Information Sciences 471 (2019) 80–96

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

# A complete expected improvement criterion for Gaussian process assisted highly constrained expensive optimization

Ruwang Jiao[a], Sanyou Zeng[a,*], Changhe Li[b,c,*], Yuhong Jiang[d], Yaochu Jin[e]

[a] School of Mechanical Engineering and Electronic Information, China University of Geosciences, Wuhan 430074, China
[b] School of Automation, China University of Geosciences, Wuhan 430074, China
[c] Hubei Key Laboratory of Advanced Control and Intelligent Automation for Complex Systems, China University of Geosciences, Wuhan 430074, China
[d] School of Computer Science, China University of Geosciences, Wuhan 430074, China
[e] Department of Computer Science, University of Surrey, Guildford GU27XH, UK

ARTICLE INFO

*Article history:*
Received 8 May 2018
Revised 29 August 2018
Accepted 1 September 2018
Available online 4 September 2018

*Keywords:*
Evolutionary computation
Expensive optimization
Expected improvement
Constrained optimization
Gaussian process

ABSTRACT

Expected improvement (EI) is a popular infill criterion in Gaussian process assisted optimization of expensive problems for determining which candidate solution is to be assessed using the expensive evaluation method. An EI criterion for constrained expensive optimization (constrained EI) has also been suggested, which requires that feasible solutions exist in the candidate solutions. However, the constrained EI criterion will fail to work in case there are no feasible solutions. To address the above issue, this paper proposes a new EI criterion for highly constrained optimization that can work properly even when no feasible solution is available in the current population. The proposed constrained EI criterion can not only exploit local feasible regions, but also explore infeasible yet promising regions, making it a complete constrained EI criterion. The complete constrained EI is theoretically validated and empirically verified. Simulation results demonstrate that the proposed complete constrained EI is better than or comparable to five existing infill criteria.



## 1. Introduction

Most of the science and engineering optimization problems in the real world are highly constrained. These constrained optimization problems (COPs) present serious challenges to existing optimization techniques. A general COP [36,46] can be defined as:

$$
\begin{array}{ll}
\min & y = f(\mathbf{x}) \\
\text{st}: & \mathbf{l} \le \mathbf{g}(\mathbf{x}) = (g_1(\mathbf{x}), g_2(\mathbf{x}), \dots, g_m(\mathbf{x})) \le \mathbf{u} \\
\text{where} & \mathbf{l} = (l_1, l_2, \dots, l_m), \mathbf{u} = (u_1, u_2, \dots, u_m); \\
& \mathbf{x} = (x_1, x_2, \dots, x_n) \in \mathbf{X} \\
& \mathbf{X} = \{\mathbf{x} | \mathbf{x_l} \le \mathbf{x} \le \mathbf{x_u}\} \\
& \mathbf{x_l} = (x_{l1}, x_{l2}, \dots, x_{ln}), \mathbf{x_u} = (x_{u1}, x_{u2}, \dots, x_{un})
\end{array}
\tag{1}
$$

* Corresponding authors.
*E-mail addresses:* sanyouzeng@gmail.com (S. Zeng), lichanghe@cug.edu.cn (C. Li).

https://doi.org/10.1016/j.ins.2018.09.003
0020-0255/

where $\mathbf{x}$ is a solution vector (solution for short) within the solution space $\mathbf{X}$, $\mathbf{g}(\mathbf{x})$ are constraints, $\mathbf{l}$ and $\mathbf{u}$ denote the lower and upper constraint bounds, respectively. If a solution $\mathbf{x} \in \mathbf{X}$ satisfies all constraints $\mathbf{g}(\mathbf{x})$, it is called a feasible solution; otherwise, it is called an infeasible solution.

Due to the inherent characteristics of gradient-free and insensitivity to the local optimal, evolutionary algorithms (EAs) are much preferable for various complex and non-convex optimization problems, including COPs. Constraint handling techniques based on EAs can be categorized as follows [30,47]: feasibility rules, stochastic ranking, $\varepsilon$ constrained method, novel penalty functions, novel special operators, multi-objective concepts and ensemble of constraint-handling techniques.

Nevertheless, many engineering optimization problems require expensive computer simulations or physical experiments for evaluating candidate solutions, such as in wind turbine design [34], drug design [44], antenna design [18] and aerodynamic design [21]. Traditional EAs cannot directly solve them since a large number of function evaluations is unaffordable for this kind of problems. To address this issue, surrogate-assisted EAs (SAEAs) have been developed, where part of expensive fitness evaluations are replaced by computationally cheap approximate models often known as surrogates or meta-models. In the optimization process, computationally expensive fitness functions are replaced by some previously built surrogate models based on historical data, so that the cost of the time-consuming or resource-consuming fitness functions can be reduced.

The Gaussian process, also known as Kriging in traditional design optimization, is the most popular model when compared to others because of its ability to provide uncertainty estimation for the approximated values, and it has been increasingly employed as surrogates in evolutionary single and multi-objective optimization [2,32]. After building a GP model, how to manage the tradeoff between the accuracy and the uncertainty of surrogates is the main issue in GP-assisted EAs. Infill sampling criteria make use of the estimates of fitness and estimated uncertainty (also known as confidence level) to assess the value of a solution with respect to the optimality and uncertainty. If a point is expected to be promising according to an infill sampling criterion, it will be selected to be evaluated using the real expensive fitness function. Maximizing the expected improvement (EI) [22] is a widely-used sampling strategy used in selecting sample solutions for updating GP models. Using EI is advantageous since it is likely to be larger at unsampled areas or at under sampled areas near the global optimum and offers solutions with both exploration and exploitation of the GP model.

For expensive COPs, Schonlau [38] proposed a constrained EI by maximizing the multiplication of the EI and the probability feasibility (PF), which are both statistical measures determined by GP models of fitness and constraints. The constrained EI is based on the current best feasible solution. However, for highly constraint functions with small feasible regions, e.g., the well-known constrained benchmark test suite IEEE CEC2006 [26] consists of 24 problems, but 19 of them with the feasible ratio less than 1%, so using surrogates for these problems to obtain a feasible solution can be very challenging. When a feasible solution is not provided in the sampling data, the existing constrained EI no longer work, which means they are incomplete. To fill this gap, this study introduces a complete constrained EI as infill sampling criterion for efficiently dealing with computationally expensive COPs. The motivation of this paper is to adopt EI of constraint violation to reach feasible regions. The preliminary idea of the EI of constraint violation has presented in [19]. Note that in this paper, the objective and constraints are assumed mutually independent .

New contributions of the paper are summarized as follows:

(1) This paper is the first attempt to propose the idea that concentrating on EI of constraint violation to deal with highly constrained problems where no feasible solution is available in the sampling data for an expensive COP. Different from the maximization of the feasibility probability [3,15] in the case of no feasible point available, the proposed method adopts the EI of constraint violation as the metric for selecting a new potential solution. The level of constraint violation of a solution reflects the distance to the feasible space, hence it is often employed to handle the constraint difficulty. In addition, the maximum EI value enables the GP model to efficiently explore the optimum as well as improve the model accuracy in single-objective optimization [48]. Maximizing the EI of constraint violation will drive the search towards promising feasible regions.

(2) A suitable formulation of COP in Eq. (1) is suggested for the GP-assisted expensive optimization since the widely-used typical formulation is not suitable, since it needs to introduce additional constraints and dependencies among the objective and constraints. Handling the additional constraints and dependencies would cost additional computational resource, especially the additional dependencies are likely to degrade the performance of expensive optimization technologies, since most technologies are under the assumption of mutual independency among the objective and constraints.

The remainder of this article is organized as follows. The related work is briefly discussed in Section 2. A brief description of related techniques is provided in Section 3. The proposed method is introduced and theoretically discussed in Section 4. A surrogate-assisted evolutionary algorithm framework is presented in Section 5. Numerical results on benchmark problems and comparison with five existing infill sampling criteria are presented in Section 6. Finally, conclusions and future work are discussed in Section 7.

## 2. Related work

A lot of efforts and progress have been made in developing the surrogate-based EAs. Many machine learning models can be utilized to build surrogates, including: Gaussian process (GP) [22], multivariate polynomials (particularly quadratic models) [13], artificial neural networks [17,33], radial basis function (RBF) [23,41,45], support vector machines (SVMs) [16] and hybrid models [5,24,27]. A comprehensive survey with detailed discussions of the existing techniques for SAEAs can be found in [20].

For COPs, some attempts have been made to handle constraints with expensive cost. Basudhar [4] introduced an efficient global optimization algorithm for COPs, where GP and SVMs are applied for approximation of the objective function and the boundary of feasible regions, respectively. Several correlated constraints can be represented by a unique SVM, which can simplify the complexity of COPs. Regis et. al[35]. proposed a surrogate-assisted evolutionary programming algorithm for expensive inequality COPs. They use a cubic RBF to model the objective and each constraints. However, all test problems are given a feasible starting solution in advance in their experiment. In [7], a surrogate-assisted multi-objective evolution strategy (SMES) was developed for handling computationally expensive multi-objective optimization problems with constraints. One limitation of the SMES is that it is not expected to work well on problems where the interior of the feasible region is empty.

The original EI [22] based on GP model is only for unconstrained optimization problems. Schonlau [38] combined EI infill sampling criterion with the probability feasibility to deal with computationally expensive COPs. This criterion fails for highly constrained problems,or when the global optimum is near the boundary. In [1], an expected violation was introduced where a threshold on violation was used instead of the actual violation, thereby it allows a greater possibility of sampling around constraint boundaries. In [11], the authors discussed three modified infill sampling criteria for COPs: probability of improvement, EI and lower confidence bound (LCB). Durantin et al. [10] proposed a three-objective method which took into account the EI, PF and prediction accuracy of constraints.

**Discussions:** the aforementioned studies have made some efforts to deal with COPs with computationally expensive objective and constraints. However, they are all based on the assumption that a feasible candidate solution is available. These techniques will fail when dealing with highly constrained problems where none of the initially sampled solutions is feasible. Apparently, developing a complete constrained EI to handle situations where no feasible solution is available is highly in demand. To address this difficulty, this paper proposes a complete constrained EI infill sampling criterion to tackle expensive COPs. When a feasible point is not available, the EI of constraint violation is adopted to approach the feasible region for the purpose of obtaining a feasible point as soon as possible.

## 3. Background

In this section, we provide a brief theoretical overview of GP, EI and existing criteria of constrained EI.

### 3.1. Gaussian process

To model an unknown function $f(\mathbf{x})$, GP assumes that $f(\mathbf{x})$ at any point $\mathbf{x}$ is a Gaussian random variable $F(\mathbf{x})\sim N(\mu, \sigma^2)$, where $\mu$ and $\sigma$ are two constants which are independent of $\mathbf{x}$. For any $\mathbf{x}$, $f(\mathbf{x})$ is denoted as a sample of $F(\mathbf{x})$. Unlike deterministic models, GP provides an estimate of the fitness (mean) together with an estimate of the uncertainty (variance):

$$\hat{f}(\mathbf{x}) = \hat{\mu} + \mathbf{r}^T C^{-1}(f^N - \mathbf{1}\hat{\mu}) \tag{2}$$

$$s^2(\mathbf{x}) = \hat{\sigma}^2\left[1 - \mathbf{r}^T C^{-1}\mathbf{r} + \frac{(1-\mathbf{1}^T C^{-1}\mathbf{r})^2}{\mathbf{1}^T C^{-1}\mathbf{1}}\right] \tag{3}$$

where $\mathbf{r}$ is the $N \times 1$ vector of correlations between $F(\mathbf{x})$ and $F(\mathbf{x}^{(i)})$ for $i = 1, \dots, N$, $\mathbf{C}$ is a $N \times N$ matrix whose $(i, j)$-element is correlations between $F(\mathbf{x}^{(i)})$ and $F(\mathbf{x}^{(j)})$. $N(\hat{f}(\mathbf{x}), s^2(\mathbf{x}))$ is regarded as a predictive distribution for $F(\mathbf{x})$ on the tested $F^N = f^N$ at the points $\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(N)}$ (e.g., $\hat{f}(\mathbf{x})$ is predicted).

### 3.2. Expected improvement (EI)

After building a GP model, a metric for measuring the merit of evaluating at a new untested point should be defined. EI has been proposed in [22] to balance exploitation and exploration of the predictive distribution model.

Suppose $N(\hat{f}(\mathbf{x}), s^2(\mathbf{x}))$ is a surrogate model for an objective function $F(\mathbf{x})$ on the tested $f^N = (f(\mathbf{x}^{(1)}), \dots, f(\mathbf{x}^{(N)}))^T$ at $\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(N)}$, and the best value of $F(\mathbf{x})$ over all the $f^N$ is $f_{\min}$ (in an unconstrained problem). The improvement of $F(\mathbf{x})$ at an untested point $\mathbf{x}$ is

$$I(\mathbf{x}) = \max\{f_{\min} - F(\mathbf{x}), 0\} \tag{4}$$

Thus, the EI on $F^N = f^N$ is

$$\begin{aligned} E[I(\mathbf{x})|f^N] &= E[\max\{f_{\min} - F(\mathbf{x}), 0\}|f^N] \\ &= (f_{\min} - \hat{f}(\mathbf{x}))\Phi\left(\frac{f_{\min}-\hat{f}(\mathbf{x})}{s(\mathbf{x})}\right) + s(\mathbf{x})\phi\left(\frac{f_{\min}-\hat{f}(\mathbf{x})}{s(\mathbf{x})}\right) \end{aligned} \tag{5}$$

where the predicted expectation $\hat{f}(\mathbf{x})$ and the square root of the predicted variance $s(\mathbf{x})$ are calculated according to Eqs. (2) and (3). $\Phi(\cdot)$ is the standard normal cumulative distribution function and $\phi(\cdot)$ is the standard normal probability density function.

The value of $E[I(\mathbf{x})|f^N]$ will increase with either the increase in $f_{\min} - \hat{f}(\mathbf{x})$ or $s(\mathbf{x})$, maximizing $E[I(\mathbf{x})|f^N]$ achieves a trade-off between exploration and exploitation.

### *3.3. Constrained expected improvement*

While the EI formulation in Eq. (5) is for unconstrained optimization problems, it is common to use a penalized form of EI [38] to deal with computationally expensive COPs. For COPs, the goal is to minimize $f(\mathbf{x})$ subject to $\mathbf{x}$ satisfying $m$ constraints $l_i \le g_i(\mathbf{x}) \le u_i$ for $i = 1, \ldots, m$. Assume the objective function $F(\mathbf{x})$ and constraints functions $G_i(\mathbf{x}) (i = 1, \ldots, m)$ are Gaussian processes. All $F(\mathbf{x})$ and $G_i(\mathbf{x})$ are mutually independent in this paper. For consistency, the following notations are introduced.

Given a point $\mathbf{x}$, a Gaussian random vector is denoted:

$$\mathbf{G}(\mathbf{x}) = (G_1(\mathbf{x}), \ldots, G_m(\mathbf{x})).$$

Given $N$ points $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)}$, we denote Gaussian random vectors:

$$F^N = (F(\mathbf{x}^{(1)}), \ldots, F(\mathbf{x}^{(N)}))^T,$$

$$G_1^N = (G_1(\mathbf{x}^{(1)}), \ldots, G_1(\mathbf{x}^{(N)}))^T,$$

$$\ldots,$$

$$G_m^N = (G_m(\mathbf{x}^{(1)}), \ldots, G_m(\mathbf{x}^{(N)}))^T \text{ and}$$

$$G^{mN} = (G_1(\mathbf{x}^{(1)}), \ldots, G_m(\mathbf{x}^{(1)}), \ldots, G_1(\mathbf{x}^{(N)}), \ldots, G_m(\mathbf{x}^{(N)}))^T.$$

And their tested function values:

$$f^N = (f(\mathbf{x}^{(1)}), \ldots, f(\mathbf{x}^{(N)}))^T,$$

$$g_1^N = (g_1(\mathbf{x}^{(1)}), \ldots, g_1(\mathbf{x}^{(N)}))^T,$$

$$\ldots,$$

$$g_m^N = (g_m(\mathbf{x}^{(1)}), \ldots, g_m(\mathbf{x}^{(N)}))^T \text{ and}$$

$$g^{mN} = (g_1(\mathbf{x}^{(1)}), \ldots, g_m(\mathbf{x}^{(1)}), \ldots, g_1(\mathbf{x}^{(N)}), \ldots, g_m(\mathbf{x}^{(N)}))^T$$

are regarded as samples of $F^N$, $G_1^N$, ... , $G_m^N$ and $G^{mN}$.

In this paper, $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)}$ are regarded as tested points while $\mathbf{x}$ is regarded as an untested point. In the same way as Section 3.1, one can predict $f(\mathbf{x})$, $g_i(\mathbf{x}) (i = 1, 2, \ldots, m)$ at any untested point $\mathbf{x}$ based on the $f^N$, $g_i^N (i = 1, 2, \ldots, m)$ at tested points $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)}$.

The improvement of the objective under satisfying constraints is defined

$$I_{c,N}(\mathbf{x}) = \begin{cases} f_{\min}^N - F(\mathbf{x}), & F(\mathbf{x}) \le f_{\min}^N \text{ and } l_i \le G_i(\mathbf{x}) \le u_i, \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

where $f_{\min}^N$ is the best objective function value over the feasible points in the tested $N$ points. It can be seen that any constraint violation leads to zero improvement.

With the assumption of the mutual independence between $F(\mathbf{x})$ and $G_i(\mathbf{x}) (i = 1, \ldots, m)$, the **constrained expected improvement (constrained EI)** based on $F^N = f^N$ and $G^{mN} = g^{mN}$ is given by [38]

$$E[I_{c,N}(\mathbf{x})|f^N, g^{mN}] = E[I(\mathbf{x})|f^N] \times \prod_{i=1}^{m} P\{l_i \le G_i(\mathbf{x}) \le u_i|g_i^N\} \tag{7}$$

where $E[I(\mathbf{x})|f^N]$ is calculated by using Eq. (5), and where $\prod_{i=1}^{m} P\{l_i \le G_i(\mathbf{x}) \le u_i|g_i^N\}$ is the probability of feasibility (PF).

Eq. (7) transforms the constrained EI into multiplying the EI by the probability of feasibility of a solution. The magnitude of constrained EI will be driven to zero where there is a very low probability of feasibility for any of the constraints. One concern often noticed is that $f_{\min}^N$ indicates the minimum feasible objective value in tested points. Once there are no feasible points in the tested points, the constrained EI in Eq. (7) is no longer applicable. We would address this issue in the following section.

## 4. Proposed constrained expected improvement criterion

The constrained EI proposed in this paper is discussed for two situations where feasible solutions are not available and available, respectively.

### 4.1. When no feasible solutions are available

If a feasible solution is not available, the existing constrained EI does not work any more. To address this difficulty, we need to find a feasible point as soon as possible. We consider EI of constraint violation as a new infill sampling criterion for obtaining a better update solution with the smaller constraint violation.

In this paper, the constraint violation of a solution $\mathbf{x}$ is defined as the maximum of the violations of all constraints

$$G^+(\mathbf{x}) = \max_{i=1,2,\dots,m} \{G_i^+(\mathbf{x})\} \tag{8}$$

where $G_i^+(\mathbf{x}) = \max\{0, l_i - G_i(\mathbf{x}), G_i(\mathbf{x}) - u_i\}$ is the violation of $i$th constraint. Note that $G_i^+(\mathbf{x})$ and $G^+(\mathbf{x})$ do not satisfy Gaussian distribution any more. If a sampled or tested $G^+(\mathbf{x}) = 0$, it means $\mathbf{x}$ satisfies all constraints and thus $\mathbf{x}$ is feasible, otherwise $\mathbf{x}$ is infeasible.

#### 4.1.1. Formulation of constrained EI

In case there are no feasible solutions, the constrained improvement is the improvement of constraint violation at an untested point $\mathbf{x}$ beyond the current best solution defined as

$$I_{c,N}(\mathbf{x}) = \begin{cases} g_{\min}^{+N} - G^+(\mathbf{x}), & if\ G^+(\mathbf{x}) \le g_{\min}^{+N} \\ 0, & \text{otherwise} \end{cases} \tag{9}$$

where $g_{\min}^{+N}$ is the current best constraint violation of all the tested $N$ solutions.

The formulation of the constrained EI of a solution $\mathbf{x}$ is as follows

$$\begin{aligned} & E[I_{c,N}(\mathbf{x})|g^{mN}] = \int_0^{g_{\min}^{+N}} P_{G^+(\mathbf{x})|g^{mN}}(z)dz - g_{\min}^{+N} \times P_{G^+(\mathbf{x})|g^{mN}}(0) \\ & \text{where} \\ & P_{G^+(\mathbf{x})|g^{mN}}(z) = \prod_{i=1}^{m} \left[ \Phi\left(\frac{u_i + z - \hat{g}_i(\mathbf{x})}{s_{g_i}(\mathbf{x})}\right) - \Phi\left(\frac{l_i - z - \hat{g}_i(\mathbf{x})}{s_{g_i}(\mathbf{x})}\right) \right] \end{aligned} \tag{10}$$

where $\hat{g}_i(\mathbf{x})$ and $s_{g_i}^2(\mathbf{x})$ $(i = 1, \dots, m)$ are the expectations and variances of the $i$th constraint respectively, which are calculated by same ways of Eqs. (2) and (3). $\Phi(\cdot)$ is the standard normal cumulative distribution function, which can be expressed as follows

$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} \exp\left(-\frac{t^2}{2}\right) dt \tag{11}$$

Maximizing the EI of constraint violation is simply to reduce the constrained violation of potential solutions. Consequently, it can drive the search towards feasible regions.

#### 4.1.2. Theoretical derivation of constrained EI

The computation of the cumulative distribution function $P_{G^+(\mathbf{x})}(z)$ is divided into two cases in terms of the value of $z$.

- If $z < 0$:
  the probability of $\{G^+(\mathbf{x}) \le z\}|g^{mN}$ is 0,
  then the cumulative distribution function is $P_{G^+(\mathbf{x})|g^{mN}}(z) = 0$.
- If $z \ge 0$:
  the cumulative distribution function of $G^+(\mathbf{x})$ under the condition $G^{mN} = g^{mN}$ is

$$\begin{aligned} P_{G^+(\mathbf{x})|g^{mN}}(z) &= P\{\{G^+(\mathbf{x}) \le z\}\} \\ &= P\{[l_1 - G_1(\mathbf{x}) \le z] \cap [G_1(\mathbf{x}) - u_1 \le z] \cap \dots \cap [l_m - G_m(\mathbf{x}) \le z] \cap [G_m(\mathbf{x}) - u_m \le z]\} \\ &= P\{[l_1 - z \le G_1(\mathbf{x}) \le u_1 + z] \cap \dots \cap [l_m - z \le G_m(\mathbf{x}) \le u_m + z]\} \\ &= \int_{l_1 - z}^{u_1 + z} dg_1 \cdots \int_{l_m - z}^{u_m + z} dg_m \times p_{\mathbf{G}(\mathbf{x})|g^{mN}}(g_1, \dots, g_m) \end{aligned} \tag{12}$$

where $l_i$ and $u_i$ are the lower and upper constraint boundaries, respectively. $p_{\mathbf{G}(\mathbf{x})|g^{mN}}(g_1, \dots, g_m)$ denotes the joint Gaussian probability density function of $m$ random variables $\mathbf{G}(\mathbf{x})$ under the condition $G^{mN} = g^{mN}$. Since we assume that the objective and each constraint are mutually independent, the joint Gaussian probability density function $p_{\mathbf{G}(\mathbf{x})|\mathbf{G}^N}(g_1, \dots, g_m)$ can be calculated by multiplying them separately

$$p_{\mathbf{G}(\mathbf{x})|g^{mN}}(g_1, \dots, g_m) = \prod_{i=1}^{m} p_{G_i(\mathbf{x})|g_i^N}(g_i)$$

$$= \prod_{i=1}^{m} \frac{1}{s_{g_i}(\mathbf{x})} \phi\left(\frac{g_i - \hat{g}_i(\mathbf{x})}{s_{g_i}(\mathbf{x})}\right). \tag{13}$$

The $p_{\mathbf{G}(\mathbf{x})|g^{mN}}(g_1, \dots, g_m)$ in Eq. (12) is substituted by Eq. (13), then we obtain the cumulative distribution function

$$\begin{aligned} P_{G^+(\mathbf{x})|g^{mN}}(z) &= \int_{l_1-z}^{u_1+z} dg_1 \cdots \int_{l_m-z}^{u_m+z} dg_m \times \prod_{i=1}^{m} \frac{1}{s_{g_i}(\mathbf{x})} \phi\left(\frac{g_i - \hat{g}_i(\mathbf{x})}{s_{g_i}(\mathbf{x})}\right) \\ &= \prod_{i=1}^{m} \int_{l_i-z}^{u_i+z} \frac{1}{s_{g_i}(\mathbf{x})} \phi\left(\frac{g_i - \hat{g}_i(\mathbf{x})}{s_{g_i}(\mathbf{x})}\right) dg_i \\ &= \prod_{i=1}^{m} \left[\Phi\left(\frac{u_i + z - \hat{g}_i(\mathbf{x})}{s_{g_i}(\mathbf{x})}\right) - \Phi\left(\frac{l_i - z - \hat{g}_i(\mathbf{x})}{s_{g_i}(\mathbf{x})}\right)\right] \end{aligned} \tag{14}$$

Subsequently, the EI of constraint violation can be obtained

$$\begin{aligned} E[I_{c,N}(\mathbf{x})|g^{mN}] &= \int_0^{g_{\min}^{+N}} (g_{\min}^{+N} - z) \times p_{G^+(\mathbf{x})|g^{mN}}(z) dz \\ &= \int_0^{g_{\min}^{+N}} (g_{\min}^{+N} - z) dP_{G^+(\mathbf{x})|g^{mN}}(z) \\ &= \int_0^{g_{\min}^{+N}} P_{G^+(\mathbf{x})|g^{mN}}(z) dz - g_{\min}^{+N} \times P_{G^+(\mathbf{x})|g^{mN}}(0) \end{aligned} \tag{15}$$

where $P_{G^+(\mathbf{x})|g^{mN}}(\cdot)$ is the cumulative distribution function in Eq. (14). The definition of Eq. (15) is respect to the prediction in the Gaussian random field and it measures how much degree of constraint violation can be achieved by evaluating the new point, considering the uncertainty of constraints.

### 4.2. When feasible solutions are available

If feasible points are already existed in the tested data, the goal in this situation is to improve the objective value in feasible regions. The EI of the objective under the feasible condition will be maximized in order to obtain a feasible solution with the best objective value.

#### 4.2.1. Formulation of constrained EI

The constrained EI of a solution **x** can calculated by using (7). The formulation of the constrained EI in this situation is

$$\begin{aligned} E[I_{c,N}(\mathbf{x})|f^N, g^{mN}] = &\left[(f_{\min}^N - \hat{f}(\mathbf{x}))\Phi\left(\frac{f_{\min}^N - \hat{f}(\mathbf{x})}{s_f(\mathbf{x})}\right) + s_f(\mathbf{x})\phi\left(\frac{f_{\min}^N - \hat{f}(\mathbf{x})}{s_f(\mathbf{x})}\right)\right] \\ &\times \prod_{i=1}^{m}\left[\Phi\left(\frac{u_i - \hat{g}_i(\mathbf{x})}{s_{g_i}(\mathbf{x})}\right) - \Phi\left(\frac{l_i - \hat{g}_i(\mathbf{x})}{s_{g_i}(\mathbf{x})}\right)\right] \end{aligned} \tag{16}$$

where $f_{\min}^N$ is the best feasible fitness over all the tested $N$ points, $\phi(\cdot)$ is the standard normal probability density function, $\Phi(\cdot)$ is the standard normal cumulative distribution function, and $\hat{f}(\mathbf{x})$, $\hat{g}_i(\mathbf{x})$, $s_f(\mathbf{x})$ and $s_{g_i}(\mathbf{x})$ $(i = 1, \dots, m)$ are the expectations and variances of the objective and constraints respectively, which are calculated by Eqs. (2) and (3).

#### 4.2.2. Theoretical analysis of constrained EI

$p_{F(\mathbf{x})\mathbf{G}(\mathbf{x})|f^N, g^{mN}}(f, g_1, \dots, g_m)$ denotes the joint Gaussian probability density function of random variable $F(\mathbf{x})$ and random vector $\mathbf{G}(\mathbf{x})$ on conditions $F^N = f^N$ and $G^{mN} = g^{mN}$. With the assumption of the mutual independence among $F(\mathbf{x})$ and $G_i(\mathbf{x}) (i = 1, \dots, m)$, the joint Gaussian probability density function has the form

$$\begin{aligned} p_{F(\mathbf{x})\mathbf{G}(\mathbf{x})|f^N, g^{mN}}(f, g_1, \dots, g_m) &= p_{F(\mathbf{x})|f^N}(f) \times p_{G_1(\mathbf{x})|g_1^N}(g_1) \times \dots \times p_{G_m(\mathbf{x})|g_m^N}(g_m) \\ &= \frac{1}{s_f(\mathbf{x})}\phi\left(\frac{f - \hat{f}(\mathbf{x})}{s_f(\mathbf{x})}\right) \times \prod_{i=1}^{m} \frac{1}{s_{g_i}(\mathbf{x})}\phi\left(\frac{g_i - \hat{g}_i(\mathbf{x})}{s_{g_i}(\mathbf{x})}\right) \end{aligned} \tag{17}$$

According to the definition of improvement of objective when all constraints are satisfied in Eq. (6), we have the EI as follows

$$E[I_{c,N}(\mathbf{x})|f^N, g^{mN}] = \int_{-\infty}^{f_{\min}^N} df \int_{l_1}^{u_1} dg_1 \cdots \int_{l_m}^{u_m} dg_m (f_{\min}^N - f) \times p_{F(\mathbf{x})G^+(\mathbf{x})|f^N, g^{mN}}(f, g_1, \dots, g_m) \tag{18}$$

Substituting $p_{F(\mathbf{x})\mathbf{G}(\mathbf{x})|f^N,g^{mN}}(f, g_1, \ldots, g_m)$ in Eq. (18) with Eq. (17), we have

$$\begin{aligned} E[I_{c,N}(\mathbf{x})|f^N, g^{mN}] &= \int_{-\infty}^{f_{\min}^N} (f_{\min}^N - f)\frac{1}{s_f(\mathbf{x})}\phi\left(\frac{f-\hat{f}(\mathbf{x})}{s_f(\mathbf{x})}\right)df \times \prod_{i=1}^{m}\int_{l_i}^{u_i}\frac{1}{s_{g_i}(\mathbf{x})}\phi\left(\frac{g_i-\hat{g}_i(\mathbf{x})}{s_{g_i}(\mathbf{x})}\right)dg_i \\ &= E[I(\mathbf{x})|f^N] \times \prod_{i=1}^{m}\left[\Phi\left(\frac{u_i-\hat{g}_i(\mathbf{x})}{s_{g_i}(\mathbf{x})}\right) - \Phi\left(\frac{l_i-\hat{g}_i(\mathbf{x})}{s_{g_i}(\mathbf{x})}\right)\right] \end{aligned} \tag{19}$$

Substituting $E[I(\mathbf{x})|f^N]$ in Eq. (19) with Eq. (5), we obtain

$$\begin{aligned} E[I_{c,N}(\mathbf{x})|f^N, g^{mN}] = &\left[(f_{\min}^N - \hat{f}(\mathbf{x}))\Phi\left(\frac{f_{\min}^N-\hat{f}(\mathbf{x})}{s_f(\mathbf{x})}\right) + s_f(\mathbf{x})\phi\left(\frac{f_{\min}^N-\hat{f}(\mathbf{x})}{s_f(\mathbf{x})}\right)\right] \\ &\times \prod_{i=1}^{m}\left[\Phi\left(\frac{u_i-\hat{g}_i(\mathbf{x})}{s_{g_i}(\mathbf{x})}\right) - \Phi\left(\frac{l_i-\hat{g}_i(\mathbf{x})}{s_{g_i}(\mathbf{x})}\right)\right] \end{aligned} \tag{20}$$

Overall, the complete constrained EI of a solution $\mathbf{x}$ is summarized as

**When no feasible solutions are available:**

$$E[I_{c,N}(\mathbf{x})|g^{mN}] = \int_0^{g_{\min}^{+N}} P_{G^+(\mathbf{x})|g^{mN}}(z)dz - g_{\min}^{+N} \times P_{G^+(\mathbf{x})|g^{mN}}(0)$$

where

$$P_{G^+(\mathbf{x})|g^{mN}}(z) = \prod_{i=1}^{m}\left[\Phi\left(\frac{u_i+z-\hat{g}_i(\mathbf{x})}{s_{g_i}(\mathbf{x})}\right) - \Phi\left(\frac{l_i-z-\hat{g}_i(\mathbf{x})}{s_{g_i}(\mathbf{x})}\right)\right] \tag{21}$$

**When feasible solutions are available:**

$$\begin{aligned} &E[I_{c,N}(\mathbf{x})|f^N, g^{mN}] \\ &= \left[(f_{\min}^N - \hat{f}(\mathbf{x}))\Phi\left(\frac{f_{\min}^N-\hat{f}(\mathbf{x})}{s_f(\mathbf{x})}\right) + s_f(\mathbf{x})\phi\left(\frac{f_{\min}^N-\hat{f}(\mathbf{x})}{s_f(\mathbf{x})}\right)\right] \times \prod_{i=1}^{m}\left[\Phi\left(\frac{u_i-\hat{g}_i(\mathbf{x})}{s_{g_i}(\mathbf{x})}\right) - \Phi\left(\frac{l_i-\hat{g}_i(\mathbf{x})}{s_{g_i}(\mathbf{x})}\right)\right] \end{aligned}$$

We should notice a special case, which has been widely used in evolutionary computation field, that the lower constraint bound $\mathbf{l} = -\infty$ and upper constraint bound $\mathbf{u} = \mathbf{0}$. Then COP in Eq. (1) has the form

$$\begin{array}{ll} \min & y = f(\mathbf{x}) \\ \text{st}: & \mathbf{g}(\mathbf{x}) = (g_1(\mathbf{x}), \ldots, g_m(\mathbf{x})) \leq \mathbf{0} \\ \text{where} & \mathbf{x} = (x_1, \ldots, x_n) \in \mathbf{X} \\ & \mathbf{X} = \{\mathbf{x}|\mathbf{x_l} \leq \mathbf{x} \leq \mathbf{x_u}\} \\ & \mathbf{x_l} = (x_{l1}, \ldots, x_{ln}), \mathbf{x_u} = (x_{u1}, \ldots, x_{un}) \end{array} \tag{22}$$

In this case, the complete constrained EI of a solution $\mathbf{x}$ is simplified as

$$\begin{aligned} &\textbf{When no feasible solutions are available:} \\ &E[I_{c,N}(\mathbf{x})|g^{mN}] = \int_0^{g_{\min}^{+N}} \prod_{i=1}^{m}\Phi\left(\frac{z-\hat{g}_i(\mathbf{x})}{s_{g_i}(\mathbf{x})}\right)dz - g_{\min}^{+N} \times \prod_{i=1}^{m}\Phi\left(\frac{-\hat{g}_i(\mathbf{x})}{s_{g_i}(\mathbf{x})}\right) \\ &\textbf{When feasible solutions are available:} \\ &E[I_{c,N}(\mathbf{x})|f^N, g^{mN}] = \left[(f_{\min}^N - \hat{f}(\mathbf{x}))\Phi\left(\frac{f_{\min}^N-\hat{f}(\mathbf{x})}{s_f(\mathbf{x})}\right) + s_f(\mathbf{x})\phi\left(\frac{f_{\min}^N-\hat{f}(\mathbf{x})}{s_f(\mathbf{x})}\right)\right] \times \prod_{i=1}^{m}\Phi\left(\frac{-\hat{g}_i(\mathbf{x})}{s_{g_i}(\mathbf{x})}\right) \end{aligned} \tag{23}$$

## 5. A surrogate-assisted evolutionary algorithm framework

To test the effectiveness of the proposed constrained EI infill sampling criterion, we integrate the proposed method and the compared infill sampling criteria into a GP surrogate-assisted EA framework. The algorithm framework is listed in Algorithm 1, and it is based on a famous algorithm: Efficient Global Optimization (EGO) [22]. Note that a major concern of this study is to propose a constrained infill sampling criterion, instead of presenting a better evolutionary algorithm. We therefore simply use Algorithm 1 to show the performance of the proposed constrained EI and to compare it with other criteria. The proposed constrained EI can also be integrated into other surrogate-assisted EAs.

In initial experiment design of Algorithm 1, a popular space-infilling sampling method Latin hypercube design (LHD) [39] is used to get initial sampling data. The LHD method samples the design space uniformly and it has been widely used.

A key issue in an SAEA is how to use a reasonable amount of computational effort to build a good model for locating the most promising candidate solution. The computational cost for constructing the GP model becomes non-trivial when the number of tested points is large. The determination of hyper parameters via optimization becomes much more difficult as the size of the tested points grows, owing to both numerical difficulties in matrix inversion and the fact that this inversion

**Algorithm 1** The GP surrogate-assisted EA framework.

**Input:** $n$: dimension of the problem; $m$: number of constraints.
**Output:** $x_{best}$ and its fitness.

1: **Initialization:** Generate $11n-1$ points by LHD sampling method and expensive test them, find the best point $x_{best}$ and judge whether it is feasible.
2: **while** the halting criterion is not satisfied **do**
3:     **Clustering:** Cluster the tested points into several small clusters, see Algorithm 2.
4:     **Model Building:** Build local predictive GP models for objective and each constraint, respectively.
5:     **Evolutionary Optimization:** Apply DE algorithm to generate $NP$ candidate points, and estimate them by **using a constrained infill sampling criterion** (e.g., the proposed constrained EI).
6:     **Expensive Evaluation:** Ascertain the most promising point to evaluate using the original expensive objective and constraint functions. Update $x_{best}$.
7: **end while**

**Algorithm 2** Fuzzy clustering.

**Input:** $\mathbf{x}^{(1)},\ldots,\mathbf{x}^{(N)}$: the solutions to be clustered; $c_{size}$: the number of clusters; $\varepsilon$: stopping criterion; $\alpha$: a constant used in computing the objective $J$.
**Output:** $u_{ij}$ $(i=1,\ldots,N$ and $j=1,\ldots,c_{size})$: the membership of $\mathbf{x}^{(i)}$ in $j$th cluster; $v^1,\ldots v^{c_{size}}$: the cluster centers.

1: Initialize $u_{ij}^0$ $(i=1,\ldots,N$ and $j=1,\ldots,c_{size})$ and set $t$ =0.
2: Calculate the $j$th cluster center

$$v^j=\frac{\sum_{i=1}^N (u_{ij}^t)^\alpha x^i}{\sum_{i=1}^N (u_{ij}^t)^\alpha}.$$

3: Calculate the membership of $x_i$ in cluster $j$

$$u_{ij}^{t+1}=\frac{1}{\sum_{k=1}^{c_{size}}\left(\frac{||x^i-v^j||}{||x^i-v^k||}\right)^{\frac{2}{\alpha-1}}}.$$

4: **if** $\max\limits_{1\le i\le N, 1\le j\le c_{size}} |u_{ij}^{t+1}-u_{ij}^t|<\varepsilon$ **then**
5:     Stop and output $v^j$, $u_{ij}=u_{ij}^{t+1}$;
6: **else**
7:     $t=t+1$, go to step 1;
8: **end if**

must be repeated within the optimization loop. One way to overcome this drawback is selecting a small number of representative tested points for building a GP model (e.g., [28]). Nevertheless, the most obvious weakness of this method is it does not make full use of all the tested points. Partitioning all the tested points into several small clusters and then building several local predictive models based on these clusters can effectively alleviate GP surrogate modeling difficulty. Solving several small GP models is much faster than solving a GP model of all sample points. The clustering method of this paper adopted in Algorithm 2 is fuzzy clustering [49] because it can overcome the drawback that if the solution to be predicted is in boundary areas among different clusters. Note that other clustering techniques can also be candidates.

Fuzzy clustering needs two control parameters $L_1$ and $L_2$, $L_1>L_2$. Where $L_1$ is the maximal number of points which a local model contains and $L_2$ is the number of points for adding one more local model. Given $N$ tested points $\mathbf{x}^{(1)},\ldots,\mathbf{x}^{(N)}$, when $N\le L_1$, all the $N$ tested points are directly used for building one model. When $N>L_1$, fuzzy clustering follows two steps:

(1) Calculate the number of clusters

$$c_{\text{size}}=1+\lceil\frac{N-L_1}{L_2}\rceil.$$

(2) Cluster the $N$ tested points into $c_{size}$ clusters. To do this, the following objective function should be minimized

$$J=\sum_{i=1}^{N}\sum_{j=1}^{c_{\text{size}}}u_{ij}^\alpha||x^i-v^j||^2,$$

where $\alpha$ is a constant larger than 1, $v^j$ is the center of cluster $j$, $u_{ij}$ is the degree of membership of $x_i$ in cluster $j$, and ||.|| is Euclidean norm. Algorithm 2 shows details of fuzzy clustering. More details about fuzzy clustering can be found in [49].

**Table 1**
Summary of benchmark characteristics.

| Prob | $n$ | Type of $f$ | $f(\mathbf{x}^*)$ | $\rho$ | LI | NI | LE | NE | $a$ |
|---|---|---|---|---|---|---|---|---|---|
| G02mod | 2 | Nonlinear | – | – | 0 | 2 | 0 | 0 | 1 |
| G03mod | 2 | Polynomial | – | – | 0 | 0 | 0 | 1 | 1 |
| G04 | 5 | Quadratic | −30665.539 | 52.123% | 0 | 6 | 0 | 0 | 2 |
| G06 | 2 | Cubic | −6961.8139 | 0.0066% | 0 | 2 | 0 | 0 | 2 |
| G08 | 2 | Nonlinear | −0.095825 | 0.8560% | 0 | 2 | 0 | 0 | 0 |
| G09 | 7 | Polynomial | 680.630 | 0.5121% | 0 | 4 | 0 | 0 | 2 |
| G11 | 2 | Quadratic | 0.7499 | 0.0000% | 0 | 0 | 0 | 1 | 1 |
| G12 | 3 | Quadratic | −1.0 | 4.7713% | 0 | 1 | 0 | 0 | 0 |
| G24 | 2 | Linear | −5.508 | 79.6556% | 0 | 2 | 0 | 0 | 2 |

Building or updating a GP model for a cluster needs to maximize the likelihood function to determine parameters. Differential evolution (DE) [40] serves as search engine to maximize the likelihood function. Note, some state-of-the-art algorithms can also be regarded as the alternative optimizer (e.g., grasshopper optimization algorithm [37], social engineering optimizer [12], salp swarm algorithm [31]).

## 6. Experimental study

### 6.1. Benchmark problems

It is well-known that GP-assisted EAs are mainly suited for low-dimensional problems (particularly less than 10 decision variables) [6,42,43], due to the fact that the computational cost of constructing the GP model is $O(N^3)$, where $N$ is the number of sample points. As a result, in this section, nine representative benchmark test functions with dimension $n < 10$ and constraints number $m \leq 6$ collected in [26] were employed to validate the capability of the proposed constrained EI. We assume that the computation of these objective and constraint functions is expensive in this empirical study. Details of these test cases are reported in Table 1, where $n$ is the number of decision variables, $f(\mathbf{x}^*)$ represents the best known fitness, *LI* is the number of linear inequality constraints, *NI* is the number of nonlinear inequality constraints, *LE* is the number of linear equality constraints, and *NE* is the number of nonlinear equality constraints. $a$ is the number of active constraints. Note that *G*02 and *G*03 are scalable to their dimension $n$. In this paper, we set $n = 2$ for these two problems.

The feasibility ratio $\rho$ approximates the size of the feasible space in relation to the search space. The difficulty of a COP can be measured in terms of the estimated ratio $\rho$ between the feasible region and the entire search space. Intuitively, it is harder to find feasible solutions for problems with small $\rho$ values as compared to larger $\rho$ values.

### 6.2. Experimental settings

For fair comparison, all compared criteria adopt the same parameters.

The setup of Algorithm 1:

- The number of initial samples generated by LHD: $11n - 1$ [6,14,49], where $n$ is the number of variables of the problem.

The setup of Algorithm 2:

- The maximal number of points used for building a local model $L_1 = 80$ [49].
- The number of points for adding one more local model $L_2 = 20$ [49].
- $\alpha = 2, \varepsilon = 0.05$ [49].

The setup of DE:

- Population size $NP = 30$.
- Number of generations is 500.
- Crossover probability $CR = 0.9$ [40].
- Scaling factor $F = 0.5$ [40].

Other settings:

- Number of function evaluations $FEs = 50 \times n$.
- To increase the reliability of the comparison between the proposed approach and the alternative methods, the number of independent runs is set to 50.

To test the statistical significance, the Friedman's test is employed to sort the performance of all compared approaches and the Wilcoxon's rank sum test at significance level $\alpha = 0.05$ and $\alpha = 0.1$ is also implemented for all approaches in this paper.

### 6.3. Comparison with five infill sampling criteria

There are few such infill sampling criteria can solve expensive COPs if any feasible solution is not initially provided. Therefore, we only compare the proposed constrained EI with five infill sampling criteria which can deal with this issue:

1. EI based on penalty function (EI-PF):
Penalty function is the simplest and earliest approach in solving COPs. This method adds a penalty coefficient to the constraint violation value of each infeasible solution so that it will be penalized for violating constraints. The following penalty function is used in this paper:
$$F(\mathbf{x}) = f_{nor}(\mathbf{x}) + \eta * \mathbf{G}^{+}_{nor}(\mathbf{x}) \tag{24}$$
where $f_{nor}(\mathbf{x})$ and $\mathbf{G}^{+}_{nor}(\mathbf{x})$ are normalized fitness and constraint violation respectively, and $\eta$ is a penalty coefficient ($\eta = 10$ in our experiment). This method transforms a COP into an unconstrained one, then EI in Eq. (5) is selected as the infill sampling criterion.
2. EI based on feasibility probability (CEI-FP):
This method attempts to locate infill locations that maximize the probability of feasibility (until a feasible solution is identified) and then switches to maximize the constrained EI function in Eq. (7) [15].
3. MOP-EI:
In multi-objective based methods for COPs, the original objective and the degree of constraint violation are treated as two objectives to be optimized. During the optimization process, solutions with better objective value and less degree of constraint violation will be selected. For solving expensive COPs, a bi-objective technique borrowed from [10] is used in this paper:
$$\max \quad \mathbf{F}(\mathbf{x}) = (F_1(\mathbf{x}), F_2(\mathbf{x})) = (E[I(\mathbf{x})|f^N], \prod_{i=1}^{m} P\{l_i \le G_i(\mathbf{x}) \le u_i\}) \tag{25}$$
This approach treats objective improvement and constraint satisfaction as separate objectives for improving the choice of infill solutions. The non-dominated sorting procedure of NSGA-II [9] was used for the selection of the next generation from the current population and trial vector population for the DE. Finally a solution is chosen for expensive evaluation by maximizing the product of EI and feasibility probability.
4. Feasibility rule (FR):
In feasibility rule [8], one solution $x_i$ is regarded as superior to $x_j$ under the following conditions:
   - $x_i$ is feasible and $x_j$ is infeasible;
   - $x_i$ and $x_j$ are both feasible but the objective value of $x_i$ is better than $x_j$;
   - $x_i$ and $x_j$ are both infeasible but the degree of constraint violation of $x_i$ is smaller than $x_j$.

   Without the prediction uncertainty, in this method we only use the prediction fitness in Eq. (2) to compare and select points based on FR.
5. Lower Confidence Bound based Feasibility Rule (FR-LCB):
Liu et al. [29] adopt feasibility rule to solve expensive optimization problems with inequality constraints. They use LCB infill sample criterion to predict fitness:
$$y_{lcb}(\mathbf{x}) = \hat{y} - \omega\hat{s}(\mathbf{x}),\ \omega \in [0, 3] \tag{26}$$
where $\hat{y}$ and $\hat{s}(\mathbf{x})$ are predicted fitness and uncertainty in Eqs. (2) and (3), respectively. For objective function, $\omega$=2, and for constraint functions, $\omega$ is set to 0. Thus, it is obvious that this method only uses the predicted fitness value in Eq. (2) for each constraint function, and the uncertainty information of each constraint is neglected.

For fair comparison, these five mentioned infill sampling criteria and the proposed constrained EI (denoted by **New-CEI**) all adopt the framework in Algorithm 1 framework. All experiments run on a Python 2.7.5 Intel(R) Core(TM) i7-4770 CPU 3.40 GHz desktop machine. The best, average, worst and standard deviation of the objective function values obtained by different infill sampling criteria over 50 runs are listed in Table 2, where *fr* represents feasible rate that equals the ratio of running times where at least one feasible solution is found to the total running times. From Table 2, it can be seen that EI-PF and MOP-EI criteria on 3 (*G*06, *G*08, *G*09) and 5 (*G*03*mod*, *G*06, *G*08, *G*09, *G*11) problems cannot find a feasible point over all 50 runs, respectively, while FR, CEI-FP, FR-LCB and New-CEI can achieve feasible solutions on all test functions.

From Table 2, in terms of the mean results we can see that the proposed New-CEI performs better than FR, EI-PF, MOP-EI, CEI-FP and FR-LCB on 6, 9, 7, 5 and 4 test functions, respectively. By contrast, FR, EI-PF, MOP-EI, CEI-FP and FR-LCB have better performance than New-CEI on 2, 0, 1, 3 and 3 test functions. Therefore, we can conclude that, overall, the performance of New-CEI is superior or comparable to that of other five competitors.

Table 3 shows statistical test results based on the multiple-problem Wilcoxon's test. R+ $>$ R− means that the New-CEI is better than the compared approach and vice versa. As shown in Table 3, New-CEI achieves higher R+ values than R- values in four cases, which means that New-CEI performs better than FR, EI-PF, MOP-EI and CEI-FP. At $\alpha = 0.05$, significant differences can be observed in two situations (New-CEI vs EI-PF, New-CEI vs MOP-EI), which signifies that New-CEI has an edge over EI-PF and MOP-EI at $\alpha$=0.05. Besides, the Friedman's test in Fig. 1 shows that New-CEI and FR-LCB have the best ranking among all strategies.

**Table 2**
Function values achieved by different methods over 50 independent runs.

| Problem | | FR | EI-PF | MOP-EI | CEI-FP | FR-LCB | New-CEI |
|---|---|---|---|---|---|---|---|
| **G02mod** | Best | −0.295837 | −0.261338 | −0.364992 | −0.364840 | −0.364769 | −0.364858 |
| | Mean | −0.218564 | −0.199826 | −0.300833 | −0.297117 | **−0.307987** | −0.297884 |
| | Worst | −0.174744 | −0.122105 | −0.255903 | −0.199241 | −0.150148 | −0.237787 |
| | Std | 4.73E−02 | 4.32E−01 | 7.09E−01 | 4.79E−02 | 6.33E−02 | 4.39E−02 |
| | fr | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| **G03mod** | Best | −1.000081 | −0.619851 | −1.004265 | −1.000083 | −1.000077 | −1.000095 |
| | Mean | −0.999829 | −0.103773 | −0.883851* | −0.999886 | −0.999905 | **−0.999935** |
| | Worst | −0.999828 | −0.000003 | −0.941279 | −0.999260 | −0.998470 | −0.999481 |
| | Std | 6.53E−03 | 1.20E−01 | 7.33E−03 | 1.92E−04 | 3.06E−04 | 1.49E−04 |
| | fr | 1.00 | 1.00 | 0.00 | 1.00 | 1.00 | 1.00 |
| **G04** | Best | −30341.172 | −30203.119 | −30611.003 | −30687.009 | −30612.004 | −30606.316 |
| | Mean | −30112.999 | −29113.446 | −30449.209 | −30462.297 | −30549.586 | **−30569.126** |
| | Worst | −29861.131 | −29250.190 | −30558.594 | −30408.983 | −30386.523 | −30369.125 |
| | Std | 2.13E+02 | 2.77E+02 | 1.51E+01 | 2.01E+01 | 9.48E+01 | 9.47E+01 |
| | fr | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| **G06** | Best | −6922.722 | −7643.743 | −7657.340 | −6904.856 | −6865.667 | −6841.711 |
| | Mean | **−6804.485** | −4412.093* | −6823.001* | −6654.270 | −6739.981 | −6628.076 |
| | Worst | −6591.795 | −543.495 | −5416.568 | −6374.732 | −6556.421 | −6086.643 |
| | Std | 7.45E+01 | 2.11E+03 | 4.29E+02 | 1.37E+02 | 7.84E+01 | 1.65E+02 |
| | fr | 1.00 | 0.04 | 0.92 | 1.00 | 1.00 | 1.00 |
| **G08** | Best | −0.095825 | −0.056681 | −0.093379 | −0.095816 | −0.095822 | −0.095821 |
| | Mean | −0.064264 | −0.021774* | −0.044710* | −0.084673 | **−0.090226** | −0.087268 |
| | Worst | −0.001002 | 0.0286671 | 0.003673 | −0.029111 | −0.029138 | −0.028949 |
| | Std | 3.21E−02 | 2.06E−02 | 2.98E−02 | 1.70E−02 | 1.36E−02 | 1.39E−02 |
| | fr | 1.00 | 0.94 | 0.94 | 1.00 | 1.00 | 1.00 |
| **G09** | Best | 832.116 | 842.134 | 881.752 | 702.556 | 756.223 | 708.213 |
| | Mean | 1056.887 | 18889.661* | 19326.006* | 821.005 | 878.799 | **812.669** |
| | Worst | 1572.863 | 5080.321 | 1933.691 | 987.334 | 981.951 | 894.923 |
| | Std | 4.23E+02 | 1.08E+03 | 5.96E+02 | 2.99E+02 | 1.03E+02 | 4.98E+01 |
| | fr | 1.00 | 0.94 | 0.14 | 1.00 | 1.00 | 1.00 |
| **G11** | Best | 0.7500 | 0.7851 | 0.7356 | 0.7499 | 0.7450 | 0.7499 |
| | Mean | 0.7540 | 0.9143 | 0.8016* | **0.7509** | 0.7526 | 0.7516 |
| | Worst | 0.7822 | 1.0000 | 0.8802 | 0.7579 | 0.7692 | 0.7644 |
| | Std | 6.84E−03 | 5.15E-02 | 1.98E-02 | 1.60E-03 | 4.14E-03 | 2.95E-03 |
| | fr | 1.00 | 1.00 | 0.00 | 1.00 | 1.00 | 1.00 |
| **G12** | Best | -1.000 | -1.000 | -1.000 | -1.000 | -1.000 | -1.000 |
| | Mean | **-1.000** | -0.997 | **-1.000** | **-1.000** | **-1.000** | **-1.000** |
| | Worst | -1.000 | -0.971 | -1.000 | -1.000 | -1.000 | -1.000 |
| | Std | 1.08E-05 | 8.57E-03 | 1.53E-05 | 1.54E-06 | 8.37E-05 | 7.10E-05 |
| | fr | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| **G24** | Best | -5.5075 | -5.3986 | -5.5071 | -5.5066 | -5.5077 | -5.5064 |
| | Mean | -5.5069 | -5.0189 | -5.5021 | **-5.5038** | -5.5035 | -5.5035 |
| | Worst | -5.5058 | -4.6422 | -5.4959 | -5.4977 | -5.5047 | -5.4981 |
| | Std | 5.12E-04 | 1.87E-01 | 3.46E-03 | 2.30E-03 | 7.91E-04 | 1.91E-03 |
| | fr | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| + | | **6** | **9** | **7** | **5** | **4** | \ |
| - | | **2** | **0** | **1** | **3** | **3** | \ |
| ≈ | | **1** | **0** | **1** | **1** | **2** | \ |

"+", "-", and "≈" represent that the performance of the New-CEI is better than, worse than, and similar to that of the corresponding method, respectively. * represents infeasible solution, *fr* represents feasible rate.

**Table 3**
Results of the multiple-problem Wilcoxon's test for 3 infill sampling criteria.

| Algorithm | R+ | R− | $\alpha = 0.05$ | $\alpha = 0.1$ |
|---|---|---|---|---|
| New-CEI vs FR | 27.0 | 9.0 | ≈ | ≈ |
| New-CEI vs EI-PF | 45.0 | 0.0 | + | + |
| New-CEI vs MOP-EI | 34.0 | 2.0 | + | + |
| New-CEI vs CEI-FP | 24.0 | 12.0 | ≈ | ≈ |
| New-CEI vs FR-LCB | 14.0 | 14.0 | ≈ | ≈ |

R+, R− represent sum ranks of which New-CEI is better and worse than the compared approach with regard to the mean values, respectively.

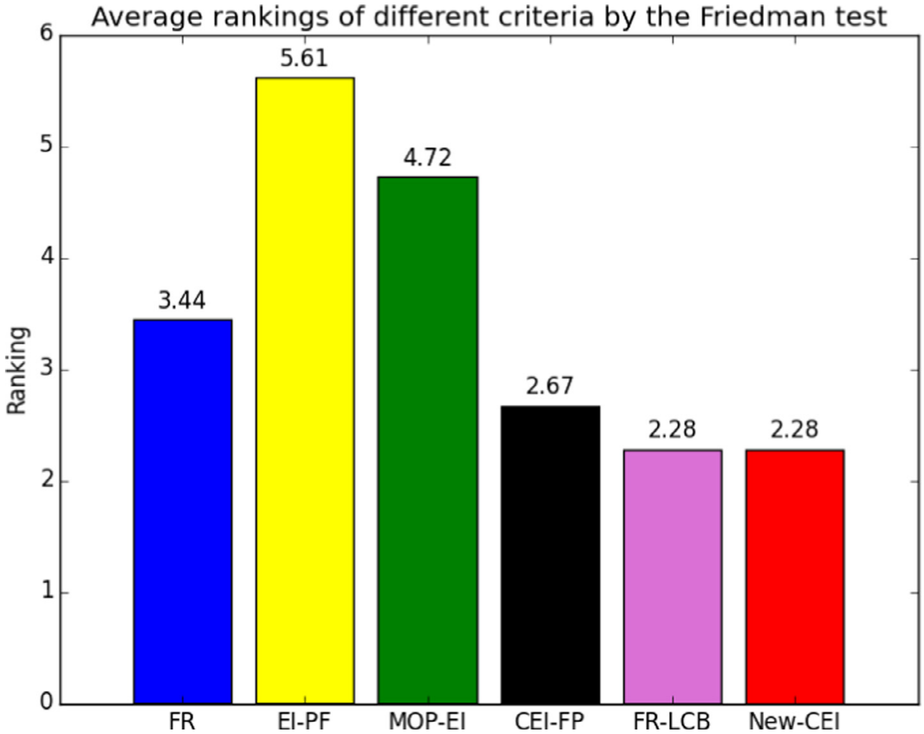


**Fig. 1.** Average rankings of different criteria by the Friedman test for the CEC 2006 functions. The lower the ranking, the better the performance.

**Results analysis**: Based on above experimental results, we make the following remarks: EI-PF transforms a COP into an unconstrained one, then the original unstrained EI infill sampling criterion in Eq. (5) is used for selecting potential individuals for expensive evaluation. However, one problem is that the transformed fitness in Eq. (24) does not satisfy Gaussian distribution any more, so this can result in performance degeneration. MOP-EI optimizes two objectives (EI in Eq. (5) and feasibility probability) simultaneously, and it is the same as CEI in Eq. (7) to some extent. If no feasible point can be obtained in the sample, MOP-EI still does not work, this can be explained why MOP-EI obtains infeasible solutions for problems $G03mod$, $G06$, $G08$, $G09$ and $G11$. FR and FR-LCB both adopt feasibility rule to select good points. Compared with FR, FR-LCB takes advantage of uncertainty information of objective function, and therefore has better performance than FR. Nevertheless, FR-LCB and FR all neglect the uncertainty information of the constraint functions. Since GP model can provide the mean value together with the variance, neglecting the variance of each constraint function will lead to the waste of the uncertainty information. A point with high uncertainty should be far away from the points which have been evaluated in the previous search. As mentioned in [14], the uncertainty information of predicted fitness values of objective and constraints plays an important role in SAEAs, since sampling the most uncertain points can not only efficiently enhance the surrogate model quality, but also explore the unexplored region of the search space. Therefore, it is better to utilize the uncertainty information for objective and constraints when designing a constrained infill criterion.

In conclusion, the above comparisons and discussions show that the proposed New-CEI criterion is better than or comparable to five referred infill sampling criteria on nine benchmark test functions in terms of the feasible rate and solution quality.

### *6.4. Effectiveness of searching a feasible solution*

To demonstrate the effectiveness of the proposed constrained EI criterion for searching feasible solutions, we perform an additional experiment to compare the average number of real expensive evaluations used by different infill sampling criteria in finding a feasible point over 50 independent runs. Six benchmark problems (G03mod, G06, G08, G09, G11 and G12) with thin feasible regions are tested. For these six problems, initial feasible solutions cannot be obtained directly through the initial Latin hypercube design sampling procedure. Therefore, the most important task for solving these six test functions is to find a feasible point as quickly as possible.

Fig. 2 shows results of these six test functions obtained by different infill sampling criteria, in which "NA" denotes the failure of finding a feasible solution when the termination condition is met (given a relatively limited number of simulations). We can see that both EI-PF and MOP-EI fail to achieve feasible solutions under a limited budget on 3 and 5 problems. The reason seems straightforward: after combining the objective and constraint violation as a penalty function, EI-PF does

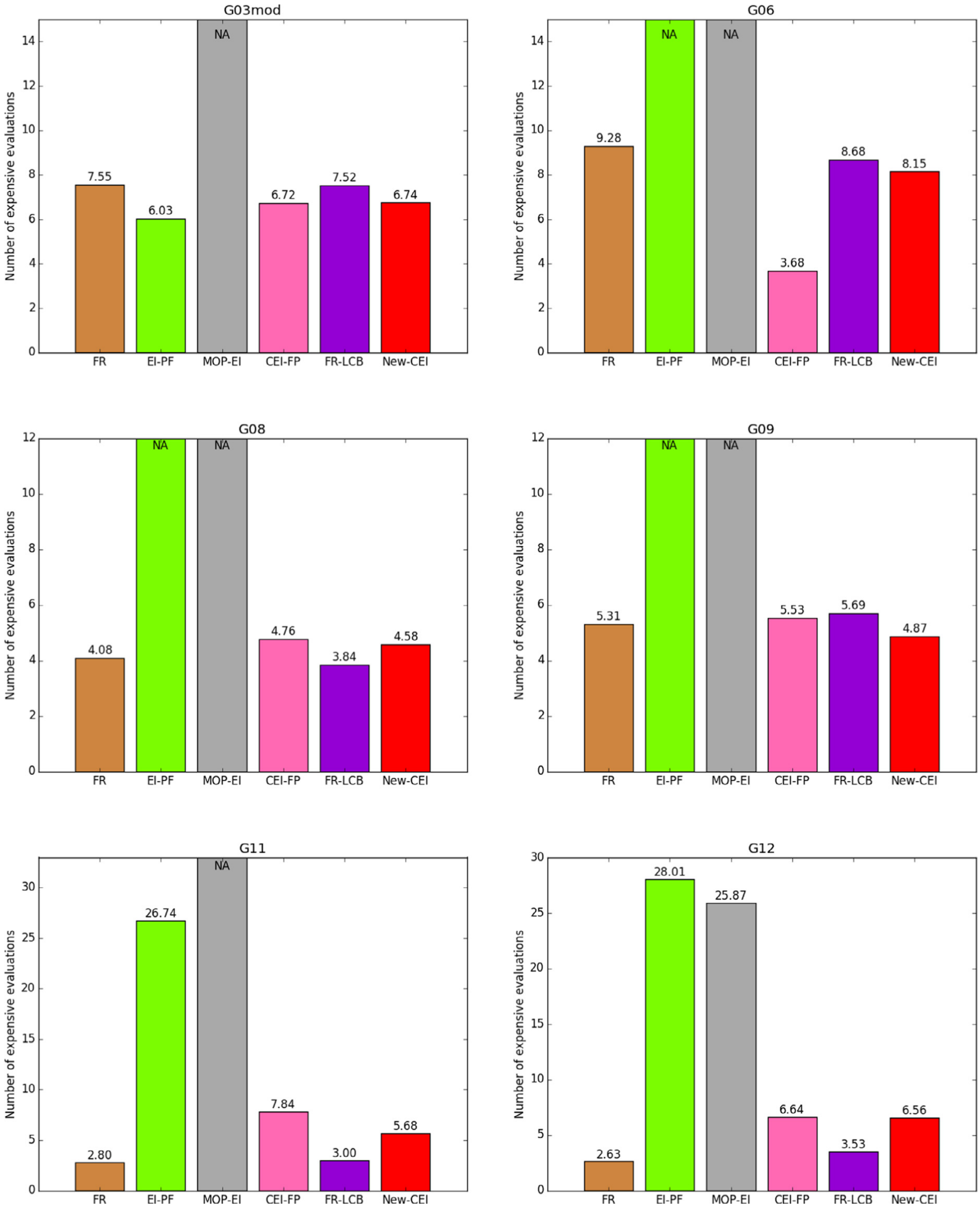


**Fig. 2.** Average number of expensive evaluations for finding a feasible solution.

not satisfy Gaussian distribution any more; MOP-EI only work under the condition that a feasible point is provided. As a result, these two techniques cannot perform well on these problems. From the Fig. 3, we can observe that New-CEI and FR have the best ranking according to the Friedman's test. This demonstrates that the proposed New-CEI criterion is able to search a feasible point within a few number of expensive evaluations, which verifies the effectiveness of the New-CEI criterion in approaching the feasible area from large infeasible regions.

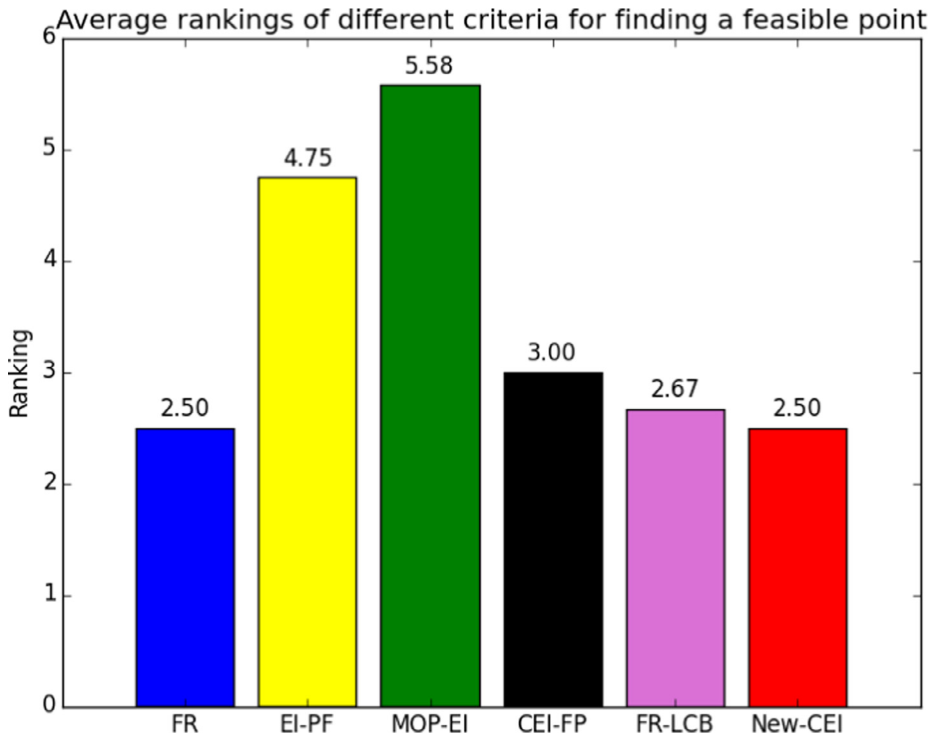


**Fig. 3.** Average rankings of different criteria by the Friedman test for finding a feasible point. The lower the ranking, the better the performance.

### *6.5. Effectiveness of the proposed general form of COP*

In this paper, the proposed and compared infill sampling criteria all adopt a general form of COP in Eq. (1). In this subsection, we take a test problem with an equality constraint as an example to discuss why we recommend the general form of COP rather than the special form in Eq. (22) which is widely used in the evolutionary computation community.

Supposing an equality constraint occurs (i.e., $h(\mathbf{x}) = 0$), to match the special form of COP in Eq. (22), it is usually changed into two inequality constraints with the form

$$\begin{aligned} &h_1(\mathbf{x}) - \epsilon \le 0, \\ &-h_2(\mathbf{x}) - \epsilon \le 0 \end{aligned} \tag{27}$$

where $\epsilon$ is the tolerance and is usually set to 0.0001 [26]. Note these two inequality constraints are dependent, and the correlation coefficient equals $-1$. It means the assumption of independency does not hold.

On the other hand, to match the general form of COP in Eq. (1), the equality constraint can be transformed into an inequality constraint with the form

$$-\epsilon \le h(\mathbf{x}) \le \epsilon. \tag{28}$$

Notably, in the general case, no additional dependency occurs.

Here, we take the G03mod problem as an example:

$$\begin{aligned} \min \quad & f(\mathbf{x}) = -(\sqrt{2})^2 \prod_{i=1}^{2} x_i \\ \text{st}: \quad & h(\mathbf{x}) = \sum_{i=1}^{2} x_i^2 - 1 = 0 \\ & \mathbf{0} \le x_1, x_2 \le \mathbf{1} \end{aligned}$$

To match the special case of COP in Eq. (22), G03mod is changed into

$$\begin{aligned} \min \quad & f(\mathbf{x}) = -(\sqrt{2})^2 \prod_{i=1}^{2} x_i \\ \text{st}: \quad & g_1(\mathbf{x}) = \sum_{i=1}^{2} x_i^2 - 1 - 0.0001 \le 0 \\ & g_2(\mathbf{x}) = -(\sum_{i=1}^{2} x_i^2 - 1) - 0.0001 \le 0 \end{aligned} \tag{29}$$

**Table 4**
Results obtained and average run times (in seconds) spent by proposed constrained EI for problems (29) and (30).

| | Special case | General case |
|---|---|---|
| Best | −1.000076 | −1.000081 |
| Mean | −0.999937 | −0.999859 |
| Worst | −0.999482 | −0.997678 |
| Std | 1.84E−04 | 4.60E−04 |
| Running time | 0:57:44 | 1:35:56 |

To match the general case of COP in Eq. (1), it is changed into

$$\begin{aligned} \min \quad & f(\mathbf{x}) = -(\sqrt{2})^2 \prod_{i=1}^{2} x_i \\ \text{st}: \quad & -0.0001 \le g(\mathbf{x}) = \sum_{i=1}^{2} x_i^2 - 1 \le 0.0001 \end{aligned} \tag{30}$$

It is obvious that Eq. (29) has one more constraint than Eq. (30), which means that the special form of COP will consume more time and resource to build a GP model for the second constraint.

Therefore, compared with the general form of COP in Eq. (1), the special form of COP in Eq. (22) will double the number of constraints and introduce additional dependencies, which leads to two difficulties in expensive COP:

1. Handling these double constraints and additional dependencies increase the expense of computational resources since the GP model for each constraint needs to be built, and double constraints increase the dimension of the covariance matrix while additional dependencies determine the dimension could not be reduced.
2. Most technologies for expensive COPs may fail to work since these technologies are based on the assumption of mutual independency among the objective and constraints.

Table 4 shows the average running time of two forms of COP in solving G03mod problem by an Intel(R) Core(TM) i7-4770 CPU 3.40 GHz desktop machine. From Table 4, we can see that the general form of COP has a much larger average computational overhead compared with the special form of COP. Besides, the special form of COP has better performance than the general form of COP according to mean results, though the advantage is not obvious. This is understandable due to the fact that the computational complexity for the special form of COP includes the time spent building the GP model for additional constraint function, and the assumption of independencies are violated.

Therefore, we can conclude that the proposed general form of COP is effective in terms of the average computational overhead and solution quality compared with the classical form of COP.

## 7. Conclusion and future research

Most COPs in the engineering design with costly objective and constraints. For an expensive highly COP, obtaining a feasible point is very hard, thus the existing constrained EI infill sampling criteria cannot work well since they assume that at least one feasible point is obtainable in candidate solutions. In other words, the constrained EI is incomplete. To remedy this issue, this paper proposed a complete constrained EI infill sampling criterion to deal with expensive highly COPs. When a feasible solution is not available in the sampling data, the EI of constraint violation is adopted for this infeasible case. The degree of constraint violation of a solution represents the distance to the feasible region, it is usually adopt to tackle the constraint difficulty. Maximizing the EI of constraint violation guides the search towards the feasible areas. The proposed EI of constraint violation is a complement to the existing constrained EI. We have theoretically verified the validity of the complete constrained EI infill sampling criterion.

Nine benchmark test problems collected from the IEEE CEC2006 were used to investigate the efficiency of the proposed complete constrained EI criterion. Experimental results show that the proposed constrained EI criterion can successfully find feasible solutions under a limited budget for all problems, which validates the effectiveness of adopting EI of constraint violation to approach feasible regions. Besides, compared with five infill sampling criteria based on the GP model, the proposed constrained EI criterion is better than or comparable to existing constrained infill sampling criteria.

The following are a few possible future research topics along the line of this work:

1. In this study, we assume that the objective and constraints are mutually independent. As mentioned in Section 6.5, the assumption of independency between the objective and each constraint may lead to the performance degradation. In the future, we will study the case that objective and each constraint are statistically dependent.
2. Solving equality constraints is not an easy work. Only a very limited number of evaluations are allowed for computationally expensive optimization. But with equality constraints, more evaluations are generally required to get a feasible point. As a result, the proposed constrained EI can be combined with some state-of-art constraint-handling techniques to solve expensive COPs with equality constraints.

3. The lack of the expensive benchmark problems with constraints hinders the development of constrained SAEAs to some extent. So in the future, we will develop a benchmark expensive COP test suite based on the Free Peaks framework [25] and use it to evaluate the proposed constrained EI criterion.

## Acknowledgment

This work was supported in part by the National Natural Science Foundation of China under Grant 61673355, in part by the Hubei Provincial Natural Science Foundation of China under Grant 2015CFA010, in part by the 111 project under Grant B17040.

## References

[1] C. Audet, D. J., D. Moore, A. Booker, P. Frank, A surrogate-model-based method for constrained optimization, in: 8th Symposium on Multidisciplinary Analysis and Optimization, 2000, pp. 1–8. Long Beach, CA, USA.
[2] N. Awad, M. Ali, R. Mallipeddi, P. Suganthan, An improved differential evolution algorithm using efficient adapted surrogate model for numerical optimization, Inf. Sci. 451 (2018) 326–347.
[3] S. Bagheri, W. Konen, R. Allmendinger, J. Branke, K. Deb, J. Fieldsend, D. Quagliarella, K. Sindhya, Constraint handling in efficient global optimization, in: Proceedings of the Genetic and Evolutionary Computation Conference, ACM, Berlin, Germany, 2017, pp. 673–680.
[4] A. Basudhar, C. Dribusch, S. Lacaze, S. Missoum, Constrained efficient global optimization with support vector machines, Struct. Multidiscip. Optim. 46 (2) (2012) 201–221.
[5] K. Bhattacharjee, H. Singh, T. Ray, J. Branke, Multiple surrogate assisted multiobjective optimization using improved pre-selection, in: 2016 IEEE Congress on Evolutionary Computation (CEC), 2016, pp. 4328–4335. Vancouver, BC, Canada.
[6] T. Chugh, Y. Jin, K. Miettinen, J. Hakanen, K. Sindhya, A surrogate-assisted reference vector guided evolutionary algorithm for computationally expensive many-objective optimization, IEEE Trans. Evol. Comput. 22 (1) (2018) 129–142.
[7] R. Datta, R. Regis, A surrogate-assisted evolution strategy for constrained multi-objective optimization, Expert Sys. Appl. 57 (1) (2016) 270–284.
[8] K. Deb, An efficient constraint handling method for genetic algorithms, Comput. Methods Appl. Mech. Eng. 186 (2–4) (2000) 311–338.
[9] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Trans. Evol. Comput. 6 (2) (2002) 182–197.
[10] C. Durantin, J. Marzat, M. Balesdent, Analysis of multi-objective Kriging-based methods for constrained global optimization, Comput. Optim. Appl. 63 (3) (2016) 903–926.
[11] M. Emmerich, K. Giannakoglou, B. Naujoks, Single- and multiobjective evolutionary optimization assisted by Gaussian random field metamodels, IEEE Trans. Evol. Comput. 10 (4) (2006) 421–439.
[12] A. Fathollahi-Fard, M. Hajiaghaei-Keshteli, R. Tavakkoli-Moghaddam, The social engineering optimizer (SEO), Eng. Appl. Artif. Intel. 72 (2018) 267–293.
[13] C. Goh, D. Lim, L. Ma, Y. Ong, P. Dutta, A surrogate-assisted memetic co-evolutionary algorithm for expensive constrained optimization problems, in: 2011 IEEE Congress on Evolutionary Computation (CEC), 2011, pp. 744–749. New Orleans, LA, USA.
[14] D. Guo, Y. Jin, J. Ding, T. Chai, Heterogeneous ensemble based infill criterion for evolutionary multi-objective optimization of expensive problems, IEEE Trans. Cybern. (2018) 1–13, doi:10.1109/TCYB.2018.2794503.
[15] A. Habib, H. Singh, T. Ray, A study on the effectiveness of constraint handling schemes within efficient global optimization framework, in: 2016 IEEE Symposium Series on Computational Intelligence (SSCI), 2016, pp. 1–8. Athens, Greece.
[16] S. Handoko, C. Kwoh, Y. Ong, Feasibility structure modeling: an effective chaperone for constrained memetic algorithms, IEEE Trans. Evol. Comput. 14 (5) (2010) 740–758.
[17] S. Horng, S. Lin, Evolutionary algorithm assisted by surrogate model in the framework of ordinal optimization and optimal computing budget allocation, Inf. Sci. 233 (1) (2013) 214–229.
[18] R. Jiao, Y. Sun, J. Sun, Y. Jiang, S. Zeng, Antenna design using dynamic multi-objective evolutionary algorithm, IET Microwave Antenna Propag. (2018), doi:10.1049/iet-map.2018.5298.
[19] R. Jiao, S. Zeng, C. Li, J. Y., J. Wang, Expected improvement of constraint violation for expensive constrained optimization, in: Genetic and Evolutionary Computation Conference (GECCO), 2018, pp. 1039–1046. Kyoto, Japan.
[20] Y. Jin, Surrogate-assisted evolutionary computation: recent advances and future challenges, Swarm Evol. Comput. 1 (2) (2011) 61–70.
[21] Y. Jin, B. Sendhoff, A systems approach to evolutionary multiobjective structural optimization and beyond, IEEE Comput. Intell. Mag. 4 (3) (2009) 62–76.
[22] D. Jones, M. Schonlau, W. Welch, Efficient global optimization of expensive black-box functions, J. Global Optim. 13 (4) (1998) 455–492.
[23] A. Kattan, Y. Ong, Surrogate genetic programming: a semantic aware evolutionary search, Inf. Sci. 296 (2015) 345–359.
[24] M. Krithikaa, R. Mallipeddi, Differential evolution with an ensemble of low-quality surrogates for expensive optimization problems, in: 2016 IEEE Congress on Evolutionary Computation (CEC), 2016, pp. 78–85. Vancouver, BC, Canada.
[25] C. Li, T.T. Nguyen, S. Zeng, M. Yang, M. Wu, An open framework for constructing continuous optimization problems, IEEE Trans. Cybern. (99) (2018) 1–15, doi:10.1109/TCYB.2018.2825343.
[26] J. Liang, T. Runarsson, E. Mezura-Montes, M. Clerc, P. Suganthan, C. Coello, K. Deb, Problem definitions and evaluation criteria for the CEC 2006 special session on constrained real-parameter optimization, J. Appl. Mech. 41 (8) (2006).
[27] D. Lim, Y. Jin, Y. Ong, B. Sendhoff, Generalizing surrogate-assisted evolutionary computation, IEEE Trans. Evol. Comput. 14 (3) (2010) 329–355.
[28] B. Liu, Q. Zhang, G. Gielen, A Gaussian process surrogate model assisted evolutionary algorithm for medium scale expensive optimization problems, IEEE Trans. Evol. Comput. 18 (2) (2014) 180–192.
[29] B. Liu, D. Zhao, P. Reynaert, G. Gielen, Gaspad: a general and efficient mm-wave integrated circuit synthesis method based on surrogate model assisted evolutionary algorithm, IEEE Trans. Comput. Aided Des. Integr. Circuits Syst. 33 (2) (2014) 169–182.
[30] E. Mezura-Montes, C. Coello, Constraint-handling in nature-inspired numerical optimization: past, present and future, Swarm Evol. Comput. 1 (4) (2011) 173–194.
[31] S. Mirjalili, A. Gandomi, S. Mirjalili, S. Saremi, H. Faris, S. Mirjalili, Salp swarm algorithm: a bio-inspired optimizer for engineering design problems, Adv. Eng. Softw. 114 (2017) 163–191.
[32] N. Namura, K. Shimoyama, S. Obayashi, Expected improvement of penalty-based boundary intersection for expensive multiobjective optimization, IEEE Trans. Evol. Comput. 21 (6) (2017) 898–913.
[33] L. Pan, C. He, Y. Tian, H. Wang, X. Zhang, Y. Jin, A classification based surrogate-assisted evolutionary algorithm for expensive many-objective optimization, IEEE Trans. Evol. Comput. (2018) 1–15, doi:10.1109/TEVC.2018.2802784.
[34] R. Preen, L. Bull, Toward the coevolution of novel vertical-axis wind turbines, IEEE Trans. Evol. Comput. 19 (2) (2015) 284–294.
[35] R. Regis, Evolutionary programming for high-dimensional constrained expensive black-box optimization using radial basis functions, IEEE Trans. Evol. Comput. 18 (3) (2014) 326–347.
[36] T. Santner, B. Williams, W. Notz, The design and analysis of computer experiments, Springer Verlag, 2003.
[37] S. Saremi, S. Mirjalili, A. Lewis, Grasshopper optimisation algorithm: theory and application, Adv. Eng. Softw. 105 (2017) 30–47.
[38] M. Schonlau, W. Welch, D. Jones, Global versus local search in constrained optimization of computer models, in: Lecture Notes-Monograph Series, 1998, pp. 11–25.

[39] M. Stein, Large sample properties of simulations using latin hypercube sampling, Technometrics 29 (2) (1987) 143–151.
[40] R. Storn, K. Price, Differential evolution a simple and efficient heuristic for global optimization over continuous spaces, J. of Global Optim. 11 (4) (1997) 341–359.
[41] C. Sun, Y. Jin, R. Cheng, J. Ding, J. Zeng, Surrogate-assisted cooperative swarm optimization of high-dimensional expensive problems, IEEE Trans. Evol. Comput. 21 (4) (2017) 644–660.
[42] C. Tinkle, S. Karthik, H. Jussi, M. Kaisa, A survey on handling computationally expensive multiobjective optimization problems with evolutionary algorithms, Soft Comput. (2017) 1–30, doi:10.1007/s00500-017-2965-0.
[43] H. Wang, Y. Jin, J. Doherty, Committee-based active learning for surrogate-assisted particle swarm optimization of expensive problems, IEEE Trans. Cybern. 47 (9) (2017) 2664–2677.
[44] H. Wang, Y. Jin, J. Jansen, Data-driven surrogate-assisted multiobjective evolutionary optimization of a trauma system, IEEE Trans. Evol. Comput. 20 (6) (2016) 939–952.
[45] H. Yu, Y. Tan, J. Zeng, C. Sun, Y. Jin, Surrogate-assisted hierarchical particle swarm optimization, Inf. Sci. 454 (2018) 59–72.
[46] S. Zeng, R. Jiao, C. Li, B. Lan, H. Li, J. Sun, Y. Sun, Typical constrained optimization formulation in evolutionary computation not suitable for expensive optimization, in: Computational Intelligence and Intelligent Systems - 9th International Symposium(ISICA 2017), 2018, pp. 232–245. Guangzhou, China.
[47] S. Zeng, R. Jiao, C. Li, X. Li, J.S. Alkasassbeh, A general framework of dynamic constrained multiobjective evolutionary algorithms for constrained optimization, IEEE Trans. Cybern. 47 (9) (2017) 2678–2688.
[48] D. Zhan, Y. Cheng, J. Liu, Expected improvement matrix-based infill criteria for expensive multiobjective optimization, IEEE Trans. Evol. Comput. 21 (6) (2017) 956–975.
[49] Q. Zhang, W. Liu, E. Tsang, B. Virginas, Expensive multiobjective optimization by MOEA/D with gaussian process model, IEEE Trans. Evol. Comput. 14 (3) (2010) 456–474.