

OPTIMIZATION VIA SIMULATION USING GAUSSIAN PROCESS-BASED SEARCH

Lihua Sun

Department of Economics and Finance, School of Economics and Management
Tongji University
1239 Siping Road
Shanghai 200092, China

L. Jeff Hong

Department of Industrial Engineering and Logistics Management
The Hong Kong University of Science and Technology
Hong Kong, China

Zhaolin Hu

Department of Management Science and Engineering, School of Economics and Management
Tongji University
1239 Siping Road
Shanghai 200092, China

ABSTRACT

Random search algorithms are often used to solve optimization-via-simulation (OvS) problems. The most critical component of a random search algorithm is the sampling distribution that is used to guide the allocation of the search effort. A good sampling distribution can balance the tradeoff between the effort used in searching around the current best solution (which is called exploitation) and the effort used in searching largely unknown regions (which is called exploration). However, most of the random search algorithms for OvS problems have difficulties in balancing this tradeoff in a seamless way. In this paper we propose a new random search algorithm, called the Gaussian Process-based Search (GPS) algorithm, which derives a sampling distribution from a fast fitted Gaussian process in each iteration of the algorithm. We show that the sampling distribution has the desired properties and it can automatically balance the exploitation and exploration tradeoff.

1 INTRODUCTION

Many globally convergent random search algorithms have been proposed to solve optimization-via-simulation (OvS) problems, e.g., stochastic ruler of Yan and Mukai (1992), nested partition of Shi and Ólafsson (2000), model reference method of Hu et al. (2007, 2008), and the shrinking ball method of Andradóttir and Prudius (2010). In every iteration of these algorithms, a sampling distribution needs to be constructed based on all the information collected through the last iteration and it is used to guide the search effort in the current iteration.

To achieve fast global convergence, these algorithms have to balance the search effort in the current local neighborhood and in the unknown regions. This is known as the exploitation and exploration tradeoff. Exploitation refers to the search around current solution. Because there are often better solutions near the

current solution, the exploitative search often has a high chance to find better solutions. Exploration refers to the search in the entire feasible region. Because there may be regions that are significantly better than the regions searched so far, explorative search may identify those regions to save the search effort the algorithm may otherwise spend on searching the inferior region. Therefore, how to balance the exploitation and exploration is a critical issue in designing globally convergent random search algorithms.

In this paper we propose a new approach to constructing sampling distributions. In each iteration of a random search algorithm, our approach first constructs a Gaussian process passing through the sample mean of previously visited points and takes into consideration the estimation errors. Based on the Gaussian process, it calculates the probability of any solution better than the current best solution and normalizes these probabilities into a sampling distribution. We show that the sampling distribution automatically balances the tradeoff between exploitation and exploration.

We also design two OvS algorithms based on the sampling distribution proposed in this paper. One algorithm is designed to solve deterministic continuous OvS problems and the other is designed to solve stochastic discrete OvS problems. We call these algorithms Gaussian Process-based Search (GPS). We show that the GPS algorithms are globally convergent for respective problems.

The rest of this paper is organized as follows. In Section 2, we discuss the desired properties of a sampling distribution and illustrate our idea through a simple one-dimensional example. In Section 3, we propose a novel way to build sampling distribution based on Gaussian process and discuss how to sample from it. In Section 4, we give Gaussian process-based search algorithms for both deterministic continuous and stochastic discrete problems. The results of the numerical study are given in Section 5.

2 DESIRED PROPERTIES OF SAMPLING DISTRIBUTIONS

In this paper, we are interested in solving deterministic continuous optimization problems and stochastic discrete optimization problems which have the following form

$$\max_{x \in \Theta} g(x) \quad (1)$$

where Θ is either a finite discrete set or a compact continuous set and the closed-form expression of $g(x)$ is not available. Let Θ^* be the set of optimal solutions, g^* be the maximal value of g on Θ and M be the lower bound of g on Θ . Note that Θ is either a finite discrete set or a compact continuous set. It is not difficult to show that there exists $M \leq g(x) < \infty$ for any $x \in \Theta$ and there exists at least one point $x^* \in \Theta$ such that $g(x^*) = g^*$. For any $x \in \Theta$, if $g(x)$ could only be estimated with noise, we further assume $g(x) = E[G(x)]$, where $G(x)$ is a measurable and integrable for all $x \in \Theta$.

Random search algorithms are often used to solve Problem (1). In each iteration of a random search algorithm, a sampling distribution needs to be constructed and new solutions are sampled based on this distribution. Selection of a proper sampling distribution is central to the tradeoff between exploration and exploitation and is critical to the performance of the algorithm. In this section, we discuss the desired properties of sampling distributions through a simple deterministic example. We then illustrate our idea of constructing a proper sampling distribution using this example.

2.1 Desired Properties of Sampling Distribution

Suppose that we want to solve a one-dimensional optimization problem whose objective function $g(x)$ can be evaluated without noise. In the current iteration of a random search algorithm, we have evaluated six points, x_1, \dots, x_6 , as shown in Figure 1. Based on the given information, how should the sampling effort be allocated?

For clearer illustration, we divide the feasible region into five subregions, denoted as R_1, \dots, R_5 . We believe that a reasonable sampling distribution for this example should have at least the following properties:

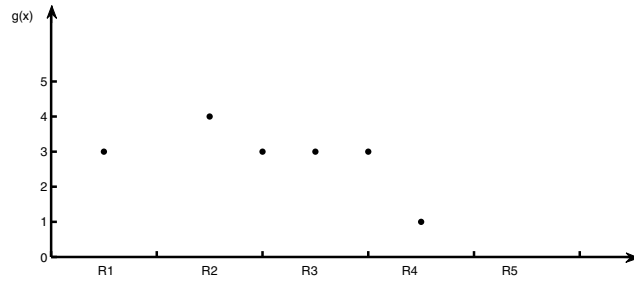


Figure 1: Function values of f at some points.

- allocating more probabilities to points in R_2 because it is likely to find better solutions around the current best solution,
- allocating more probabilities to R_1 than to R_3 even though the evaluated solutions in these two regions have the same objective value because R_1 is less explored,
- allocating more probabilities to R_5 than R_4 because R_5 is more likely to contain better solutions.

Although all these properties make sense, to the best of our knowledge, none of the existing random search algorithms have constructed a sampling distribution that satisfies all these properties. In the rest of this section, we show how to construct a sampling distribution that satisfies all these properties.

2.2 Idea of Building Sampling Distribution

Suppose that $g(x)$ is a sample path of a Brownian motion process $Y(x)$ starting from time $-\infty$. Note that we know that the Brownian motion process passes through $(x_1, g(x_1)), \dots, (x_6, g(x_6))$. This limits the possibilities that the sample path may take (Figure 2). Given this limit, we can derive the mean and variance of all the points in the feasible region (Figure 3) based on the properties of a Brownian motion process. Then, we can further calculate the probability that each point has a value that is better than the current best solution, i.e., $P(Y(x) > g_{k-1}^*)$.

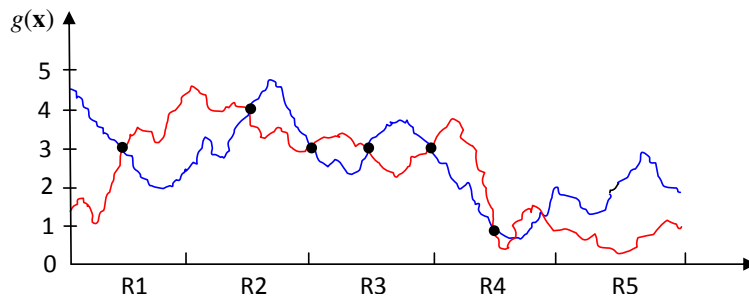


Figure 2: Brownian motion processes passing through x_1, \dots, x_6 .

We normalize these probabilities and use it as a sampling distribution. From Figure 4, it is clear that this sampling distribution satisfies all the desired properties discussed in Section 2.1.

To apply the idea to multi-dimensional OvS problems, we need to consider the following issues:

- how to extend the one-dimensional Brownian motion process to a stochastic process that allows multiple inputs and only one output,
- how to handle the possible noise in the estimation of objective values, and
- how to sample efficiently from the sampling distribution.

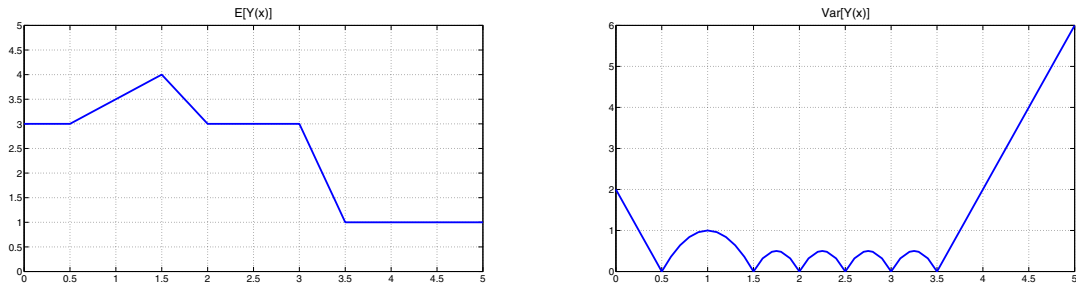


Figure 3: Mean and variance of the Brownian motion process.

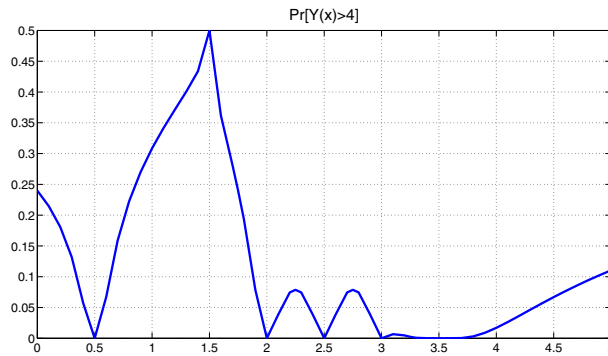


Figure 4: $P(Y(x) > g_{k-1}^*)$ under the Brownian motion process.

We will discuss all these issues in Section 3 and discuss how to design OvS algorithms based on the sampling distribution in Section 4.

3 GAUSSIAN PROCESS-BASED SAMPLING DISTRIBUTION

In the previous section, we use a simple one dimension example to illustrate our idea of constructing a balanced sampling distribution. In this section, similar idea is extended to higher dimensional problems. We first discuss how to fit response surface and construct sampling distribution based on traditional kriging metamodeling and discuss the limitations of applying kriging idea. We then propose a novel approach which is more suitable for constructing response surface and sampling distribution.

3.1 Kriging-based Search

In kriging metamodeling, it is usually assumed that a stochastic output on replication j at point x is

$$G_j(x) = u + M(x) + \varepsilon_j(x), \tag{2}$$

where u is a constant, $M(x)$ is a Gaussian process with mean 0 and stationary covariance function $\sigma^2 \gamma(\cdot, \cdot)$, and $\varepsilon(x)$ is a normal random variable that has zero mean, variance $\sigma_\varepsilon^2(x)$ and covariance $\text{Cov}(\varepsilon(x), \varepsilon(x')) = 0$ for any $x \neq x'$. If we have taken $n_i, i = 1, \dots, n$ replications at point x_i with sample mean $\bar{G}(x_i)$. Ankenman et al. (2010) show that the IMSE-optimal linear predictor of g at x_0 is

$$\hat{g}(x_0) = u + \lambda(x_0)^T (\bar{\mathbf{G}} - u\mathbf{1}),$$

with $\bar{\mathbb{G}} = (\bar{G}(x_1), \dots, \bar{G}(x_n))^T$ and

$$\lambda(x_0)^T = \gamma(x_0)^T \left(\Gamma + \frac{1}{\sigma^2} \Sigma_\varepsilon \right)^{-1},$$

where $\gamma(x_0)$ denotes the vector of coefficient $(\gamma(x_0, x_1), \dots, \gamma(x_0, x_n))^T$, Γ denotes the $n \times n$ coefficient matrix whose (i, j) -th element is $\gamma(x_i, x_j)$, the (i, i) -th element of Σ_ε is σ_ε^2/n_i whereas the other elements are all 0. The optimal MSE is

$$\text{MSE}(x_0) = \sigma^2 \left(1 - \gamma(x_0)^T \left(\Gamma + \frac{1}{\sigma^2} \Sigma_\varepsilon \right)^{-1} \gamma(x_0) \right).$$

Remark 1 In kriging metamodeling, we assume $g(x)$ is a sample path of the Gaussian process $u + M(x)$. If $g(x)$ can be observed without noise, by setting $\sigma_\varepsilon^2(x) \equiv 0$ this degenerates to the deterministic kriging model.

Based on the kriging model, we know the predicted value of $g(x_0)$ is normally distributed with mean $\hat{g}(x_0)$ and variance $\text{MSE}(x_0)$ for every $x_0 \in \Theta$. Denote the predicted value of $g(x_0)$ as $Y(x_0)$. For any $z \in \mathbb{R}$, $p(z) = \Pr\{Y(x_0) > z\}$ can be calculated by normal distribution function. Let $z = \hat{g}_{k-1}^* := \bar{G}_{k-1}(x_{k-1}^*)$. Then, $p(z)$ represents the probability that x_0 has a higher objective function value than the current optimal solution. Intuitively, the higher the $p(z)$, the more sampling effort should be allocated for x_0 . The sampling distribution is constructed based on this intuitive idea. We set $f_k(x)$, the probability density (mass) function of sampling distribution at iteration k as

$$f_k(x) = \frac{\Pr\{Y(x) > \hat{g}_{k-1}^*\}}{\int_{\Theta} \Pr\{Y(z) > \hat{g}_{k-1}^*\} dz}. \quad (3)$$

There are several difficulties when applying this sampling distribution function. First, in each iteration we need to invert an $n \times n$ matrix $(\Gamma + \frac{1}{\sigma^2} \Sigma_\varepsilon)$. If the number of visited points is large, it is time-consuming to invert this matrix. Moreover, if points are very close to each other, inverting the matrix may be impossible because the matrix is not full-rank. Second, it will be time consuming and sometimes even impossible to calculate the closed form of $f_k(x)$ because the denominator of the right side of (3) is an integration.

Kriging is originally used for surface fitting, based on the known points on the surface. The parameters of the Gaussian process is first estimated by statistical methods such as MLE (Maximum Likelihood Estimation) and then a response surface is constructed on conditional probability. The matrix $\Gamma + \frac{1}{\sigma^2} \Sigma_\varepsilon$ needs to be inverted for only one time, its dimension is usually not very high, and points in most cases scatter in the whole solution set. Therefore, it will not be difficult to invert the matrix. However, when we apply kriging metamodeling for optimization, response surfaces need to be updated iteratively according to the latest information, the dimension of Γ increases quickly and it is highly possible that Γ will be ill conditioned because points will cluster in the good performance region. As has been mentioned in Jones et al. (1998), applying kriging modeling directly in optimization will meet some difficulties when inverting the matrix. In this paper, to overcome this difficulty, in the following subsection, we propose a new way for building response surface to avoid the problem of inverting matrices.

3.2 Fast Construction of Gaussian Process

As in the preceding subsection, suppose we have evaluated the objective function on the set $\mathbb{S} = \{x_1, x_2, \dots, x_n\}$, and for any $x_i \in \mathbb{S}, i = 1, \dots, n$, we have collected n_i observations on x_i with the mean observation value $\bar{G}(x_i)$ and sample variance $\hat{\sigma}^2(x_i)$ by the $(k-1)$ th iteration.

We construct the following Gaussian process:

$$Y_k(x) = Z(x) + \lambda(x)^T (\bar{\mathbb{G}}_{\mathbb{S}} - \mathbb{Z}_{\mathbb{S}}) + \lambda(x)^T \varepsilon_{\mathbb{S}}, \quad (4)$$

where $\bar{\mathbb{G}}_{\mathbb{S}} = (\bar{G}(x_1), \dots, \bar{G}(x_n))^T$, $\mathbb{Z}_{\mathbb{S}} = (Z(x_1), \dots, Z(x_n))^T$ with $Z(x)$ being a Gaussian process with mean 0 and covariance function $\sigma^2 \gamma(\cdot, \cdot)$, and $\varepsilon_{\mathbb{S}} = (\varepsilon_1, \dots, \varepsilon_n)^T$ where $\varepsilon_1, \dots, \varepsilon_n$ are n independent normal random variables with mean 0 and covariance matrix

$$\hat{\Sigma}_{\varepsilon} = \mathbf{diag} \left\{ \frac{\max \{ \hat{\sigma}^2(x_1), \sigma_0^2 \}}{n_1}, \dots, \frac{\max \{ \hat{\sigma}^2(x_n), \sigma_0^2 \}}{n_n} \right\}.$$

We assume $\lambda(\cdot)$ satisfies the following conditions:

- i. $\sum_{i=1}^n \lambda_i(x) = 1$;
- ii. $\lambda_i(x) \geq 0$ for any $i = 1, \dots, n$;
- iii. $\lambda_i(x_j) = \delta(i, j)$, where $\delta(i, j)$ is an indicator function.

Remark 2 There are many choices of $\lambda(x)$ that satisfy the three conditions, e.g., $\lambda_i(x) = \frac{|1-\gamma(x, x_i)|^{-1}}{\sum_{j=1}^n |1-\gamma(x, x_j)|^{-1}}$ or $\lambda_i(x) = \frac{e^{|1-\gamma(x, x_i)|^{-1}}}{\sum_{j=1}^n e^{|1-\gamma(x, x_j)|^{-1}}}$ for any $x \neq x_i$.

By simple mathematical calculation, we obtain the following property about $Y_k(x)$.

Proposition 1 For any $x \in \Theta$, $E[Y_k(x)] = \lambda(x)^T \bar{\mathbb{G}}_{\mathbb{S}}$, and

$$\text{Var}[Y_k(x)] = \sigma^2 (1 - 2\lambda(x)^T \gamma(x) + \lambda(x)^T \Gamma \lambda(x)) + \lambda(x)^T \hat{\Sigma}_{\varepsilon} \lambda(x)$$

where Γ denotes the coefficient matrix, $\gamma(x)$ denotes the coefficient vector. Specially, if $x = x_i$, $E[Y_k(x_i)] = \bar{G}(x_i)$, $\text{Var}[Y_k(x_i)] = \max \{ \hat{\sigma}^2(x_i), \sigma_0^2 \} / n_i$.

The sampling distribution has the same form as in (3).

Remark 3 If g can be observed without noise, then the variance at the estimated points is zero, $g(x)$ becomes a sample path of the following process

$$Y_k(x) = Z(x) + \lambda(x)^T (\bar{\mathbb{G}}_{\mathbb{S}} - \mathbb{Z}_{\mathbb{S}}).$$

Remark 4 In our model, we let $\text{Var}[\varepsilon(x_i)] = \text{Var}[Y_k(x_i)] = \max \{ \hat{\sigma}^2(x_i), \sigma_0^2 \} / n_i$ where σ_0 could be an arbitrarily small number, rather than $\text{Var}[Y(x_i)] = \hat{\sigma}^2(x_i) / n_i$. This is because with finite number of observations, it is impossible to decide whether $\text{Var}[Y_k(x_i)]$ equals to 0 or not. For example, if $\Pr\{Y_k(x_i) = 1\} = 1 - 10^{-5}$ and $\Pr\{Y_k(x_i) = 0\} = 10^{-5}$, then it is highly possible that $\hat{\sigma}^2(x_i) = 0$ with finite number of observations though the true variance is not 0.

We directly construct a Gaussian process passing through the estimated points. The distribution of unestimated points in our example is calculated by unconditional probability and therefore we avoid inverting a high dimensional matrix.

In our model, $E[Y_k(x)]$ is the response surface based on the evaluated points. Numerical experiments show that the newly built response surface still performs very well in surface fitting. The variance of $Y_k(x)$ is a measure about uncertainty of surface fitting at the point. It consists of two parts, one representing the uncertainty in surface fitting caused by the allocation of the visited points in the solution set and the other representing the uncertainty caused by the estimation error at the visited points. For points in the regions that have been densely evaluated, the first part will be small. Whereas for points in the regions that have rarely been estimated, we can prove that the first part will be strictly larger than 0. This is identical to our intuition that more points estimated in the region, less uncertainty in surface fitting. The second part of $\text{Var}[Y_k(x)]$ is strictly larger than 0 for any point and it is largely affected by the estimation error of the points close to it. This is also identical to our intuition that estimation error at a visited point will have largest impact on surface fitting at points near to it.

3.3 Sampling From the Sampling Distribution

The other problem is how to generate samples from $f_k(x)$ effectively. Note that

$$f_k(x) \leq c \cdot \frac{1}{m(\Theta)},$$

where $m(\Theta)$ is the Lebesgue measure of Θ and

$$c = \frac{m(\Theta)}{2} \frac{1}{\int_{\Theta} \Pr\{Y_k(z) > \hat{g}_{k-1}^*\} dz}.$$

We suggest using acceptance-rejection method to generate sample points. The detailed implementation of the acceptance rejection method is as follows.

Generate x from $f_k(x)$ by acceptance rejection method

Step 1. Generate a sample Z uniformly in Θ and U uniformly in $(0, 1)$.

Step 2. If $U \leq 2 \cdot \Pr\{Y_k(Z) > \hat{g}_{k-1}^*\}$, accept Z and set $x = Z$. Otherwise, go to Step 1.

The following proposition verifies the acceptance-rejection method. We omit its proof due to the space limit.

Proposition 2 For any $A \subseteq \Theta$,

$$\Pr\{x \in A\} = \int_A f_k(z) dz$$

where x is generated by acceptance rejection method.

4 GAUSSIAN PROCESS-BASED SEARCH ALGORITHM

In this section, we will give algorithms for deterministic continuous optimization problems and stochastic discrete optimization problems respectively.

4.1 GPS Algorithm for Deterministic Continuous Optimization Problems

We will now develop Gaussian process-based search algorithm for deterministic continuous optimization problems. Let Θ^* be the set of optimal points of g on Θ . First, we have the following assumption about the problem.

Assumption 1 The set Θ^* is non-empty and for any $\varepsilon > 0$, there exists $B(\delta_\varepsilon) = \{x \in \Theta : |x - x^*| < \delta_\varepsilon \text{ for some } x^* \in \Theta^*\}$, such that $\min_{x \in B(\delta_\varepsilon)} g(x) > g^* - \varepsilon$ and $B(\delta_\varepsilon)$ has a positive Lebesgue measure.

The GPS algorithm for deterministic continuous optimization problems is as follows.

Gaussian Process-based Search (GPS) Algorithm for Deterministic Continuous Optimization

Step 0. Set iteration count $k = 0$. Generate $x_{0,1}, \dots, x_{0,s}$ uniformly from Θ and set $\mathbb{S}_0 = \{x_{0,1}, \dots, x_{0,s}\}$.

Step 1. Evaluate $g(x)$ at $x_{k,1}, \dots, x_{k,s}$. Find out $g_k^* = \max_{x \in \mathbb{S}_k} g(x)$ and update $f_{k+1}(x)$ where

$$f_{k+1}(x) = \frac{\Pr\{Y_{k+1}(x) > g_k^*\}}{\int_{z \in \Theta} \Pr\{Y_{k+1}(z) > g_k^*\} dz}.$$

Step 2. Set $k = k + 1$. Independent of everything else, sample $x_{k,1}, \dots, x_{k,s}$ independently from $f_k(x)$. Let $\mathbb{S}_k = \mathbb{S}_{k-1} \cup \{x_{k,1}, \dots, x_{k,s}\}$. Go to Step 1.

To ensure the convergence of the GPS algorithm, we need the following conditions about the covariance function.

Condition 1: Conditions on correlation function

For any $x_1, x_2 \in \Theta$, the correlation function of the Gaussian process $Z(x)$ in (4) satisfies the following condition: $\gamma(x_1, x_2) = h(|x_1 - x_2|) \geq 0$ where $h(\cdot)$ is a decreasing function with respect to its variable on \mathbb{R}^+ , and for any $x_0, x_1, x_2 \in \Theta$, $h(|x_1 - x_2|) \geq h(|x_0 - x_1|) \cdot h(|x_0 - x_2|)$.

Remark 5 The correlation function in kriging metamodeling satisfies Condition 1 in most cases. For example, the exponential correlation function $h(x) = \exp(-\theta|x|)$ with $\theta > 0$ and Gaussian correlation function $h(x) = \exp(-\theta x^2)$ with $\theta > 0$ satisfy the condition.

Theorem 1 Suppose Condition 1 and Assumption 1 are satisfied. Then for any $\varepsilon > 0$,

$$\lim_{k \rightarrow \infty} \Pr\{|g_k^* - g^*| > \varepsilon\} = 0,$$

where g_k^* is generated by the GPS algorithm.

4.2 GPS Algorithm for Stochastic Discrete Optimization Problems

We will now develop the GPS algorithm for stochastic discrete optimization problems. At the very beginning, we have the following assumptions about $G(x)$.

Let $G_i(x)$ denote the i th observation of $G(x)$ on x . We make the following assumption about $G(x)$.

Assumption 2 For any $\varepsilon > 0$, there exists a positive number n^* such that for all $n \geq n^*$,

$$\sup_{x \in \Theta} \Pr \left\{ \left| \frac{1}{n} \sum_{i=1}^n G_i(x) - g(x) \right| \geq \varepsilon \right\} \leq \psi(n, \varepsilon),$$

where $\psi(n, \varepsilon)$ is a non-increasing function of n and ε . Moreover, $\psi(n, \varepsilon) \rightarrow 0$ as $n \rightarrow \infty$ for any fixed $\varepsilon > 0$, $\psi(n, \varepsilon) \rightarrow 0$ as $\varepsilon \rightarrow \infty$ for any fixed $n > 0$.

A similar assumption is also used by Hong and Nelson (2006) to analyze the convergence of their algorithm for discrete OvS problems. If $G_i(x)$, $i = 1, 2, \dots$, are independent and identically distributed, and there exist $p > 2$, $C_1 > 0$ and $C_2 > 0$ such that $C_1 < E[|G(x) - f(x)|^p] < C_2$, then by Markov's inequality, for any $\varepsilon > 0$

$$\Pr \left\{ \left| \frac{1}{n} \sum_{i=1}^n G_i(x) - g(x) \right| > \varepsilon \right\} < \frac{E[|\sum_{i=1}^n (G_i(x) - g(x))|^p]}{(n\varepsilon)^p}.$$

By Rosenthal's inequality,

$$E \left[\left| \sum_{i=1}^n (G_i(x) - g(x)) \right|^p \right] < C_p \max \left\{ nE[|G(x) - g(x)|^p], (n\text{Var}[G(x)])^{p/2} \right\}.$$

When n is large enough, we have

$$(n\text{Var}[G(x)])^{p/2} > (nC_1)^{p/2} > nC_2 > nE[|G(x) - g(x)|^p].$$

Therefore, when n is large enough

$$\Pr \left\{ \left| \frac{1}{n} \sum_{i=1}^n G_i(x) - g(x) \right| > \varepsilon \right\} < C_p \frac{(\text{Var}[G(x)])^{p/2}}{n^{p/2}\varepsilon^p} < \frac{C}{n^{p/2}\varepsilon^p}, \tag{5}$$

for some $C > 0$. Therefore, Assumption 2 is satisfied.

The GPS algorithm for stochastic discrete optimization problems is as follows.

Gaussian Process-based Search (GPS) Algorithm for Stochastic Discrete Optimization Problems

Step 0. Set iteration count $k = 0$.

Generate $x_{0,1}, \dots, x_{0,s}$ uniformly from Θ and set $\mathbb{S}_0 = \{x_{0,1}, \dots, x_{0,s}\}$.

Independent of everything else, take m_0 observations on $x_{0,i}$, $i = 1, \dots, s$.

Set $N_0(x_{0,i}) = m_0$ for $x \in \mathbb{S}_0$ and $N_0(x) = 0$ for all other points.

Calculate $\bar{G}_0(x_{0,i}) = \frac{1}{m_0} \sum_{j=1}^{m_0} G_j(x_{0,i})$, $i = 1, \dots, s$ and set $\bar{G}_0(x) = -\infty$ for all other points.

Find out x_0^* and \hat{g}_0^* .

Step 1. Let $k = k + 1$.

Construct

$$f_k(x) = \frac{\Pr\{Y_k(x) > \hat{g}_{k-1}^*\}}{\int_{z \in \Theta} \Pr\{Y_k(z) > \hat{g}_{k-1}^*\} dz}.$$

Independent of everything else, sample $x_{k,1}, \dots, x_{k,s}$ independently from $f_k(x)$. Set $\mathbb{S}_k = \mathbb{S}_{k-1} \cup \{x_{k,1}, \dots, x_{k,s}\}$.

Step 2. Independent of everything else, take m_k observations on $x_{k,1}, \dots, x_{k,s}$ and x_{k-1}^* .

Set $N_k(x) = N_{k-1}(x) + m_k$ and update $\bar{G}_k(x) = \frac{N_{k-1}(x)\bar{G}_{k-1}(x) + \sum_{j=1}^{m_k} G_j(x)}{N_k(x)}$, for $x = x_{k,1}, \dots, x_{k,s}$ and x_{k-1}^* .

Find out x_k^* and \hat{g}_k^* .

Go to Step 1.

To guarantee that the algorithm will converge to the global optimal solutions. We need the following condition about the sampling allocation rule (SAR).

Condition 2: Conditions on SAR

The sequence $\{m_k\}$ satisfies the following conditions:

- i. For any $\varepsilon > 0$, $\sum_{k=1}^{\infty} \psi(m_k, \varepsilon) < \infty$.
- ii. m_k is strictly increasing function of k .

Remark 6 If $G_i(x)$ are i.i.d observations at x and the p th moment ($p > 2$) of $G(x)$ exist. By letting $m_k = \lceil k^{2/p'} \rceil$ for any $p' < p$, Condition 2 is satisfied.

The following theorem shows the global convergence of the GPS algorithm. Recall that if $N_k(x) = 0$, we have defined $\bar{G}_k(x) = -\infty$. Following this convention we define $|\bar{G}_k(x) - g(x)| = \infty$ for any x with $N_k(x) = 0$.

Theorem 2 Suppose Assumption 2 and Conditions 1 and 2 are satisfied. Then for any $\varepsilon > 0$,

$$\lim_{k \rightarrow \infty} \Pr\{|\hat{g}_k^* - g^*| > \varepsilon\} = 0$$

where $\{g_k^*\}$ is generated by the GPS algorithm.

5 NUMERICAL EXAMPLES

In this section, we apply the GPS algorithm to a function that has 25 local optima. Let the OvS problem be

$$\max g(x_1, x_2), \quad 0 \leq x_1, x_2 \leq 100,$$

where $g(x_1, x_2) = 10 \cdot \frac{\sin^6(0.05\pi x_1)}{2^2 \left(\frac{x_1-90}{80}\right)^2} + 10 \cdot \frac{\sin^6(0.05\pi x_2)}{2^2 \left(\frac{x_2-90}{80}\right)^2}$.

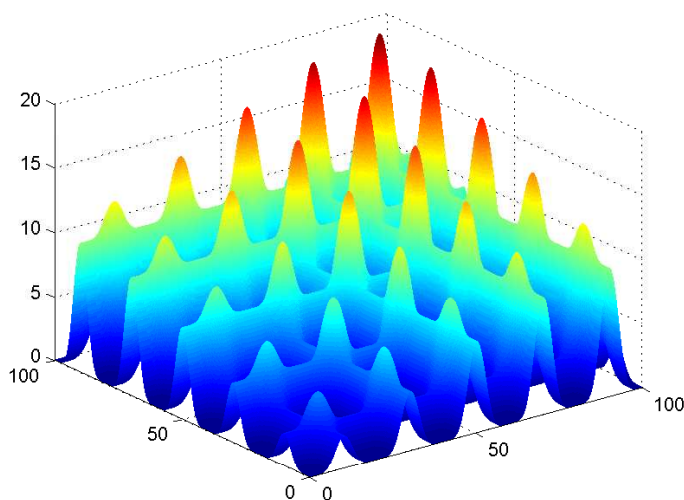


Figure 5: Function $g(x_1, x_2)$.

A similar example is also used by Xu et al. (2010). Note that g has 25 local optima with a global optimum $g(90, 90) = 20$. The second best value of g is 19.17 at $(90, 70)$ and $(70, 90)$ (see Figure 5). Without knowing the closed form, this problem is difficult to solve because it has 25 local optima and the difference between the values of the best and second best solutions is quite small.

We apply the GPS algorithm for deterministic optimization to solve this problem. For the algorithm, we set $\sigma^2 = 25$, $\gamma_i(x) = e^{-\|x-x_i\|^2}$ and $\lambda_i(x) = \frac{\|x-x_i\|^{-2}}{\sum_{j=1}^n \|x-x_j\|^{-2}}$ where $\|x-y\|$ denotes the Euclidean distance. The left panel of Figure 6 shows the 30 sample paths of GPS algorithm with a total sample size of 1000. We can see that the performance of the GPS algorithm is very satisfactory. To compare the GPS algorithm with other algorithms, we also use simulated annealing and pure random search algorithm to find the optimal value of g . The right panel of Figure 6 shows the average performances of the GPS algorithm, the simulated annealing algorithm and the pure random search algorithm over 30 sample paths. The simulated annealing algorithm we used from Ghate and Smith (2008) has a constant temperature of 0.1 and a set of neighborhood whose maximal distance in each dimension is smaller than 1. We can see that the performance of the GPS algorithm is much better than both the simulated annealing and the random search algorithms.

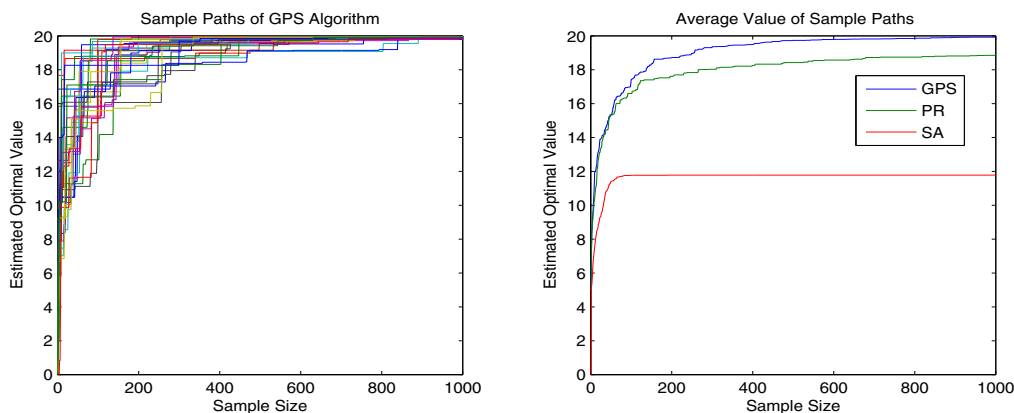


Figure 6: Performance of GPS algorithm.

To illustrate the performance of the GPS algorithm more clearly, Figure 7 shows the fitting surface, variance surface, un-normalized probability density function and the sampled points of GPS algorithm for 200, 500 and 1000 sampled points. Comparing these figures, we can see that the GPS algorithm can automatically balance exploitation and exploration very well. Within 200 iterations, GPS algorithm can identify both the promising regions and un-promising regions. As the number of points increases, the response surface gets closer to the true surface and the sampling distribution focuses more on the region that contains the global optimum.

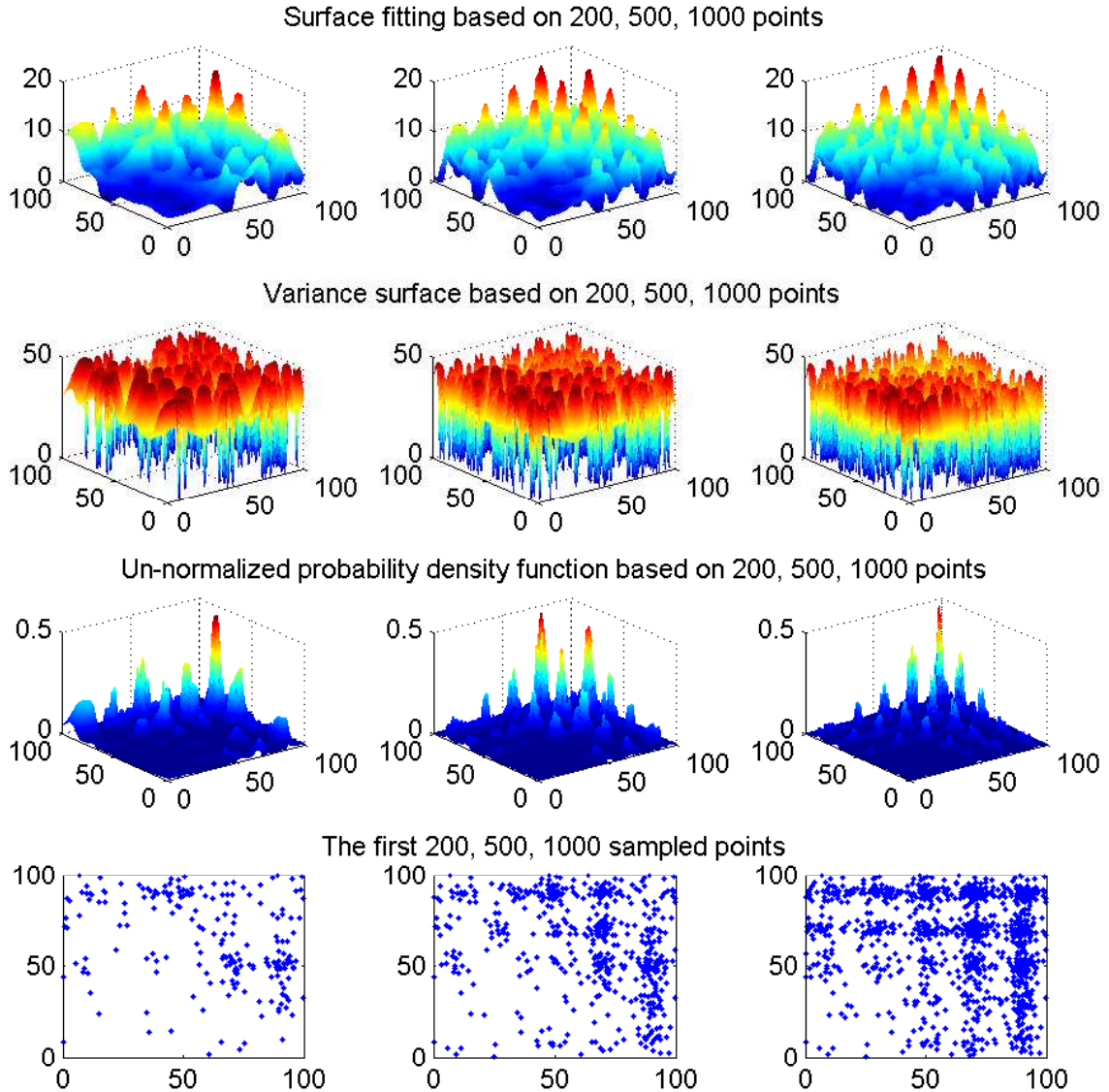


Figure 7: Mean surface, variance surface, un-normalized probability density function and sampled points of GPS algorithm based on 200, 500 and 1000 points.

ACKNOWLEDGMENTS

This research is partially supported by the Hong Kong Research Grants Council under grants GRF 613410, N_HKUST 626/10 and by the National Natural Science Foundation of China under grant No. 71101111.

REFERENCES

- Andradóttir, S., and A. A. Prudius. 2010. "Adaptive random search for continuous simulation optimization". *Naval Research Logistics* 58: 583-604.
- Ankenman, B., B. L. Nelson, and J. Staum. 2010. "Stochastic kriging for simulation metamodeling". *Operations Research* 58: 371-382.
- Ghate, A., and R. L. Smith. 2008. "Adaptive search with stochastic acceptance probabilities for global optimization". *Operations Research Letters* 36: 285-290.
- Hong, L. J., and B. L. Nelson. 2006. "Discrete optimization via simulation using COMPASS". *Operations Research* 54: 115-129.
- Hu, J., M. C. Fu, and S. I. Marcus. 2007. "A model reference method for global optimization". *Operations Research* 55: 549-568.
- Hu, J., M. C. Fu, and S. I. Marcus. 2008. "A model reference adaptive search method for stochastic global optimization". *Communications in Information and Systems* 8: 245-276.
- Jones, D. R., M. Schonlau, and W.J. Welch. 1998. "Efficient global optimization of expensive black-box functions". *Journal of Global Optimization* 13: 455-492.
- Shi, L., and S. Ólafsson. 2000. "Nested partitions method for global optimization". *Operations Research* 48: 390-407.
- Xu, J., B. L. Nelson, and L. J. Hong. 2010. "Industrial Strength COMPASS: A comprehensive algorithm and software for optimization via simulation". *ACM Transactions on Modeling and Computer Simulation* 20: 1-29.
- Yan, D., and H. Mukai. 1992. "Stochastic discrete optimization". *SIAM J. Control and Optimization* 30: 594-612.

AUTHOR BIOGRAPHIES

LIHUA SUN is an assistant professor in the Department of Economics and Finance at School of Economics and Management of Tongji University. Her research interests include simulation methodologies, financial engineering and simulation optimization. Her e-mail address is sunlihua@tongji.edu.cn.

L. JEFF HONG is a professor in the Department of Industrial Engineering and Logistics Management at The Hong Kong University of Science and Technology. His research interests include Monte Carlo simulation, financial risk management, and stochastic optimization. He is currently an associate editor of *Operations Research*, *Naval Research Logistics* and *ACM Transactions on Modeling and Computer Simulation*. His email address is hongl@ust.hk.

ZHAOLIN HU is an assistant professor in Department of Management Science and Engineering at School of Economics and Management of Tongji University. His research interests include Monte Carlo simulation and stochastic optimization. His email address is huzhaolin@gmail.com.