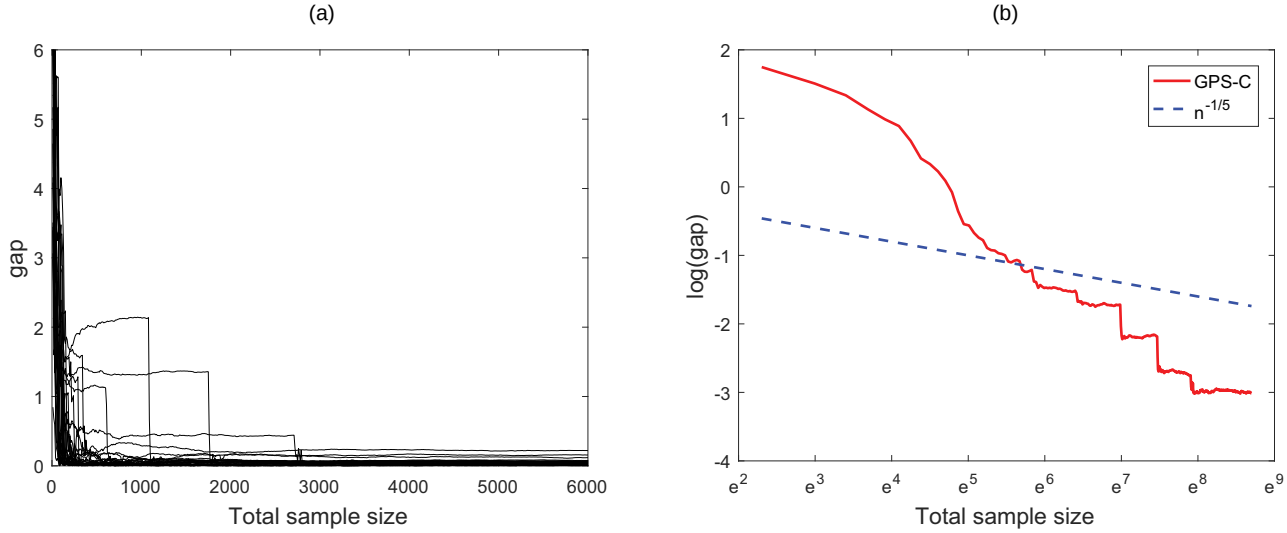
Notes. (a) The optimality gap. (b) The rate of convergence.

We leave the discussion on the influence of dimensionality to Section 6.3.

6.2. Empirical Performance on Various Test Problems

In this subsection, our goal is to check the empirical performance of the GPS-C algorithm on problems that are not generated from Gaussian processes. For comparison, two kriging-based Bayesian optimization algorithms (i.e., the SKO algorithm of Huang et al. 2006 and the KGCP algorithm of Scott et al. 2011) and three random search-based algorithms (i.e., the ASR algorithm of Andradóttir and Prudius 2010 and the IHR-SO and AP-SO algorithms of Kiatsupaibul et al. 2018) are also implemented. In their corresponding papers, the kriging-based Bayesian optimization algorithms and the random search-based algorithms are implemented and tested using problems with homoscedastic and heteroscedastic noises, respectively. Hence, we add simulation noises with either equal variances or unequal variances to the objective functions in the comparison with each type of algorithms. The GPS-C algorithm is first compared with the SKO and KGCP algorithms on problems with homoscedastic noises in Section 6.2.1, and it is then compared with the ASR, IHR-SO, and AP-SO algorithms on problems with heteroscedastic noises in Section 6.2.2.

Figure 2. (Color online) Empirical Performance of the GPS-C Algorithm on Three-Dimensional Problems



Notes. (a) The optimality gap. (b) The rate of convergence.

6.2.1. Optimization Problems with Homoscedastic Noises. The GPS-C, SKO, and KGCP algorithms are tested and compared on three two-dimensional COvS problems with homoscedastic noises. Two of them are the Branin problem and the six-hump problem, which are used in Huang et al. (2006) and Scott et al. (2011). The other one is a problem with multiple local optima, whose discrete version is used in Sun et al. (2014) to investigate the performance of the GPS algorithm for DOvS problems. Because this problem has 25 local optima and looks like hills, we call it the "Hills" problem for short. The expressions of the three problems are given in Section EC.5 of the e-companion, and their optimal values and solutions are summarized in Table EC.2 therein. Simulation noises with distribution $N(0, \lambda^2)$ are added, and different values of λ^2 are considered.

For each problem, the three algorithms all start with an initial Latin hypercube sampling (LHS) design of 20 points. The Gaussian process parameters and the variances of noises used in the three algorithms are estimated and updated until the sample size reaches 100 (for the Branin and six-hump problems) or 300 (for the Hills problem). The other parameters of GPS-C algorithm are listed in Table EC.3 in the e-companion. Note that as mentioned in Section 5.2.3, all three algorithms need to solve some optimization problems on their domains for sampling in each iteration and reporting current solutions. For efficient computation and fair comparison, we adopt the exact same method to solve these optimization problems for the three algorithms. We first evaluate the corresponding objective function on a grid (with step sizes 0.3, 0.08, and 2 for the Branin, six-hump, and Hills problems, respectively), and then, we conduct local search starting from the optimal solution found on the grid using fmincon in MATLAB. The performances of the algorithms are measured using the absolute gap between $g(\hat{x}_n^*)$ and g^*, and we run each algorithm 30 times to calculate the mean performance.

Table 1 shows the performances of the SKO algorithm, the KGCP algorithm, and the GPS-C algorithm up to different sample sizes. It can be observed that all three algorithms perform well on these optimization problems. Comparatively, the SKO algorithm and the KGCP algorithm perform better than the GPS-C algorithm in the early stage. However, when sample size reaches 800, the performance of the GPS-C algorithm appears better than the other two algorithms in most cases. This comparison shows different features of the kriging-based Bayesian optimization algorithms and the GPS-C algorithm, which is a random search-based algorithm.

6.2.2. Optimization Problems with Heteroscedastic Noises. In this part, we test the performances of the GPS-C, ASR, IHR-SO, and AP-SO algorithms and compare them on two COvS problems with heteroscedastic noises. The first one is the 2-dimensional Hills problem used in Section 6.2.1, and the second one is the 10-dimensional Rosenbrock problem. The detailed information of the Rosenbrock problem is given in Section EC.5 of the e-companion. Compared with the Rosenbrock problem used by Kiatsupaibul et al. (2018) to test the performance of the shrinking-ball algorithms (i.e., IHR-SO and AP-SO), we use a smaller scaling constant for the problem in this paper. For the Hills problem and the Rosenbrock problem, we add simulation noises with distributions $N(0, (1/4g)(\boldsymbol{x}))$ and $N(0, 0.01(1 + |g(\boldsymbol{x})|)^2)$ for $\boldsymbol{x} \in \mathcal{X}$, respectively.

The kernel-based method is used in the GPS-C algorithm to estimate the variance at each sampled point. The bandwidth of the kernel is set as $h_n = h_0(n+1)^{-\beta}$, where h_0 is the initial bandwidth, β is the contracting rate, and n is the number of sampled points. For the Hills problem, we use an LHS design of 100 points in the initialization stage to estimate the parameters of Gaussian process and the variances of the noises for the GPS-C algorithm, and then, we update the Gaussian process parameters at the 20th and 40th iterations. The same method as in Section 6.2.1 is used to find $\hat{x}_n^*$ in each iteration. For the Rosenbrock problem, because the dimensionality is high, the GPS-C algorithm starts with an initial design of 800 points. Among these points, 400 points are generated according to the LHS design, and another 400 points are randomly sampled within small balls centered at the previous 400 points in a point-to-point manner. The variances and the parameters of Gaussian process are then estimated for the GPS-C algorithm. To find $\hat{x}_n^*$ in each iteration, the fmincon in MATLAB is used with the sample best solution as the initial solution (see Section 5.2.3). The parameters of

Table 1. The Performance of SKO, KGCP, and GPS-C Algorithms up to Different Sample Size

	SKO				KGCP				GPS-C			
Problem	80	200	400	800	80	200	400	800	80	200	400	800
Branin ($\lambda^2 = 0.1^2$)	0.009	0.006	0.004	0.003	0.020	0.003	0.002	0.002	0.754	0.015	0.003	0.002
Branin ($\lambda^2 = 0.5^2$)	0.031	0.020	0.016	0.014	0.025	0.015	0.012	0.010	0.249	0.013	0.010	0.007
Six hump ($\lambda^2 = 0.1^2$)	0.006	0.004	0.003	0.002	0.010	0.003	0.003	0.002	0.152	0.005	0.002	0.001
Six hump ($\lambda^2 = 0.5^2$)	0.078	0.026	0.018	0.011	0.070	0.020	0.012	0.009	0.332	0.012	0.007	0.005
Hills ($\lambda^2 = 0.5^2$)	0.674	0.357	0.043	0.013	0.711	0.143	0.080	0.027	3.217	0.377	0.009	0.002
Hills ($\lambda^2 = 1^2$)	1.449	0.516	0.066	0.051	0.934	0.185	0.019	0.016	3.642	0.667	0.034	0.009

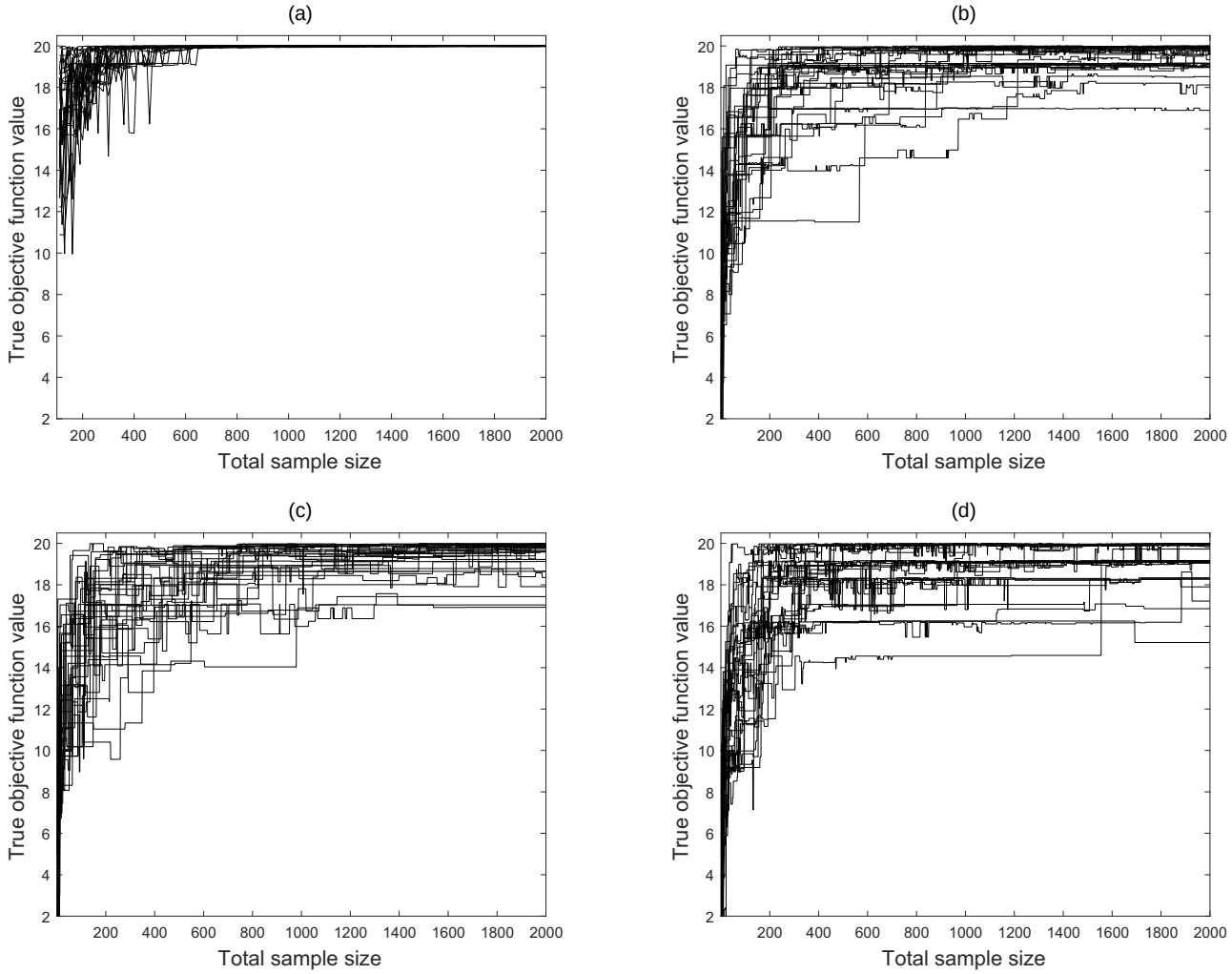
the ASR, IHR-SO, and AP-SO algorithms are basically the same as those in Andradóttir and Prudius (2010) and Kiatsupaibul et al. (2018). For each problem, all the parameters (that are not estimated) of the four algorithms are listed in Table EC.4 in the e-companion.

Figure 3 shows the 30 replications of the four algorithms when solving the Hills problem in terms of $g(\hat{x}_n^*)$ as a function of the sample size n. It can be observed that the GPS-C algorithm performs better than the other three random search algorithms. The GPS-C algorithm can soon find good solutions that are close to the global optimal solution. This comparison may imply that the GPS-C algorithm can sample design points in a more adaptive manner and converge more quickly. Figure 4 shows the 30 replications of the four algorithms when solving the the Rosenbrock problem, and similar results are observed. The GPS-C algorithm identifies good solutions in early iterations and approaches the global optimal function value gradually in most of the 30 replications. Besides the ability to balance exploration and exploitation adaptively, this excellent performance can also be attributed partly to the characteristics of the Rosenbrock function, which is quite flat in the neighborhood of the global optimal solution.

To further illustrate how the GPS-C algorithm works, in Figure 5 we plot the sampled points of the four algorithms up to different sample sizes when solving the Hills problem. It can be observed that the GPS-C algorithm achieves a good balance between exploration and exploitation compared with the other two algorithms. Many points are sampled in good regions (around the best solution and the two second-best solutions), when the algorithm keeps exploring the whole feasible region. The comparison of these three algorithms illustrates how the constructed sampling distributions of the GPS-C algorithm can guide the searches in each iteration.

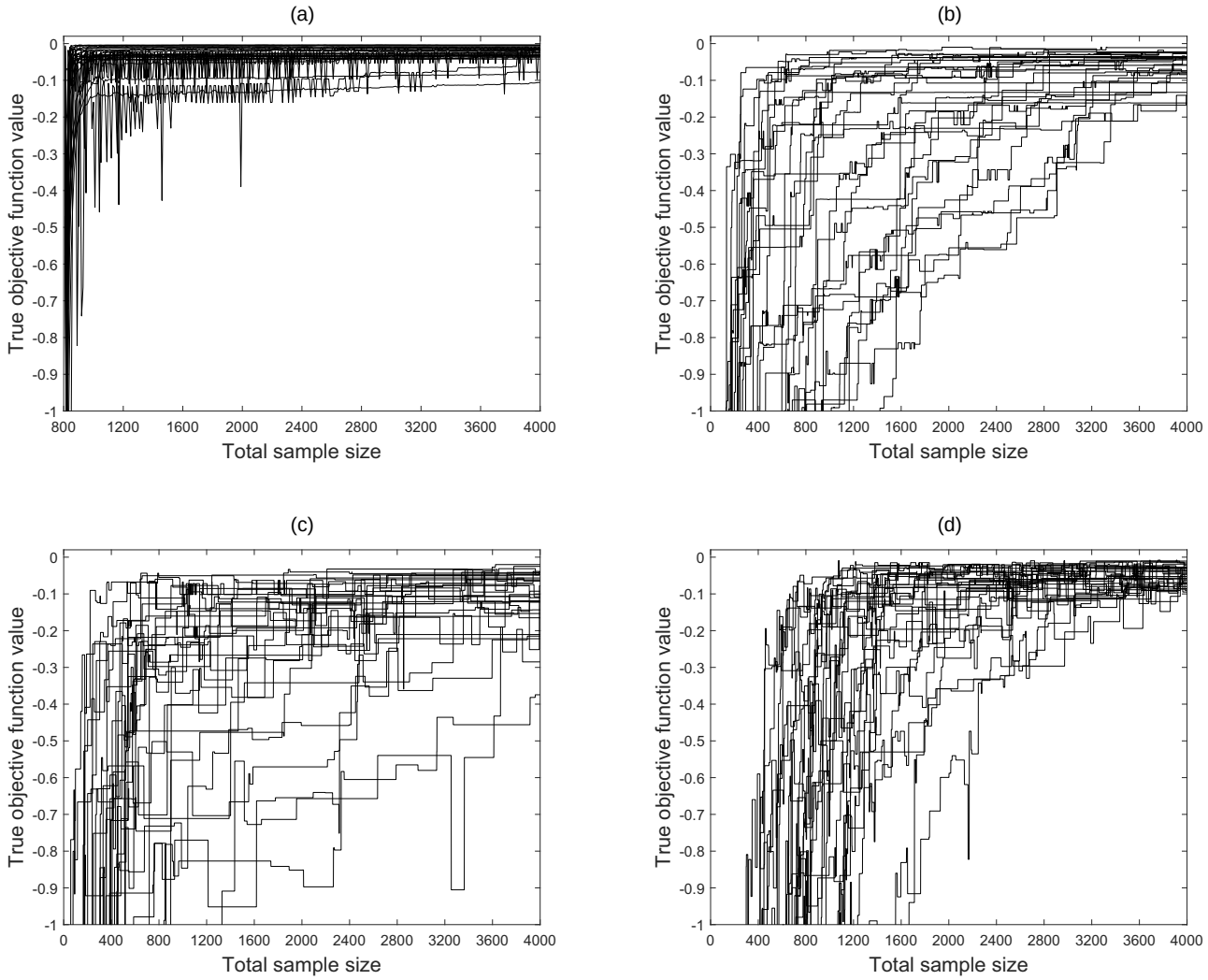
Lastly, we also implement the revised GPS-C algorithm (as described in Section 4.3) to solve the two problems. The performance of the revised GPS-C algorithm

Figure 3. Performance of the GPS-C and Other Compared Algorithms for the Hills Problem



Notes. (a) GPS-C algorithm. (b) ASR algorithm. (c) IHR-SO algorithm. (d) AP-SO algorithm.

Figure 4. Performance of the GPS-C and Other Compared Algorithm for the 10-Dimensional Rosenbrock Problem



Notes. (a) GPS-C algorithm. (b) ASR algorithm. (c) IHR-SO algorithm. (d) AP-SO algorithm.

is shown in Figure EC.2 in the e-companion with the same performance measure. It can be observed that the performance of the revised GPS-C algorithm is similar to that of the original GPS-C algorithm, whereas the computation burden for finding the optimal solution in each iteration is lower.

Several conclusions can be made from the experiment results in Sections 6.2.1 and 6.2.2. First, the global convergence of the GPS-C algorithm is verified by test problems with unknown (both equal and unequal) variance of simulation noise, and the performance of the GPS-C algorithm is robust. Second, numerical experiments show that the GPS-C algorithm maintains the advantages of the GPS algorithm (Sun et al. 2014) in balancing exploration and exploitation, and its finite-sample performance appears better than the other three random search-based algorithms, which use more rigid sampling schemes.

6.3. The Impact of the Dimensionality on the Rate of Convergence

In this subsection, our goal is to understand the impact of dimensionality on the rate of convergence of the GPS-C algorithm. Because it is difficult to generate sample paths from a high-dimensional Gaussian process, we instead use the 4-, 6-, 8-, and 10-dimensional weighted sphere problems to investigate the impact of dimensionality. For the four problems, simulation noises with distribution $N(0, 0.1^2)$ are added. The Gaussian process parameters and the variances of the noises are estimated with an LHS design of 600 points, and other parameters are listed in Table EC.6 in the e-companion.

Figure 6 shows the average optimality gap (i.e., $|\hat{\mu}(\hat{\boldsymbol{x}}_n^*) - g^*|$), with respect to the sample size n on a log-log plot. From these plots, we can see that the observed rates of convergence are better than the theoretical rates and that the dimensionality of the problem does not

Figure 5. (Color online) The Sampled Points of the GPS-C, ASR, IHR-SO, and AP-SO Algorithms

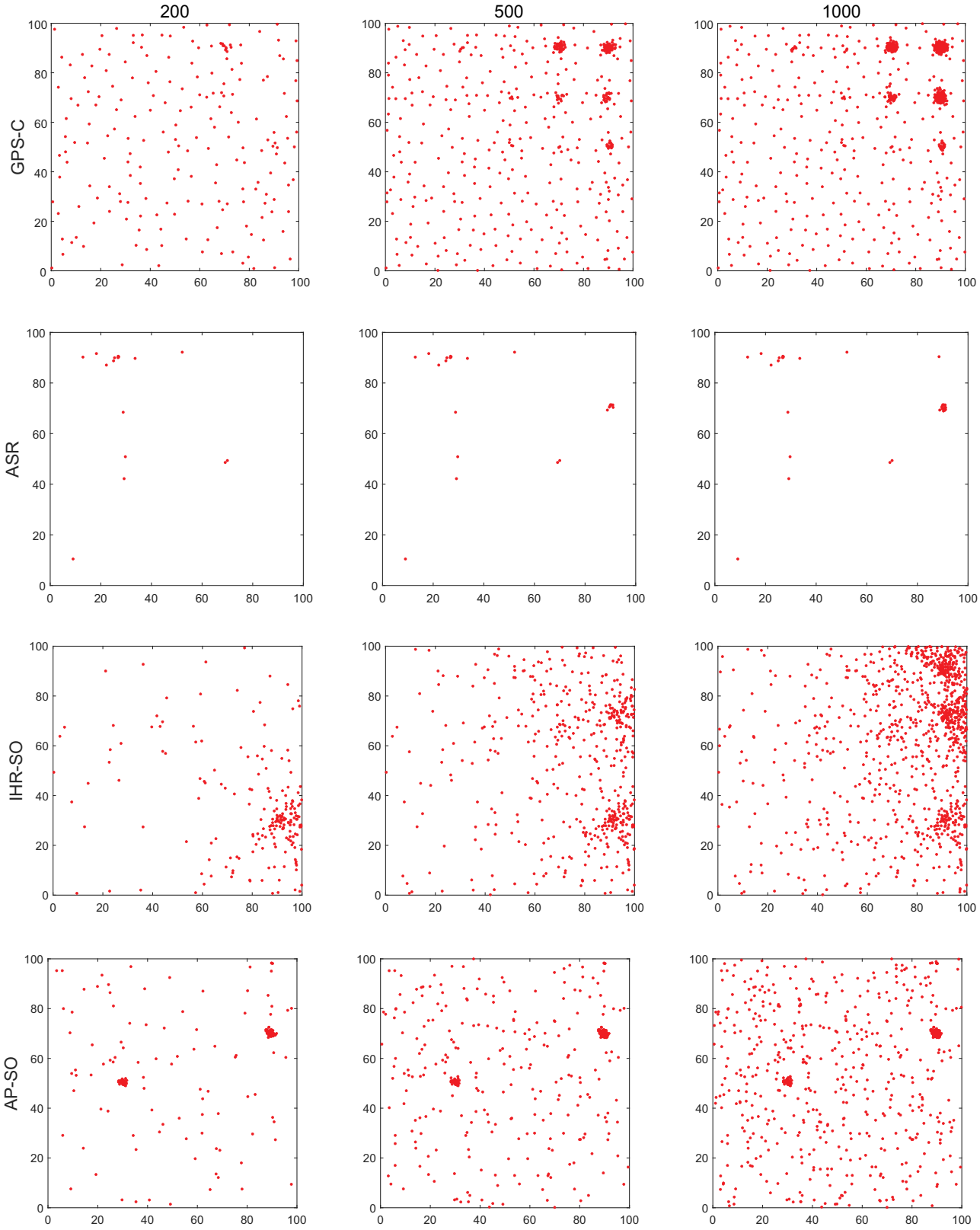
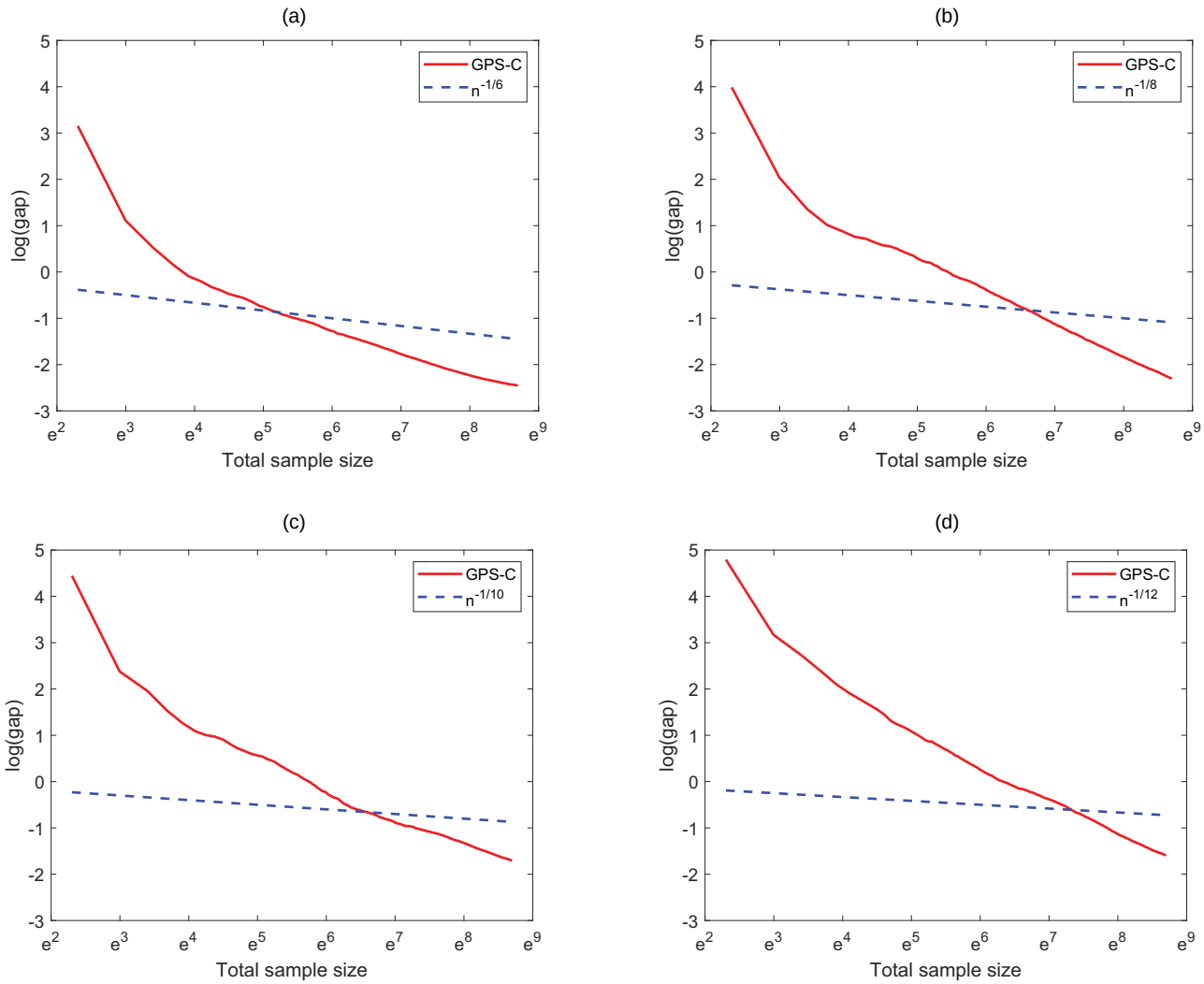


Figure 6. (Color online) Empirical Rate of Convergence of the GPS-C Algorithm on 4-, 6-, 8-, and 10-Dimensional Weighted Sphere Problems



Notes. (a) Four-dimensional problem. (b) Six-dimensional problem. (c) Eight-dimensional problem. (d) Ten-dimensional problem.

have significant impact on the empirical rate of convergence of the GPS-C algorithm. We suspect that it is because the weighted sphere function is smoother than the typical sample paths from Gaussian processes satisfying either Assumption 3 or Assumption 5. As many practical problems in the field of operations research and management sciences are in general quite smooth, we suspect that the performances of the GPS-C algorithm on the weighted sphere functions are more common and more representative.

7. Conclusions

In this paper, we propose a framework of Gaussian process-based random search algorithms for the COvS problem. Algorithms under the GPRS framework (1) use a Gaussian process surrogate model to estimate the objective function and (2) randomly sample solutions from a sequence of lower-bounded sampling distributions. Under heteroscedastic and known simulation noises, we prove the global convergence of GPRS algorithms. Moreover, when the objective functions are sampled from a Gaussian process having continuously differentiable sample paths, we prove the upper bound of the rate of convergence of GPRS algorithms, which can be $\tilde{O}_p(n^{-1/(d+2)})$. Then, the GPS-C algorithm is proposed as an example to illustrate how to design and implement an integrated GPRS algorithm. Numerical experiments show that the GPS-C algorithm performs well, even for problems with unknown variances of simulation noises.

There are several directions to potentially extend this work. First, the global convergence of GPRS algorithms with unknown and heteroscedastic simulation noises may be studied. This is an important theoretical extension, although it may be quite challenging. Second, it is interesting to examine whether the faster rate of convergence may

be established for GPRS algorithms by using either sharper inequalities or additional assumptions. Lastly, it may be interesting to further test the performance of the GPS-C algorithm in different scenarios and develop an open-source COvS solver based on the algorithm.

Acknowledgments

The authors thank the editor-in-chief John Birge, the area editor Daniel Kuhn, the associate editor, and two referees for helpful comments and suggestions, which helped to improve this paper during three revisions.

Endnote

[1] Let $\{a_n\}_{n\geq 1}$ be a sequence such that $a_n > 0$ for all n. A function $h(n)$ of n is called $\Omega(a_n)$ if there is a $c \in (0,\infty)$ such that for all $n \in \mathbb{N}$, $h(n) \geq ca_n$. A function $h(n)$ is called $O(a_n)$ if there is a $C \in (0,\infty)$ such that for all $n \in \mathbb{N}$, $0 < h(n) \leq Ca_n$. A function $h(n)$ is called $\Theta(a_n)$ if it is both $\Omega(a_n)$ and $O(a_n)$.

References

Abrahamsen P (1997) A review of Gaussian random fields and correlation functions. Technical report, Norwegian Computing Center, Oslo, Norway.

Adler RJ, Taylor JE (2007) *Random Fields and Geometry* (Springer Science & Business Media, New York).

Amaran S, Sahinidis NV, Sharda B, Bury SJ (2016) Simulation optimization: A review of algorithms and applications. *Ann. Oper. Res.* 240(1):351–380.

Andradóttir S (2006) An overview of simulation optimization via random search. Henderson SG, Nelson BL, eds. *Handbooks in Operations Research and Management Science*, vol. 13 (Elsevier, Amsterdam), 617–631.

Andradóttir S (2015) A review of random search methods. Fu M, ed. *Handbook of Simulation Optimization*, International Series in Operations Research & Management Science, vol. 216 (Springer, New York), 277–292.

Andradóttir S, Prudius AA (2009) Balanced explorative and exploitative search with estimation for simulation optimization. *INFORMS J. Comput.* 21(2):193–208.

Andradóttir S, Prudius AA (2010) Adaptive random search for continuous simulation optimization. *Naval Res. Logist.* 57(6):583–604.

Ankenman B, Nelson BL, Staum J (2010) Stochastic kriging for simulation metamodeling. *Oper. Res.* 58(2):371–382.

Azaïs JM, Wschebor M (2009) *Level Sets and Extrema of Random Processes and Fields* (John Wiley & Sons, Hoboken, NJ).

Baumert S, Smith RL (2002) Pure random search for noisy objective functions. Technical report, University of Michigan, Ann Arbor.

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

Bernstein DS (2009) *Matrix Mathematics* (Princeton University Press, Princeton, NJ).

Box GE, Wilson KB (1951) On the experimental attainment of optimum conditions. *J. Roy. Statist. Soc. B* 13(1):1–38.

Boyd S, Boyd SP, Vandenberghe L (2004) *Convex Optimization* (Cambridge University Press, Cambridge, UK).

Bull AD (2011) Convergence rates of efficient global optimization algorithms. *J. Machine Learn. Res.* 12(10):2879–2904.

Calvin J, Žilinskas A (1999) On the convergence of the p-algorithm for one-dimensional global optimization of smooth functions. *J. Optim. Theory Appl.* 102(3):479–495.

Chang KH, Hong LJ, Wan H (2013) Stochastic trust-region response-surface method (strong)—A new response-surface framework for simulation optimization. *INFORMS J. Comput.* 25(2):230–243.

Chia YL, Glynn PW (2013) Limit theorems for simulation-based optimization via random search. *ACM Trans. Model. Comput. Simulation* 23(3):1–18.

Devroye L (1978) The uniform convergence of nearest neighbor regression function estimators and their application in optimization. *IEEE Trans. Inform. Theory* 24(2):142–151.

Ding L, Hong LJ, Shen H, Zhang X (2022) Knowledge gradient for selection with covariates: Consistency and computation. *Naval Res. Logist.* 69(3):496–507.

Donoho DL (1994) Asymptotic minimax risk for sup-norm loss: Solution via optimal recovery. *Probab. Theory Related Fields* 99(2):145–170.

Dubourg V, Deheeger F, Sudret B (2011) Metamodel-based importance sampling for the simulation of rare events. Preprint, submitted April 18, https://arxiv.org/pdf/1104.3476.pdf.

Dubourg V, Sudret B, Deheeger F (2013) Metamodel-based importance sampling for structural reliability analysis. *Probab. Engrg. Mechanics* 33:47–57.

Durrett R (2010) *Probability: Theory and Examples*, 4th ed. (Cambridge University Press, Cambridge, UK).

Ensor KB, Glynn PW (1997) Stochastic optimization via grid search. Yin GG, Zhang Q, eds. *Lectures in Applied Mathematics, Mathematics of Stochastic Manufacturing Systems*, vol. 33 (American Mathematical Society, Providence, RI), 89–100.

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

Gut A (2013) *Probability: A Graduate Course*, vol. 75 (Springer Science & Business Media, New York).

Hu J, Hu P (2011) Annealing adaptive search, cross-entropy, and stochastic approximation in global optimization. *Naval Res. Logist.* 58(5):457–477.

Hu J, Fu MC, Marcus SI (2007) A model reference adaptive search method for global optimization. *Oper. Res.* 55(3):549–568.

Huang D, Allen TT, Notz WI, Zeng N (2006) Global optimization of stochastic black-box systems via sequential kriging meta-models. *J. Global Optim.* 34(3):441–466.

Jones DR, Schonlau M, Welch WJ (1998) Efficient global optimization of expensive black-box functions. *J. Global Optim.* 13(4):455–492.

Kiatsupaibul S, Smith RL, Zabinsky ZB (2018) Single observation adaptive search for continuous simulation optimization. *Oper. Res.* 66(6):1713–1727.

Kiefer J, Wolfowitz J (1952) Stochastic estimation of the maximum of a regression function. *Ann. Math. Statist.* 23(3):462–466.

Kleijnen JP (1998) Experimental design for sensitivity analysis, optimization, and validation of simulation models. Banks J, ed. *Handbook of Simulation* (John Wiley & Sons, New York), 173–223.

Kushner HJ, Yin G (1997) *Stochastic Approximation Algorithms and Applications* (Springer, New York).

Osorio C, Bierlaire M (2013) A simulation-based optimization framework for urban transportation problems. *Oper. Res.* 61(6):1333–1345.

Picheny V, Wagner T, Ginsbourger D (2013) A benchmark of kriging-based infill criteria for noisy optimization. *Structural Multidisciplinary Optim.* 48(3):607–626.

Rasmussen CE, Williams CK (2006) *Gaussian Processes for Machine Learning* (MIT Press, Cambridge, MA).

Robbins H, Monro S (1951) A stochastic approximation method. *Ann. Math. Statist.* 22(3):400–407.

Schimek MG (2013) *Smoothing and Regression: Approaches, Computation, and Application* (John Wiley & Sons, New York).

Scott W, Frazier P, Powell W (2011) The correlated knowledge gradient for simulation optimization of continuous parameters using Gaussian process regression. *SIAM J. Optim.* 21(3):996–1026.

Shapiro A, Dentcheva D, Ruszczyński A (2009) *Lectures on Stochastic Programming: Modeling and Theory* (SIAM, Philadelphia).

Shen H, Hong LJ, Zhang X (2018) Enhancing stochastic kriging for queueing simulation with stylized models. *IISE Trans.* 50(11):943–958.

Spall JC (1992) Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. *IEEE Trans. Automatic Control* 37(3):332–341.

Stein ML (1999) *Interpolation of Spatial Data: Some Theory for Kriging* (Springer Science & Business Media, New York).

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

Sun W, Hu Z, Hong LJ (2018) Gaussian mixture model-based random search for continuous optimization via simulation. Rabe M, Juan AA, Mustafee N, Skoogh A, Jain S, Johansson B, eds. *2018 Winter Simulation Conf. (WSC)* (IEEE, Piscataway, NJ), 2003–2014.

Tao T (2009) *Analysis* (Springer, New York).

Yakowitz S, L'ecuyer P, Vázquez-Abad F (2000) Global stochastic optimization with low-dispersion point sets. *Oper. Res.* 48(6):939–950.

Yu T, Zhu H (2020) Hyper-parameter optimization: A review of algorithms and applications. Preprint, submitted December 3, https://arxiv.org/pdf/2003.05689.pdf.

Zhang Q, Hu J (2022) Actor-critic–like stochastic adaptive search for continuous simulation optimization. *Oper. Res.* 70(6):3519–3537.

Žilinskas A (1985) Axiomatic characterization of a global optimization algorithm and investigation of its search strategy. *Oper. Res. Lett.* 4(1):35–39.

Xiuxian Wang is currently a postdoctoral researcher with the Sino-US Global Logistics Institute, Shanghai Jiao Tong University. His research interests include simulation optimization and healthcare operations management.

L. Jeff Hong is the Fudan Distinguished Professor and the Hongyi Chair Professor appointed by the School of Management and the School of Data Science at Fudan University. His research interests are in the broad areas of machine learning and business analytics: stochastic modeling, stochastic simulation, stochastic optimization, statistical learning, and reinforcement learning, with applications in supply chain management, revenue management, financial risk management, and healthcare analytics.

Zhibin Jiang is currently the Changjiang Scholar Chair Professor of Ministry of Education (MOE), China, and a distinguished professor with the Antai College of Economics and Management and the Dean of the Sino-US Global Logistics Institute at Shanghai Jiao Tong University. His research interests include discrete-event modelling and simulation, and operation management in manufacturing and healthcare system.

Haihui Shen is an associate professor in the Sino-US Global Logistics Institute at Shanghai Jiao Tong University. His research interests include simulation modeling, analysis, and optimization with applications in manufacturing, logistics, and healthcare.