

# Enhancing infill sampling criteria for surrogate-based constrained optimization

James M. Parr<sup>a,\*</sup>, Alexander I. J. Forrester<sup>a</sup>, Andy J. Keane<sup>a</sup> and Carren M. E. Holden<sup>b</sup>

<sup>a</sup>*University of Southampton, Southampton, England, UK*

<sup>b</sup>*Airbus Operations Ltd, Bristol, England, UK*

**Abstract.** A popular approach to handling constraints in surrogate-based optimization is through the addition of penalty functions to an infill sampling criterion that seeks objective improvement. Typical sampling metrics, such as expected improvement tend to have multimodal landscapes and can be difficult to search. When the problem is transformed using a penalty approach the search can become riddled with cliffs and further increases the complexity of the landscape. Here we avoid searching this aggregated space by treating objective improvement and constraint satisfaction as separate goals, using multiobjective optimization. This approach is used to enhance the efficiency and reliability of infill sampling and shows some promising results. Further to this, by selecting model update points in close proximity to the constraint boundaries, the regions that are likely to contain the feasible optimum can be better modelled. The resulting enhanced probability of feasibility is used to encourage the exploitation of constraint boundaries.

Keywords: Surrogate model, infill sampling, constrained optimization

## 1. Introduction

The evaluation cost of objective function(s) and any constraints is an important factor in any optimization process. When function evaluations are considered to be ‘expensive’, surrogate (meta or response surface) models can be used to replace the original functions and thus allow further inexpensive function evaluations. Local surrogates may be used to cheaply compute gradients or solutions based on heuristics, accelerating both gradient based and pattern search methods [1,6]. In evolutionary algorithms, either local or global surrogates can be used to approximate objective and constraint fitness [20]. Here we consider methods analogous to the efficient global optimization (EGO) algorithm, popularized by Jones et al. [22].

There is a wide choice of surrogate models available, all with their respective advantages and disadvantages. A taxonomy of different models is provided by Jones et al. [21] and more recently by Forrester and Keane [14]. Kriging is a popular method introduced to the field of engineering by Sacks et al. [39]. By modelling the underlying function as a realization of a stochastic process, Kriging facilitates the estimation of model uncertainty, a characteristic useful in choosing model update points. Only Kriging models have been considered in this paper, however the different infill criteria described are transferable to other strategies that provide model uncertainty estimation, such as Gaussian Radial Basis Functions as used by Sóbester et al. [44] and Regis and Shoemaker [38].

---

\*Corresponding author: James M. Parr, University of Southampton, Southampton, England SO17 1BJ, UK. Tel.: +44 7843123402; E-mail: jim.parr@soton.ac.uk.

Building a surrogate model commonly involves two stages. The first stage occurs prior to having any knowledge of the design space and makes use of an initial sample based on some Design of Experiments (DoE) technique [25]. A surrogate model is then built based on true simulations from this initial sample. The second stage extracts knowledge from the surrogate to find areas for model refinement referred to as updating. These update points are selected via an infill sampling criterion such as expected improvement [22]. This second stage is repeated until a time limit, evaluation budget, convergence, or a model accuracy is reached.

It is very common and perhaps almost always the case that real world design optimization problems are subject to a number of constraints. Although there has been extensive research into surrogate modelling and efficient infill sampling criteria over the last two decades, there has been relatively little emphasis on dealing with constraints considering how often they arise in practise.

Here constrained problems are considered where both the objective and constraint functions are assumed expensive to evaluate, and therefore replaced with surrogate models. A popular approach to handling constraints is through the addition of a penalty function to an infill sampling criterion that seeks objective improvement. The penalty is applied if any constraint surrogate predicts a violation. This approach relies on accurately modelled constraint functions for the penalty to be correctly applied. In cases where the constraint functions are well modelled and are less complex than the objective functions, using a simple penalty function can perform well [30], however as the number of constraints increase or become more complex, using a penalty function naively exploits the surrogates and no attempt is made to directly improve the constraint approximations.

Parr et al. [30] provide a review of infill sampling criteria for constrained problems, showing a probabilistic approach to constraint handling outperforms the penalty approach on a number of problems. This probabilistic approach, originally suggested by Schonlau [42] uses the product of expected improvement of the objective and the probability of feasibility of the constraints. The method gradually drives the infill criterion towards zero in the transition from feasible to infeasible regions, smoothing the sheer cliff landscape produced by a simple penalty function.

In both the penalty and probabilistic approach, the constrained problem is transformed into an unconstrained one. This manipulation of the constrained problem may result in misrepresentation, a concern considered by Audet et al. [4] and more recently by Regis [37]. Furthermore, using metrics such as expected improvement tends to lead to a multimodal landscape which can in itself be difficult to search. When the problem is transformed using either the probabilistic or penalty approach the search can become riddled with cliffs and thus further increases the difficulty of the search. This becomes even more of an issue when dealing with multiple and complex constraints. Parr et al. [31] avoid searching the aggregated space by treating objective improvement and constraint satisfaction as separate goals using the formation of Pareto sets. This gives a flexible choice of update points and encourages the selection of multiple points which can be evaluated in parallel. Here we compare the efficiency and reliability of single and multiobjective approaches, but first we begin with a brief overview of Kriging.

## 2. Overview of Kriging

This overview of Kriging is based on the introduction provided by Jones [21]. Other publications useful to the interested reader include Forrester et al. [15,16] and Sacks et al. [39].

To begin let the function prediction,  $\hat{f}(\mathbf{x})$ , be used as a surrogate to the expensive function  $f(\mathbf{x})$ . This surrogate is built using a set of inputs  $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(n)}$  and known outputs  $\mathbf{y} = \{y^{(1)}, y^{(2)}, \dots, y^{(n)}\}^T$ . Before demonstrating how this prediction is

achieved using Kriging it is necessary to view our known outputs,  $\mathbf{y}$ , as being the results from a stochastic process, denoted  $\mathbf{Y} = \{Y^{(1)}, Y^{(2)}, \dots, Y^{(n)}\}^T$ . This introduces uncertainty to the known outputs which represents the fact that we do not know the true output at most sets of inputs. This uncertainty is modelled by saying that the value of the function at  $\mathbf{x}$  is like the realization of a random variable  $Y(\mathbf{x})$  that is normally distributed with mean  $\mu$  and variance  $\sigma^2$ .

Assuming the function is smooth and continuous, two points  $\mathbf{x}^{(i)}$  and  $\mathbf{x}^{(j)}$  are close if  $\|\mathbf{x}^{(i)} - \mathbf{x}^{(j)}\|$  is small and then  $Y(\mathbf{x}^{(i)})$  and  $Y(\mathbf{x}^{(j)})$  are likely to be highly correlated. As  $\mathbf{x}^{(i)}$  and  $\mathbf{x}^{(j)}$  move apart the opposite is true. This is encapsulated in the correlation function,

$$\text{Corr} [Y(\mathbf{x}^{(i)}), Y(\mathbf{x}^{(j)})] = \exp \left( - \sum_{l=1}^d \theta_l |\mathbf{x}_l^{(i)} - \mathbf{x}_l^{(j)}|^{p_l} \right). \quad (1)$$

The reader should note that if  $\mathbf{x}^{(i)} = \mathbf{x}^{(j)}$  then the correlation is 1 and zero as  $\|\mathbf{x}^{(i)} - \mathbf{x}^{(j)}\| \rightarrow \infty$ . The parameters  $\theta_l$  and  $p_l$  are known as the hyperparameters, which must be estimated.

The covariance matrix is

$$\text{Cov} (\mathbf{Y}, \mathbf{Y}) = \sigma^2 \Psi, \quad (2)$$

where  $\mathbf{Y}$  is the vector of observed stochastic responses which has a mean of  $\mathbf{1}\mu$  and  $\Psi$  is the correlation matrix for all the observed data,

$$\Psi = \begin{pmatrix} \text{Corr} [Y(\mathbf{x}^{(1)}), Y(\mathbf{x}^{(1)})] & \dots & \text{Corr} [Y(\mathbf{x}^{(1)}), Y(\mathbf{x}^{(n)})] \\ \vdots & \ddots & \vdots \\ \text{Corr} [Y(\mathbf{x}^{(n)}), Y(\mathbf{x}^{(1)})] & \dots & \text{Corr} [Y(\mathbf{x}^{(n)}), Y(\mathbf{x}^{(n)})] \end{pmatrix}. \quad (3)$$

The distribution of  $\Psi$  is dependent on the parameters  $\mu$ ,  $\sigma$  and the hyperparameters  $\boldsymbol{\theta} = \{\theta_1, \theta_2, \dots, \theta_d\}^T$  and  $\mathbf{p} = \{p_1, p_2, \dots, p_d\}^T$ . These parameters are chosen to maximize the likelihood of the observed data. Maximizing the likelihood identifies parameters which model the function's behaviour consistently with the data seen. In practice it is simpler to maximize the natural log of the likelihood function. Ignoring constant terms, this may be written as

$$\ln(L) = -\frac{n}{2} \ln(\sigma^2) - \frac{1}{2} \ln(|\Psi|) - \frac{(\mathbf{y} - \mathbf{1}\mu)^T \Psi^{-1} (\mathbf{y} - \mathbf{1}\mu)}{2\sigma^2}. \quad (4)$$

Expressions for the optimal values of  $\mu$  and  $\sigma^2$  can be found by taking the derivatives of Eq. (4) and setting to zero, yielding

$$\hat{\mu} = \frac{\mathbf{1}^T \Psi^{-1} \mathbf{y}}{\mathbf{1}^T \Psi^{-1} \mathbf{1}}, \quad (5)$$

$$\hat{\sigma}^2 = \frac{(\mathbf{y} - \mathbf{1}\hat{\mu})^T \Psi^{-1} (\mathbf{y} - \mathbf{1}\hat{\mu})}{n}. \quad (6)$$

Substituting Eqs (5) and (6) into Eq. (4) gives the concentrated ln-likelihood function

$$\ln(L) \approx -\frac{n}{2} \ln(\hat{\sigma}^2) - \frac{1}{2} \ln(|\Psi|). \quad (7)$$

The value of this function depends only on  $\Psi$ , and hence, on the hyperparameters  $\theta$  and  $\mathbf{p}$ . Finding optimal values for  $\theta$  and  $\mathbf{p}$  is not simple and cannot generally be achieved through differentiation. Moreover when there is little data, it turns out that  $\ln(L)$  is highly multimodal. This problem requires a direct numerical global optimization.

In order to make a prediction  $\hat{y}$  at some new point  $\mathbf{x}$ , it is also necessary to maximize the likelihood of this prediction. Suppose  $\hat{y}$  is an estimated function value, the quality of this estimate can be evaluated by adding  $\hat{y}$  to the observed data,  $\tilde{\mathbf{y}} = \{\mathbf{y}, \hat{y}\}^T$ , and computing the augmented likelihood function.

The augmented correlation matrix is given as

$$\tilde{\Psi} = \begin{pmatrix} \Psi & \boldsymbol{\psi} \\ \boldsymbol{\psi}^T & 1 \end{pmatrix}, \quad (8)$$

where  $\boldsymbol{\psi}$  is the vector of correlation between the observed data and the prediction,

$$\boldsymbol{\psi} = \begin{pmatrix} \text{Corr}[Y(\mathbf{x}^{(1)}), Y(\mathbf{x})] \\ \vdots \\ \text{Corr}[Y(\mathbf{x}^{(n)}), Y(\mathbf{x})] \end{pmatrix}. \quad (9)$$

Using the optimum parameter values obtained, the augmented likelihood reflects how consistent the estimate is with the observed pattern of variation. The best value for this estimate is therefore the value of  $\hat{y}$  that maximizes the augmented likelihood function. Recalling that  $\boldsymbol{\psi}$  contains our new sample point  $\mathbf{x}$ , the Kriging predictor can thus be expressed as

$$\hat{y}(\mathbf{x}) = \hat{\mu} + \boldsymbol{\psi}^T \Psi^{-1}(\mathbf{y} - \mathbf{1}\hat{\mu}). \quad (10)$$

This predictor is for an interpolating model. In the presence of noise this procedure can be filtered by including a regression constant  $\lambda$ . The regression constant  $\lambda$  is added to the leading diagonal of  $\Psi$  producing  $\Psi + \lambda\mathbf{I}$ , where  $\mathbf{I}$  is the identity matrix. Using the same derivation as before, the regressing Kriging prediction is given by

$$\hat{y}_r(\mathbf{x}) = \hat{\mu}_r + \boldsymbol{\psi}^T (\Psi + \lambda\mathbf{I})^{-1}(\mathbf{y} - \mathbf{1}\hat{\mu}_r), \quad (11)$$

where

$$\hat{\mu}_r = \frac{\mathbf{1}^T (\Psi + \lambda\mathbf{I})^{-1} \mathbf{y}}{\mathbf{1}^T (\Psi + \lambda\mathbf{I})^{-1} \mathbf{1}}. \quad (12)$$

The regression constant  $\lambda$  is treated as an additional hyperparameter and a suitable value found using maximum likelihood estimation.

### 3. Infill sampling criteria with constraint handling

#### 3.1. Expected improvement with probability of feasibility

Using Kriging, or an alternative Gaussian process based model, permits the estimation of model uncertainty. This feature is useful for the selection of update points where the infill criterion can account for those areas of the model that will benefit from some improvement, thus adding an element of

exploration. This is achieved through estimating the regions with a high probability of improvement. The probability of improvement is given as

$$P[I(\mathbf{x})] = \frac{1}{\hat{s}\sqrt{2\pi}} \int_{-\infty}^0 e^{-[I-\hat{y}(\mathbf{x})]^2/(2\hat{s}^2)} dI \quad (13)$$

where  $I$  is a measure of improvement  $I = y_{min} - Y(\mathbf{x})$  and  $\hat{s}^2(\mathbf{x})$  is the predicted variance in the prediction of the Gaussian process based model, given as

$$\hat{s}^2(\mathbf{x}) = \sigma^2 \left[ 1 - \boldsymbol{\psi}^T \boldsymbol{\Psi}^{-1} \boldsymbol{\psi} + \frac{(1 - \mathbf{1}^T \boldsymbol{\Psi}^{-1} \boldsymbol{\psi})^2}{\mathbf{1}^T \boldsymbol{\Psi}^{-1} \mathbf{1}} \right]. \quad (14)$$

The probability of improvement does not indicate how big the improvement may be, it only suggest areas where some improvement may be made. The magnitude of improvement is exposed in the concept know as expected improvement,

$$E[I(\mathbf{x})] = \begin{cases} (y_{min} - \hat{y}(\mathbf{x})) \Phi \left( \frac{y_{min} - \hat{y}(\mathbf{x})}{\hat{s}(\mathbf{x})} \right) + \hat{s} \phi \left( \frac{y_{min} - \hat{y}(\mathbf{x})}{\hat{s}(\mathbf{x})} \right) & \text{if } \hat{s} > 0 \\ 0 & \text{if } \hat{s} = 0 \end{cases} \quad (15)$$

where  $\Phi$  and  $\phi$  are the probability distribution and probability density functions, respectively.  $y_{min}$  is the minimum point sampled so far. In the presence of constraints this should be replaced with  $y_{minfeas}$  indicating the minimum feasible point sampled so far. This concept will guarantee global convergence since an unsampled point will always indicate a predicted variance greater than zero. It results in a positive value of expected improvement at all unsampled points even if the underlying function is poorly modelled.

A straightforward approach to handling constraints considers the addition of a simple penalty to the expected improvement criteria if any constraint surrogate predicts a violation, however an inaccurate constraint model can cause a simple penalty function to be misleading. One way to better handle the constraints is suggested by Schonlau [42]. In his approach the infill criteria uses a product of the expected improvement of the objective function and the probability of feasibility calculated from the constraint functions,

$$E[I(\mathbf{x}) \cap F(\mathbf{x})] = E[I(\mathbf{x})] P[F(\mathbf{x}) > g_{limit}].$$

The probability of feasibility is calculated in the same manner as probability of improvement, however this identifies regions of feasibility, i.e. the probability the prediction will be greater or less than a constraint limit. The probability of feasibility for a single constraint is given as

$$P[F(\mathbf{x})] = \frac{1}{\hat{s}\sqrt{2\pi}} \int_0^{\infty} e^{-[(G(\mathbf{x}) - g_{limit}) - \hat{y}(\mathbf{x})]^2/(2\hat{s}^2)} dG, \quad (16)$$

where  $g$  is the constraint function,  $g_{limit}$  is the constraint limit and  $G(\mathbf{x}) - g_{limit}$  is the measure of feasibility.

This method will gradually drive the infill criteria towards zero as the search transitions between feasible and infeasible regions, adaptively softening the sheer cliff landscape produced by a one pass penalty function. Figure 1 shows a comparison of this method against a penalty approach for a simple product constraint applied to the Branin function (see Section 5.3). The contour lines show a smooth transition between feasible and infeasible space.

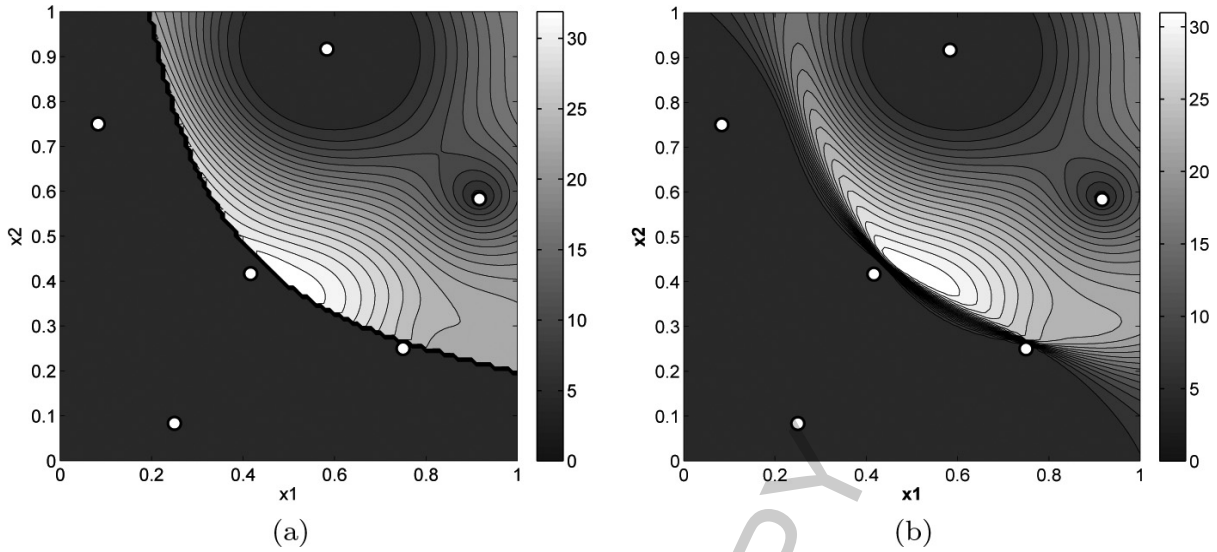


Fig. 1. Penalty and probabilistic constraint handling on the Branin function subject to a simple product constraint.  $\circ$  – sample data. (a) Penalty approach  $E[I(\mathbf{x})] - P$  (b) Probabilistic approach  $E[I(\mathbf{x}) \cap F(\mathbf{x})]$ .

In cases where more than one constraint is applied, the total probability of feasibility is the product of all individual constraint probabilities of feasibility. Forrester et al. [16] refer to this method as constrained expected improvement.

For simple constraints this method is expected to offer little improvement over the one pass penalty function. For more complicated constraints or deceptive constraint functions this infill criterion will judiciously balance exploration and exploitation without the use of arbitrary penalty functions.

### 3.2. Expected improvement versus probability of feasibility

Rather than transforming the constrained problem into an unconstrained one, an alternative is to treat the two properties as individual goals and explicitly consider trade-offs between objectives and constraints using multiobjective optimization. This concept of treating the constraint as an objective is explored in traditional optimization using filters [4,13] and in evolutionary algorithms using sub populations for each constraint [8] or using nondominance to rank constraint violation [3,29,36]. Parr et al. [31] use multiobjective optimization to better balance the measures of expected improvement and probability of feasibility, forming a set of potential update points as a Pareto set.

Figure 2 shows an example of the two goals presented as a Pareto front. In this figure it is obvious that any increase in expected improvement of the objective will be detrimental to the probability of feasibility of the constraint. Conversely, any reduction in the probability of feasibility will give a smaller expected improvement.

Constructing the Pareto set gives the designer a flexible choice of update points. The designer may make a weighted judgment depending on the complexity of the constraints and objective function, or on problems with a higher evaluation budget the designer can choose multiple updates from the Pareto set [31]. With the exception of Section 7, we only consider single update points in this paper. Since we have a set of potential update points it is necessary to select only one. Here we choose the individual point from the Pareto set with the maximum product of  $E[I(\mathbf{x})]$  and  $P[F(\mathbf{x})]$ . In this study this method

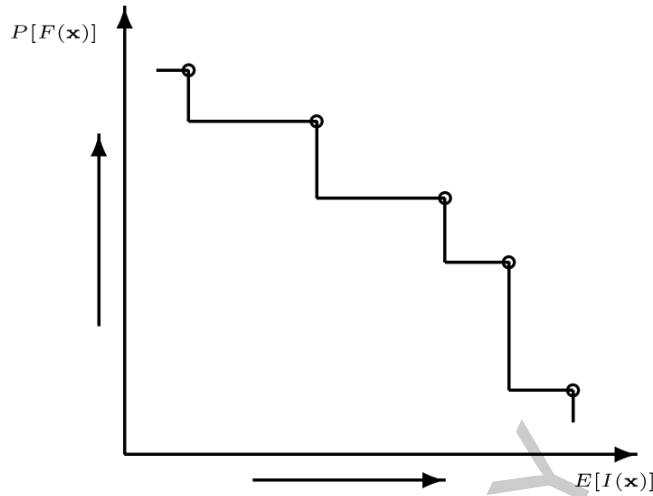


Fig. 2. Pareto front when maximizing both  $E[I(\mathbf{x})]$  and  $P[F(\mathbf{x})]$ :  $\circ$  – Pareto set.

is referred to by using the notation  $E[I(\mathbf{x})]$  vs  $P[F(\mathbf{x})]$ . An important note here is that in the single objective approach,  $E[I(\mathbf{x}) \cap F(\mathbf{x})]$  can be considered as an unknown weighted sum of  $E[I(\mathbf{x})]$  and  $P[F(\mathbf{x})]$ . Intuitively, by searching  $E[I(\mathbf{x}) \cap F(\mathbf{x})]$  directly as a single objective the solution will be nondominated and expected to lie somewhere on the Pareto front. Typically you would expect this solution to correspond to the optimum point found using  $E[I(\mathbf{x})]$  vs  $P[F(\mathbf{x})]$  when we choose the point from the Pareto set with maximum product of  $E[I(\mathbf{x})]$  and  $P[F(\mathbf{x})]$ , however when the aggregation causes the landscape to be severe and complex, treating the problem as a multiobjective one can achieve better solutions. This is a commonly observed phenomenon [9,47] and several authors have even reformulated single objective problems so they can be solved using multiobjective methods [19,24,27].

To clarify this argument Fig. 3 shows a comparison of the two approaches on the Branin function subject to a complex constraint (see Section 5.3). This figure is drawn after a number of updates so the feasible regions and areas of good objective values have already been identified. As hinted earlier the search space for the expected improvement and probability of feasibility can be far from simple and when combined as a single objective,  $E[I(\mathbf{x}) \cap F(\mathbf{x})]$ , the search space is complex with several local optima and cliffs bounding the regions of feasibility. The global maximum of  $E[I(\mathbf{x}) \cap F(\mathbf{x})]$  is then very difficult to find, situated on top of a small peak with steep edges, Fig. 3(c). The single objective approach uses a hybrid search of a 5000 evaluation genetic algorithm (GA) [18] and a 5000 evaluation dynamic hill climb (DHC) [48]. The multiobjective method uses a standard implementation of NSGA-II [11]. Both these algorithms are run with population size of 100 with 10000 total evaluations. Figure 3(d) shows the maximum  $E[I(\mathbf{x}) \cap F(\mathbf{x})]$  identified using each method. The single objective search does not find the global solution. Perhaps this is as expected since the global optimum lies in a very severe location. Using  $E[I(\mathbf{x})]$  vs  $P[F(\mathbf{x})]$ , the problem is approached in a different manner and the search can take different directions which are otherwise limited by the complex  $E[I(\mathbf{x}) \cap F(\mathbf{x})]$  design space. NSGA-II finds a set of solutions clustered in different areas of the design space and selecting the solution with the largest product of  $E[I(\mathbf{x})]$  and  $P[F(\mathbf{x})]$  corresponds to the global solution.

It is important to note that when using  $E[I(\mathbf{x})]$  vs  $P[F(\mathbf{x})]$ , further constraints will add additional probability of feasibility objectives, increasing the dimensionality of the Pareto front. Multiobjective problems with more than four objectives are classified as a many-objective optimization problem and are inherently difficult to solve [23,33,43]. In problems with more than a few constraints, the probability of

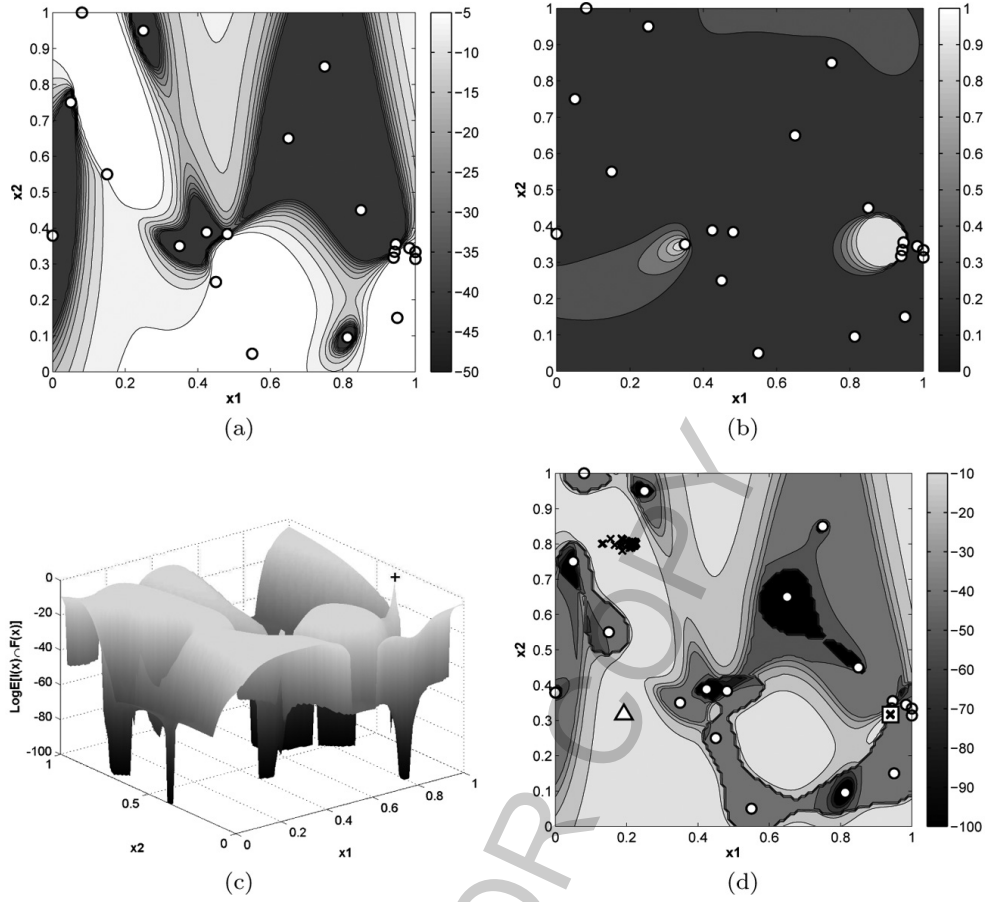


Fig. 3. Comparative search on the Branin function subject to a complex constraint.  $\circ$  – sample data. (a)  $\text{Log}E[I(\mathbf{x})]$ . (b)  $P[F(\mathbf{x})]$ . (c)  $\text{Log}E[I(\mathbf{x}) \cap F(\mathbf{x})]$  search space. + – global optimum. (d) Optimum locations found using  $\triangle$  – single objective  $E[I(\mathbf{x}) \cap F(\mathbf{x})]$  search,  $\square$  –  $E[I(\mathbf{x})]$  vs  $P[F(\mathbf{x})]$  multiobjective search,  $\times$  –  $E[I(\mathbf{x})]$  vs  $P[F(\mathbf{x})]$  Pareto set.

feasibility of each constraint can be multiplied together denoted as  $E[I(\mathbf{x})]$  vs  $\prod P[F(\mathbf{x})]$ . This reduces the problem to two objectives, one for the objective improvement and the other for the satisfaction of all constraints.

#### 4. Enhanced constraint handling

In particular problems a designer may need to find a design with a certain level of improvement or seek a design that achieves a specific response rather than finding the global optimum. In traditional optimization methods, it is common to treat any active inequality constraints as equality constraints and inactive constraints are ignored [35]. Keeping these points in mind, it becomes sensible to only construct surrogate models for the active constraints and attempt to reduce the prediction error explicitly along the constraint boundaries. As long as the region of the global optimum has been identified, large errors in the constraint approximation away from the constraint boundary become unimportant.

Jones [21] considers the likelihood that an objective function value or goal ‘could’ exist at a given level. Convergence towards this goal can be achieved by basing an infill criterion on the maximum



conditional likelihood, known as goal-seeking [16]. Ranjan and Bingham [34] provide an extension to the expected improvement criterion for contour estimation. This criterion balances exploitation of a region surrounding a contour and exploitation of the design space in areas of high uncertainty. In a similar manner, Picheny et al. [32] describe an infill criterion based on the integrated mean squared error criterion that encourages updates in the vicinity of a boundary, commenting on its potential application to constraint handling.

Bichon et al. [5] introduce a formulation of expected feasibility. This provides an indication of how well a response is expected to satisfy an equality constraint. Here we investigate a similar formulation to enhance the probability of feasibility criterion.

The probability of feasibility has been demonstrated to be a favourable method when dealing with constraints in surrogate based optimization [30]. The standard formulation, Eq. (16), identifies regions of feasibility, i.e. the probability the prediction will be greater than a constraint limit. If all inactive constraints are ignored the remaining constraints can be treated as equality constraints and the accuracy of the model away from the constraint boundary becomes less important. To encourage exploitation of the region in the vicinity of a target value (or contour) it is sensible to integrate the probability density function over the interval  $[T - \epsilon, T + \epsilon]$ , where  $T$  is the target value [32]. When exploiting a constraint boundary, integrating over  $[g_{limit} - \epsilon, g_{limit} + \epsilon]$  falsely suggests a region of feasibility in the vicinity below the constraint limit, therefore the bottom bound should be omitted. Furthermore, integrating over the interval  $[g_{limit}, g_{limit} + \epsilon]$  rather than  $[g_{limit} - \epsilon, g_{limit} + \epsilon]$  encourages the designs to be selected on the feasible side of the constraint limit and retains the smoothing property associated with the probability of feasibility along the constraint boundary. The enhanced probability of feasibility is therefore

$$P[F(\mathbf{x})]_{\epsilon} = \frac{1}{\hat{s}\sqrt{2\pi}} \int_{g_{limit}}^{g_{limit}+\epsilon} e^{-[(G(\mathbf{x})-g_{limit})-\hat{g}(\mathbf{x})]^2/(2\hat{s}^2)} dG, \quad (17)$$

where  $g$  is the constraint function,  $g_{limit}$  is the constraint limit and,  $G(\mathbf{x}) - g_{limit}$  is the measure of feasibility. Allowing  $g_{limit}^+$  to represent  $g_{limit} + \epsilon$  the enhanced probability of feasibility can be expressed as

$$P[F(\mathbf{x})]_{\epsilon} = \Phi\left(\frac{\hat{g}(\mathbf{x}) - g_{limit}}{\hat{s}(\mathbf{x})}\right) - \Phi\left(\frac{\hat{g}(\mathbf{x}) - g_{limit}^+}{\hat{s}(\mathbf{x})}\right), \quad (18)$$

where  $\Phi$  is the cumulative distribution function. This derivation is analogous to that found in Picheny et al. [32].

In this formulation, the user is required to choose a value of  $\epsilon$ . Figure 4 demonstrates the influence of this parameter on a simple product constraint (see Section 5.3). A low value of  $\epsilon$  results in exploitation of the constraint boundary prediction. At a higher value of  $\epsilon$  the enhanced probability of feasibility is clearly more conservative and generalises towards the standard probability of feasibility as  $\epsilon \rightarrow \infty$ .

This method allows the user some flexibility between exploiting and exploring the constraint boundaries, however choosing a single value for  $\epsilon$  is not intuitive. In similar work, Ranjan and Bingham [34] define  $\epsilon$  as a function of the prediction variance,  $\hat{s}(\mathbf{x})$ . Basing  $\epsilon$  on  $\hat{s}(\mathbf{x})$  encourages a wider interval in regions of sparse sampling and a smaller interval where the predicted variance is small. In this study  $\epsilon$  takes a value of 5% as suggested by Picheny et al. [32]. Since the true output range for each constraint is unlikely to be known,  $\epsilon$  is taken as 5% of the known output range simply given by the difference between  $g_{max}$  and  $g_{min}$ , the maximum and minimum sampled values. These enhanced methods are denoted as  $E[I(\mathbf{x}) \cap F(\mathbf{x})]_{5\%}$  and  $E[I(\mathbf{x})]_{vs}P[F(\mathbf{x})]_{5\%}$  respectively.

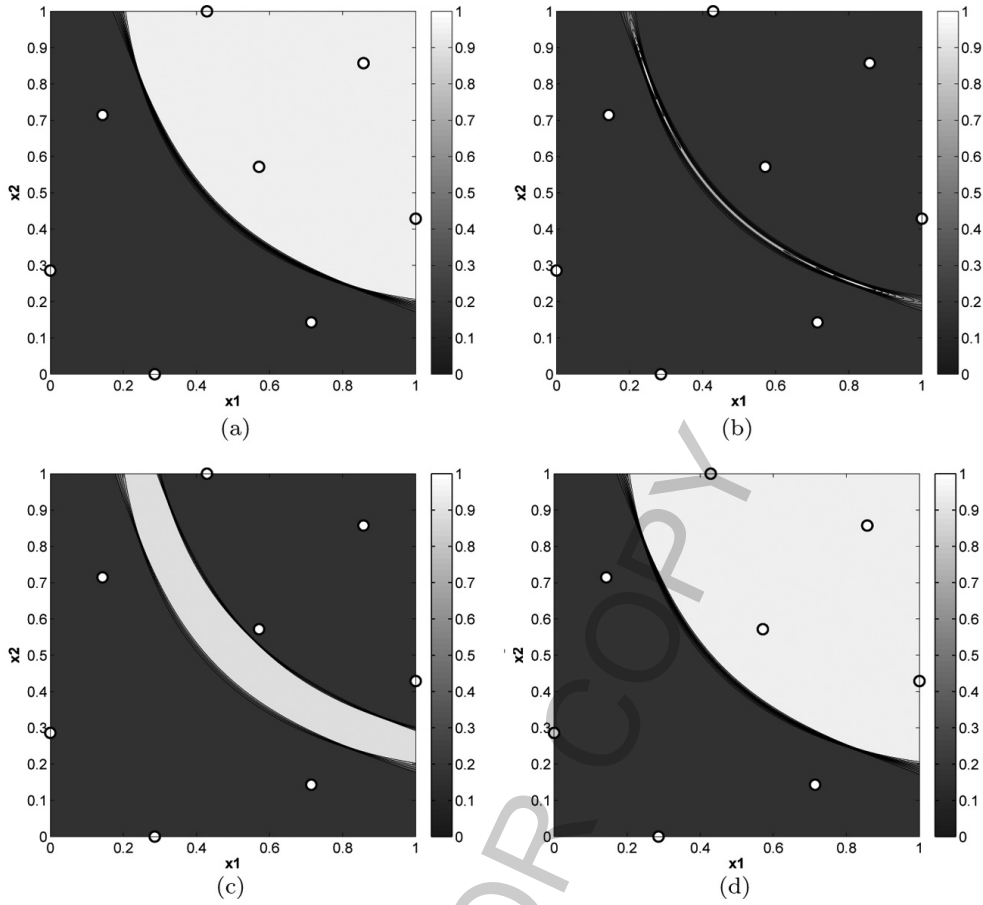


Fig. 4. Product constraint with  $g_{limit} = 0.2$ ,  $\circ$  – sample data. The standard probability of feasibility (a)  $P[F(\mathbf{x})]$ , and enhanced methods (b)  $P[F(\mathbf{x})]_{0.01}$  (c)  $P[F(\mathbf{x})]_{0.1}$  (d)  $P[F(\mathbf{x})]_1$ .

## 5. Test problems

### 5.1. Model fitting and search algorithms

Building a Kriging model requires sufficient hyperparameter tuning. Different tuning strategies have been examined by Toal et al. [46]. The light tune strategy suggested there consists of a hybrid search using a 1000 evaluation GA and a 1000 evaluation DHC to maximize the concentrated likelihood function, Eq. (7). This strategy has been adopted for hyperparameter tuning for all objective surrogates and all constraint surrogates used in this study. The hyperparameters are tuned after the initial DoE and retuned after every update stage.

As we have seen a global optimization is required to search for each update point. Here a heavy hybrid search is used to maximize  $E[I(\mathbf{x}) \cap F(\mathbf{x})]$ , consisting of a 5000 evaluation GA and a 5000 evaluation DHC.

Adopting the  $E[I(\mathbf{x})]$  vs  $P[F(\mathbf{x})]$  approach involves the formulation of a Pareto set. The nondominated sorting GA (NSGA-II), introduced by Deb et al. [12,45], is a popular method for multiobjective optimization. We aim to maximize the expected improvement of the objective function and the probability of

feasibility of the constraints. As noted previously, the number of objectives evaluated by NSGA-II is equal to the number of functions being approximated. For example, a single objective problem with two constraints will have a surrogate model for each constraint and one for the original objective function. This gives a multiobjective problem between three objectives, maximizing  $P[F(\mathbf{x})]_1$ ,  $P[F(\mathbf{x})]_2$  and  $E[I(\mathbf{x})]$ . It is also possible to consider a bi-objective approach where all the constraints are lumped into a single objective,  $E[I(\mathbf{x})]$  vs  $\prod P[F(\mathbf{x})]$ . This is used in the design of a transonic wing in Section 8. The artificial test problems encountered in this paper use NSGA-II with a population of 200 and 100 generations. For the two wing design problems a population of 400 and 200 generations is used. This produces a highly populated Pareto set with a wide choice of update points. For a single update the member of the Pareto set with the highest product of expected improvement and probability of feasibility is selected.

## 5.2. Comparison metrics

The performance of the infill criteria can be assessed in a number of ways. Here this performance is measured using the true sample data, the known outputs  $\mathbf{y} = \{y^{(1)}, y^{(2)}, \dots, y^{(d)}\}^T$  and the chosen set of inputs  $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(d)}$ , where  $d$  includes both the initial DoE and all true update samples. If the true optimum is known, an intuitive method of comparison is to find the absolute error between the true optimum,  $y^*$  at  $\mathbf{x}^*$ , and the best feasible point sampled so far,  $y^{best}$  at  $\mathbf{x}^{best}$ . In real engineering design problems the true optimum is, in general, unknown and  $y^*$  should be replaced with the best known feasible solution. The absolute error in the objective space in either case is then

$$|y^* - y^{best}| = \sqrt{(y^* - y^{best})^2}. \quad (19)$$

The performance of the infill sampling criteria depends on the placement of points in the initial sample. In some cases, by pure luck, this initial sample may include a point close to the global optimum, accelerating the search. To avoid any bias when testing, here each method uses the same initial sample. For the artificial test problems and the Nowacki beam design the results are repeated on a set of 100 random Latin Hypercube initial samples. For the aircraft wing box and transonic wing design problems, the methods are repeated 50 times on a set of optimized random Latin Hypercubes using the Morris-Mitchell criterion [26].

Given a limited evaluation budget the designer will also be concerned about the accuracy and reliability of each method. After repeating the search on a number of different initial samples the performance of each approach can be represented in terms of a probability. This probability characterizes the consistency of each method in achieving an optimum solution within a specified accuracy. By varying the specified accuracy this metric will demonstrate both the efficiency and reliability of each method in finding the region of the global optimum and the exact global optimum. Bootstrapping, a statistical resampling method, is applied to the sets of results to calculate the average probability and the 95% confidence intervals [7].

## 5.3. Constrained Branin function

Here a modified version of the Branin function is minimized, see Forrester et al. [16]. The first test problem is concerned with minimizing the Branin function subject to a simple inequality product constraint,

$$g = x_1 x_2 \quad (20)$$

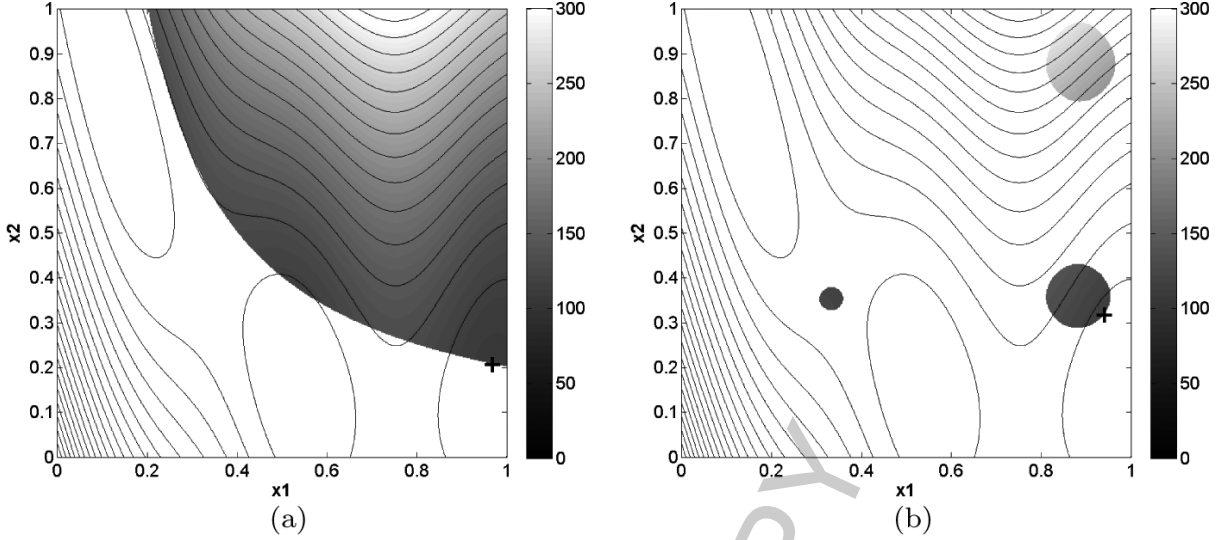


Fig. 5. Bounded feasible space. (a) Product constraint + - global optimum. (b) Gomez#3 constraint + - global optimum.

where  $x_1, x_2 \in [0, 1]$ . For the constraint to be satisfied,  $g > 0.2$ . In the second problem the complexity of the constraint is increased. The constraint function is a normalized version of the Gomez#3 function [41], with an additional sinewave to increase modality,

$$g = \left(4 - 2.1x_1^2 + \frac{1}{3}x_1^4\right)x_1^2 + x_1x_2 + (-4 + 4x_2^2)x_2^2 + 3\sin[6(1-x_1)] + 3\sin[6(1-x_2)] \quad (21)$$

where  $x_1, x_2 \in [-1, 1]$ . For the constraint to be satisfied,  $g > 6$ .

The bounded feasible space and the location of the global optimum for both of these problems is presented in Fig. 5.

The simple product constraint problem is tackled using an initial sample of 8 points and 20 updates. The complex Gomez#3 constraint problem uses an initial sample with 10 points and 30 updates. Figures 6 and 7 show the results for these two test problems. Results for the simple product constraint show there is little statistical significance between all the methods when finding a solution to a lower accuracy. As the accuracy demanded of the final optimum is increased, the enhanced methods show a clear improvement in performance with  $E[I(\mathbf{x})]$  vs  $P[F(\mathbf{x})]_{5\%}$  performing the best.

For the complex Gomez#3 constraint, it is clear the multiobjective approach significantly outperforms the single objective  $E[I(\mathbf{x}) \cap F(\mathbf{x})]$  methods, however, using the enhanced probability of feasibility has not made any significant improvements on the performance of either method.

#### 5.4. Sasena problem

To investigate the performance of these enhanced methods for multiple constraints a third artificial test function is introduced. This deceptive test function, originally used by Sasena et al. [40], has two active and one inactive constraint. Here the inactive constraint is ignored and the minimization of the constrained problem is defined as,

$$f = -(x_1 - 1)^2 - (x_2 - 0.5)^2, \quad (22)$$

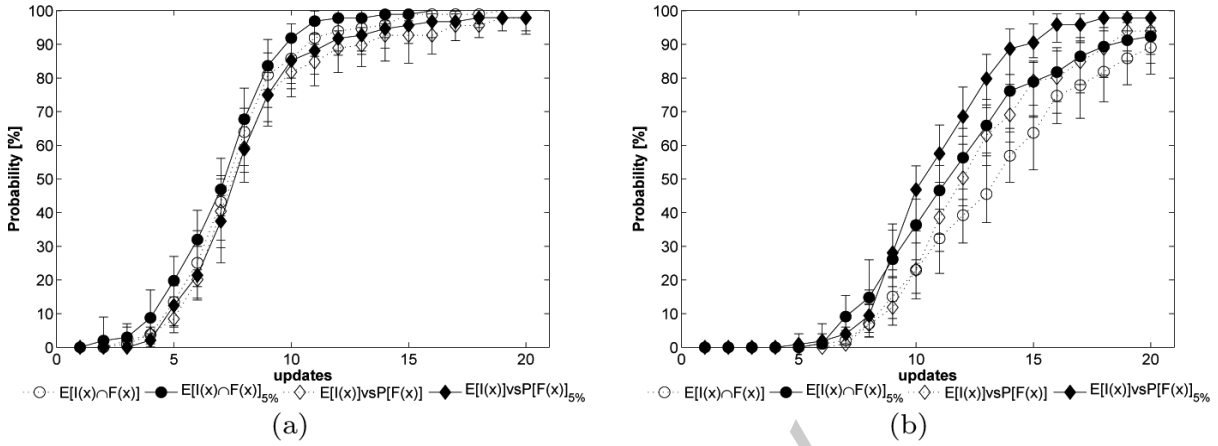


Fig. 6. Branin function subject to the product constraint. Mean probability and 95% confidence intervals of the best feasible point being within (a) 0.1% of the true optimum value (b) 0.01% of the true optimum value.

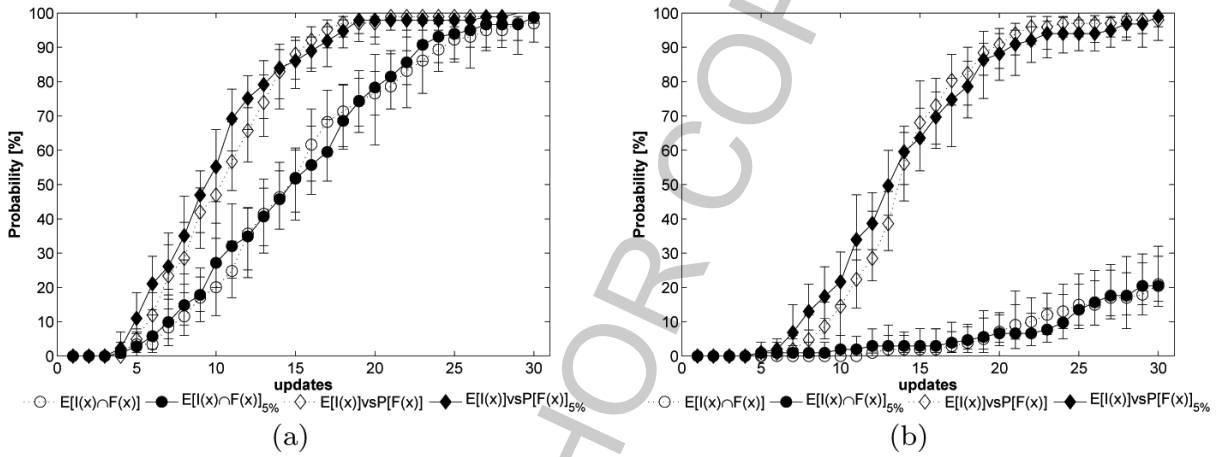


Fig. 7. Branin function subject to the complex Gomez#3 constraint. Mean probability and 95% confidence intervals of the best feasible point being within (a) 0.1% of the true optimum value (b) 0.01% of the true optimum value.

subject to,

$$g_1 = 12 - ((x_1 - 3)^2 + (x_2 + 2)^2) e^{-x_2^7}, \tag{23}$$

$$g_2 = 0.2 - (x_1 - 0.5)^2 - (x_2 - 0.5)^2, \tag{24}$$

where  $x_1, x_2 \in [0, 1]$ , For the constraints to be satisfied,  $g_1, g_2 > 0$ .

The bounded feasible space and the location of the global optimum for this problem is presented in Fig. 8.

The Sasena problem is tackled using an initial sample of 10 and 30 updates. Figure 9 shows the results for this problem. At a lower accuracy, there is little difference in the performance of  $E[I(\mathbf{x})] vs P[F(\mathbf{x})]$  and  $E[I(\mathbf{x})] vs P[F(\mathbf{x})]_{5\%}$  but the enhanced method does improve the performance of  $E[I(\mathbf{x}) \cap F(\mathbf{x})]$ . At a higher accuracy, there is little difference in the performance of  $E[I(\mathbf{x}) \cap F(\mathbf{x})]$  and  $E[I(\mathbf{x}) \cap$

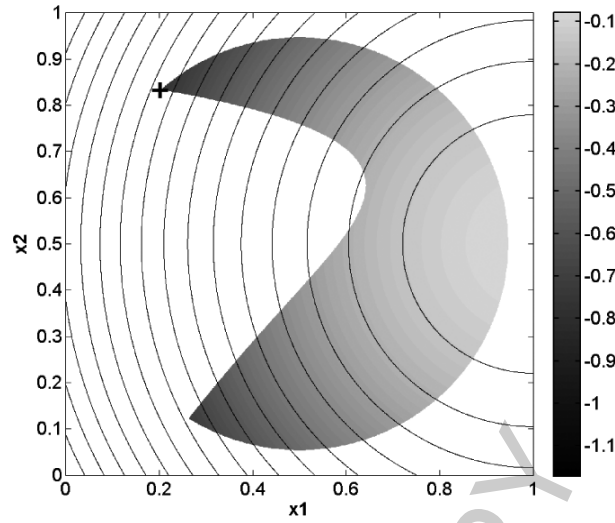


Fig. 8. Bounded feasible space for Sasena problem. + – global optimum.

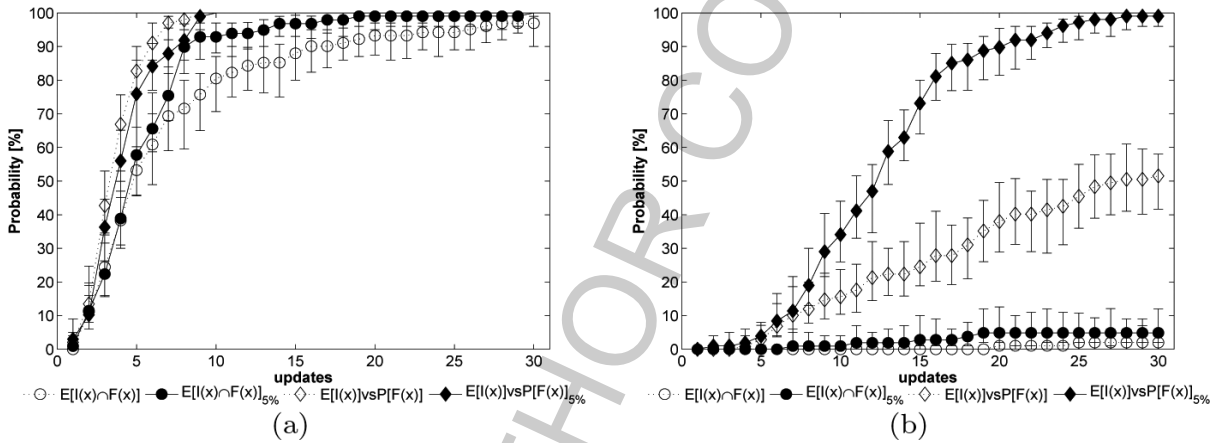


Fig. 9. Sasena problem. Mean probability and 95% confidence intervals of the best feasible point being within (a) 0.1% of the true optimum value (b) 0.01% of the true optimum value.

$F(x)]_{5\%}$ , however the results show a significant improvement in the performance using the enhanced approach where  $E[I(x)]vsP[F(x)]_{5\%}$  is twice as reliable and much more efficient than the standard  $E[I(x)]vsP[F(x)]$  approach.

### 6. Nowacki beam design

Next a set of real engineering design problems are tackled. First a relatively simple example based on a design problem described by Nowacki [28]. This involves the design of a tip-loaded encastre cantilever beam with a rectangular cross-section with breadth  $b$  and height  $h$ . The objective is to minimize the cross-sectional area with constraints placed on the tip deflection and height to breadth ratio  $h/b$ . The beam has a fixed length  $l = 1.5$  m with a tip load  $F = 5$  kN. The beam is made from a mild steel

Table 1

Initial design parameters, constraint values, and objective value.  $l = 1.5\text{m}$  and Area  $A = bh$ 

Lower limit	Value	Upper limit	Quantity
0.005	0.0275	0.05	Breadth, m
0.02	0.135	0.25	Height, m
–	0.0046	0.005	Tip deflection $\delta = Fl^3/(3EI_Y)$ , m
–	4.91	10	Height to breadth ratio
–	0.0037	–	Cross-sectional area, $\text{m}^2$

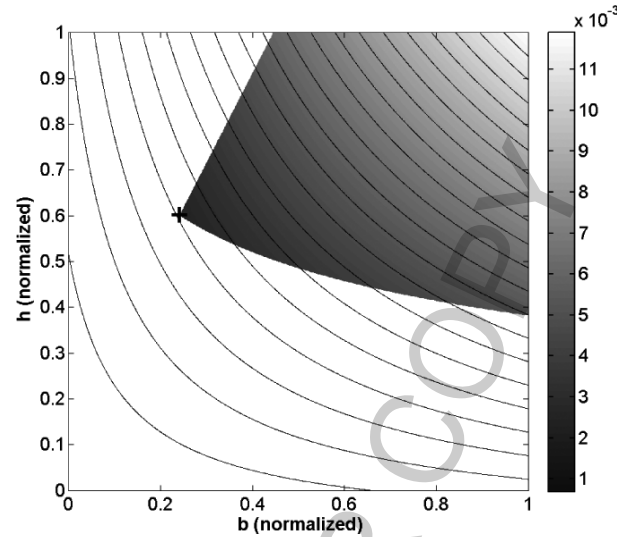


Fig. 10. Bounded feasible space for Nowacki beam problem. + – global optimum.

with yield stress  $\sigma_Y = 240\text{ MPa}$  and Young's modulus  $E = 216.62\text{ GPa}$ . The design parameters and limits are listed in Table 1. The bounded feasible design space and the location of the global optimum is pictured in Fig. 10.

This problem uses an initial sample of 8 and has 20 further updates. Figure 11 demonstrates the efficiency and reliability of each method when finding solutions to two different accuracies. When finding a solution within  $100\text{ mm}^2$  of the global solution, there is little to distinguish the performance of the single and multiobjective methods, however there is a clear benefit when employing the enhanced probability of feasibility. This distinct improvement is also seen when we consider a solution to a higher accuracy. In a similar manner to the Sasena problem we also see a large improvement when using the multiobjective approach.

## 7. Wing box design problem with multiple updates

The next engineering design problem considers the optimization of a simplified wing box structure [31]. Here the optimization problem is to minimize a single weight objective with limits placed on the key geometry variables and constraints on the structural stress and deformation. To keep this problem manageable only a single element stress at the root of the wing box and a single node displacement at the tip are constrained. Although this may not accurately represent the location of maximum stress and

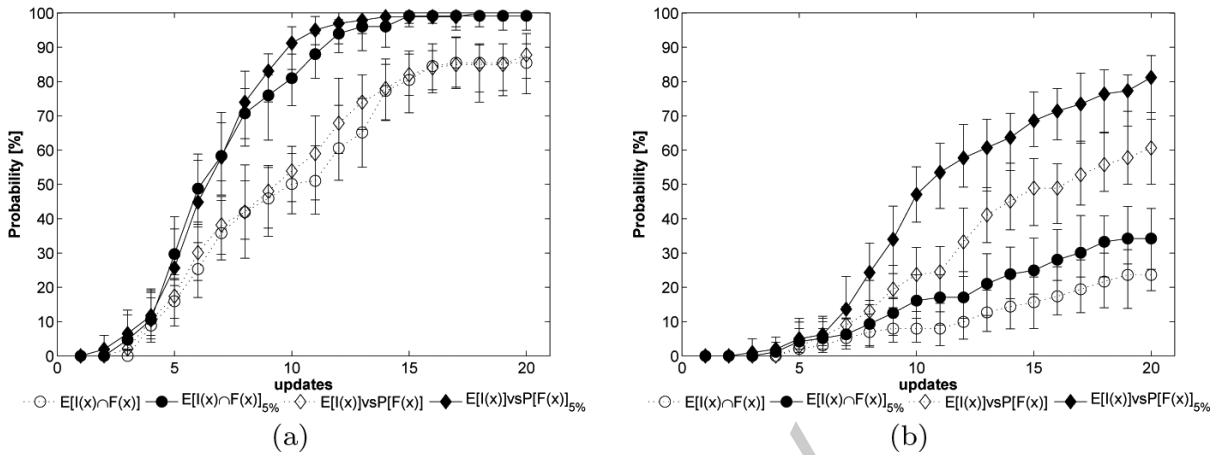


Fig. 11. Results from the Nowacki beam optimization showing the mean probability and 95% confidence intervals of the best feasible point being within (a)  $100\text{mm}^2$  of the best known solution (b)  $10\text{mm}^2$  of the best known solution.

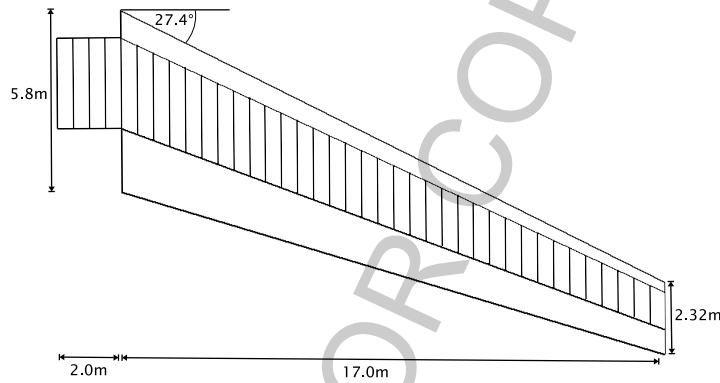


Fig. 12. Baseline wing planform and wing box

maximum deformation on every design iteration, these constraints are representative and suitable for the aims of this study.

The wing box model is composed of two spars, ribs and upper and lower surface skins. The front and rear spar are positioned at 15% and 65% chord respectively. The spacing between the ribs is kept constant to eliminate the number of ribs as a design variable. The upper and lower surface skins are assumed to have equal thickness and the root chord is kept constant. There is also a central fuselage mounting box composed of four ribs and two spars which is kept the same for every design iteration.

Figure 12 shows typical values for a straight sweep transonic civil transport wing. The design variables and limits are listed in Table 2, along with values for (an infeasible) baseline wing.

The wing box structure stresses and deformation are calculated using linear elastic analysis in ABAQUS. An encastre boundary condition is placed at the box root and aerodynamic loads are simulated by applying an elliptical pressure distribution to the main wing box with a total force equal to 20000 N, simulating a fixed lifting force.

Since this is simulating a real engineering design problem, the exact location of the true optimum is unknown. The best known feasible solution is a value of 351.5 kg identified using the  $E[I(\mathbf{x})]vsP[F(\mathbf{x})]$  criterion.



Table 2  
Initial design parameters, constraint values, and objective value. Root chord = 5.8 m

Lower limit	Value	Upper limit	Quantity
10	17	20	Span, m
0.3	0.4	1	Taper ratio
0	25	45	Quarter chord sweep angle, deg
0.002	0.008	0.01	Spar thickness, m
0.002	0.006	0.01	Rib thickness, m
0.002	0.003	0.01	Skin thickness, m
–	287	150	Element Von Mises stress, MPa
–	2.86	1.5	Node deflection, m
–	1350.2	–	Wing box weight, kg

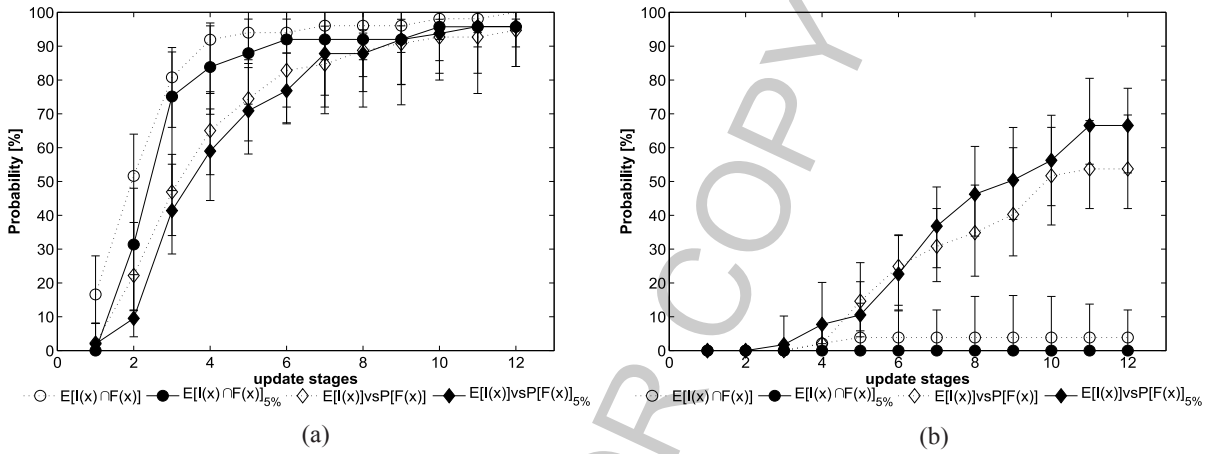


Fig. 13. Results from wing box optimization showing the mean probability and 95% confidence intervals of the best feasible point being within (a) 10 kg of the best known solution (b) 1kg of the best known solution.

This problem uses an initial sample of 40 points but evaluates multiple updates at each stage.  $E[I(\mathbf{x})] vs P[F(\mathbf{x})]$  lends itself to the selection of multiple updates due to the formation of a Pareto set. In this method the selection of  $m$  updates is performed in a two step process. First the Pareto set is grouped into  $m$  clusters in the design vector space using k-means clustering [2], then a single member with the highest product of  $E[I(\mathbf{x})]$  and  $P[F(\mathbf{x})]$  is selected from each cluster. This gives  $m$  update points which are likely to be spread along the Pareto front whilst retaining some flexibility if one region appears more promising than another. Selecting multiple updates using  $E[I(\mathbf{x}) \cap F(\mathbf{x})]$  is less straightforward. Here the Kriging Believer strategy is used [17]. In order to find a set of update points without evaluating the true responses the Kriging Believer substitutes the unknown for a dummy value given by the current Kriging predictor. This enables a set of updates to be identified in a sequential process. In both cases  $m = 6$  updates are evaluated in parallel at each stage for 12 update stages, totalling a further 72 updates.

Figure 13 shows the results for this wing box structure design problem. It is clear that at a low accuracy, the enhance probability of feasibility makes little difference to the performance of both the single and multiobjective methods. As the accuracy is increased,  $E[I(\mathbf{x})] vs P[F(\mathbf{x})]_{5\%}$  outperforms the other methods, mirroring the benefits also seen in the Nowacki beam problem.

Table 3  
Initial design parameters, constraint values, and objective value

Lower limit	Value	Upper limit	Quantity
100	168	250	Wing area, m <sup>2</sup>
6	9.07	12	Aspect ratio
0.2	0.313	0.45	Kink position
25	27.1	45	Sweep angle, deg
0.4	0.598	0.7	Inboard taper ratio
0.2	0.506	0.6	Outboard taper ratio
0.1	0.150	0.18	Root <i>t/c</i>
0.06	0.122	0.14	Kink <i>t/c</i>
0.06	0.122	0.14	Tip <i>t/c</i>
4.0	4.5	5.0	Tip washout, deg
0.65	0.75	0.84	Kink washout fraction
–	127984	135000	Wing weight, N
40.0	41.73	–	Wing volume, m <sup>3</sup>
–	4.179	5.4	Pitch-up margin
2.5	2.693	–	Undercarriage bay length, m
–	3.145	–	D/q, m <sup>2</sup>

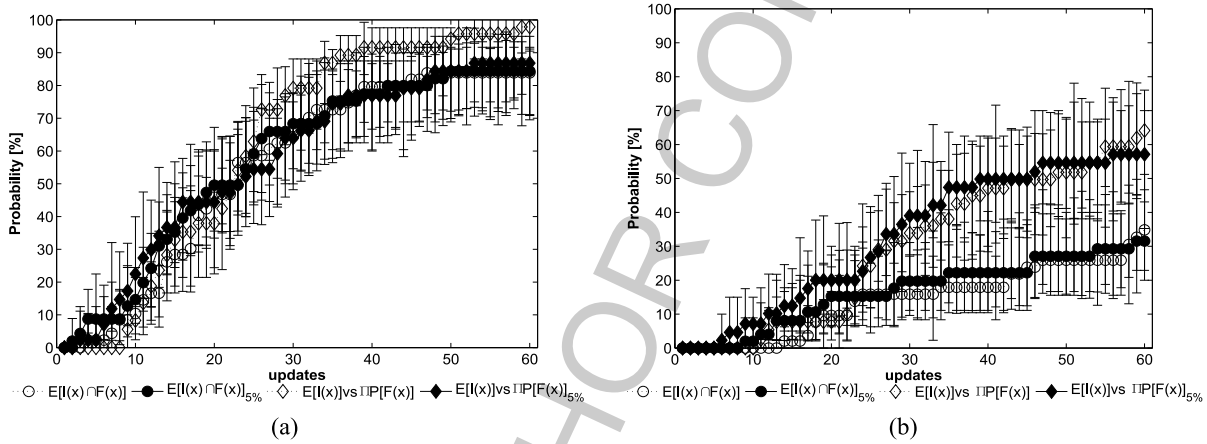


Fig. 14. Results from wing design tool simulations. (a) Mean probability and 95% confidence intervals of the minimum feasible  $D/q$  being within 0.01 of the best known solution. (b) Mean probability and 95% confidence intervals of the minimum feasible  $D/q$  being within 0.005 of the best known solution.

## 8. Aircraft wing design problem

As pointed out earlier, there are inherent problems when solving multiobjective problems with more than four objectives. This limits the use of  $E[I(\mathbf{x})] \text{ vs } P[F(\mathbf{x})]$  for heavily constrained problems. It is possible to simplify such searches by reducing the problem to only two objectives using  $E[I(\mathbf{x})] \text{ vs } P[F(\mathbf{x})]$ . Although this approach loses any benefits of dealing with each constraint separately, it makes the multiobjective method more tractable when dealing with multiple constraints. Here this approach is used to tackle a problem dominated by constraints.

In this case study the aircraft performance is computed using a lightweight version of a former Airbus conceptual design tool [10]. The optimization problem is simplified to a single low wing drag objective and limits are placed on key geometry variables and constraints exist on the wing weight, fuel tank volume, pitch-up margin and undercarriage bay length. Overall aircraft weight is adjusted to allow for

wing strength by a wing weight and sizing analysis. Table 3 shows typical values, and limits, for a transonic civil transport wing with 11 design variables. The table also includes the low drag objective and constraint function values calculated using the wing design tool.

For this problem the best feasible solution known is  $D/q = 2.758 \text{ m}^2$ , at a point where all four constraints are active. An initial sample of 110 points is used with a further 60 single updates.

Figure 14 shows that at a low accuracy the enhanced method in fact hinders the performance of  $E[I(\mathbf{x})]$  vs  $\prod P[F(\mathbf{x})]$ , however at a higher accuracy  $E[I(\mathbf{x})]$  vs  $\prod P[F(\mathbf{x})]$  and  $E[I(\mathbf{x})]$  vs  $\prod P[F(\mathbf{x})]_{5\%}$  share a very similar performance. Although the enhanced approach does not improve the performance, there is a clear indication that the lumped multiobjective approach can still outperform the single objective method.

## 9. Conclusions

This study compares single and multiobjective approaches to dealing with constraints in surrogate-based optimization. By avoiding any aggregation of the objective improvement and constraint satisfaction the multiobjective approach  $E[I(\mathbf{x})]$  vs  $P[F(\mathbf{x})]$  outperforms the single objective approach  $E[I(\mathbf{x}) \cap F(\mathbf{x})]$  in every problem tested when finding a solution to a high accuracy.

To encourage the development of the constraint boundaries, an enhanced probability of feasibility is introduced. This enhancement exploits regions in close proximity to constraint boundaries. Performance gains are seen on four out of the six problems tackled, where  $E[I(\mathbf{x})]$  vs  $P[F(\mathbf{x})]_{5\%}$  outperforms all other methods when finding a solution to a high accuracy.

Further work is required to examine the full potential of these enhanced methods and further investigation into the correct value of the constraint width exploitation parameter  $\varepsilon$  and extension to equality constraints will prove beneficial.

## References

- [1] N. Alexandrov, J.E. Denis, R.M. Lewis and V. Torczon, A trust region framework for managing the use of approximation models in optimization, *Structural and Multidisciplinary Optimization* **15** (1998), 16–23.
- [2] M.R. Anderberg, *Cluster Analysis for Applications*, Academic Press, 1973.
- [3] A. Angantyr, J. Andersson and J.O. Aidanpaa, Constrained optimization based on a multiobjective evolutionary algorithm, In *Proceedings of the Congress on Evolutionary Computation*, volume 3, Piscataway, New Jersey, Canberra, Australia, 2003.
- [4] C. Audet, J.E. Dennis, D.W. Moore and P.D. Frank A. Booker, A surrogate-model-based method for constrained optimization. In *8th Proceedings of the AIAA/NASA/USAF/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, 2000.
- [5] B.J. Bichon, M.S. Eldred, L.P. Siler, S. Mahadevan and J.M. McFarland, Multimodal reliability assessment for complex engineering applications using efficient global optimization, In *48th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials*, Hawaii, 2007.
- [6] A.J. Booker, J.E. Dennis, P.D. Frank, D.B. Serafini, V. Torczon and M.W. Trosset, A rigorous framework for optimization of expensive functions by surrogates, *Structural Optimization* **17**(1) (1999), 1–13.
- [7] M.R. Chernick, *Bootstrap Methods – A Guide for Practitioners and Researchers*. John Wiley & Sons, 2008.
- [8] C.A. Coello Coello, Treating constraints as objectives for single-objective evolutionary optimization, *Engineering Optimization* **32** (2000), 275–308.
- [9] D.W. Corne, K. Deb, P.J. Fleming and J.D. Knowles, The good of the many outweighs the good of the one: Evolutionary multi-objective optimization, *coNNectionS* **1** (2003), 9–13.
- [10] J. Cousin and M. Metcalfe, The british aerospace ltd. transport aircraft synthesis and optimization program, In *Systems and Operations Conference*, Dayton OH, September 1990. In AHS and ASEE.
- [11] K. Deb, *Multi-Objective Optimization using Evolutionary Algorithms*. John Wiley & Sons, 2001.

- [12] K. Deb, S. Agrawal, A. Pratap and T. Meyarivan, A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: Nsga-ii, *IEEE Transactions on Evolutionary Computation* **6**(2) (April 2002), 182–197.
- [13] R. Fletcher and S. Leyffer, Nonlinear programming without a penalty function, *Mathematical Programming* **91** (2002), 239–269.
- [14] A.I.J. Forrester and A.J. Keane, Recent advances in surrogate-based optimization, *Progress in Aerospace Sciences* **45** (2009), 50–79.
- [15] A.I.J. Forrester, A.J. Keane and N.W. Bressloff, Design and analysis of “noisy” computer experiments, *AIAA Journal* **44** (2006), 2331–2339.
- [16] A.I.J. Forrester, A. Sóbester and A.J. Keane, *Engineering Design via Surrogate Modelling*, John Wiley and Sons, 2008.
- [17] D. Ginsbourger, R. Le Riche and L. Carraro, *Computational Intelligence in Expensive Optimization Problems*, chapter Kriging is Well-Suited to Parallelize Optimization, Springer, 2010, pp. 131–162.
- [18] J.H. Holland, Outline for a logical theory of adaptive systems, *Journal of the ACM* **3** (1962), 297–314.
- [19] M.J. Jensen, *Guiding Single-Objective Optimization using Multi-Objective Methods*, volume 2611, chapter Applications of Evolutionary Computing, Springer, 2003, pp. 199–210.
- [20] Y. Jin, A comprehensive survey of fitness approximation in evolutionary computation, *Soft Computing* **9** (2005), 3–12.
- [21] D.R. Jones, A taxonomy of global optimization methods based on response surfaces, *Journal of Global Optimization* **21** (2001), 345–383.
- [22] D.R. Jones, M. Schonlau and W.J. Welch, Efficient global optimization of expensive black-box functions, *Journal of Global Optimization* **13** (1998), 455–492.
- [23] V. Khare, X. Yao and K. Deb, Performance scaling of multi-objective evolutionary algorithms, In *2nd International Conference on Evolutionary Multi-Criterion Optimization*, 2003.
- [24] J.D. Knowles, R.A. Watson and D.W. Corne, Reducing local optima in single-objective problems by multi-objectivization, In *Proceedings of the Evolutionary Multi-Criterion Optimization Conference*, Springer-Verlag, 2001, pp. 269–283.
- [25] M.D. Mackay, R.J. Beckman and W.J. Conover, A comparison of three methods for selecting values of input variables in the analysis of output from a computer code, *Technometrics* **21** (1979), 239–245.
- [26] M.D. Morris and T.J. Mitchell, Exploratory designs for computational experiments, *Journal of Statistical Planning and Inference* **43** (1995), 381–402.
- [27] F. Neumann and I. Wegener, *Multiobjective Problem Solving from Nature*, chapter Can Single-Objective Optimization Profit from Multiobjective Optimization? Springer-Verlag, 2008, pp. 115–130.
- [28] H. Nowacki, *Computer Science Computer Aided Design Modelling, Systems Engineering, CAD-Systems*, chapter Modelling of Design Decisions for CAD, Springer-Verlag, 1980, pp. 177–223.
- [29] A. Oyama, K. Shimoyama and K. Fujii, New constraint-handling method for multi-objective multi-constraint evolutionary optimization and its application to space plane design, In *Evolutionary and Deterministic Methods for Design, Optimization and Control with Applications to Industrial and Societal Problems*, Munich, 2005.
- [30] J.M. Parr, C.M.E. Holden, A.I.J. Forrester and A.J. Keane, Review of efficient surrogate infill sampling criteria with constraint handling, In *2nd International Conference on Engineering Optimization*, Lisbon, September 2010.
- [31] J.M. Parr, A.J. Keane, A.I.J. Forrester and C.M.E. Holden, Infill sampling criteria for surrogate-based optimization with constraint handling, *Engineering Optimization*, iFirst.
- [32] V. Picheny, D. Ginsbourger, O. Roustant, R.T. Haftka and N. Kim, Adaptive designs of experiments for accurate approximation of a target region, *Journal of Mechanical Design* **132** (2010).
- [33] K. Praditwong and X. Yao, How well do multi-objective evolutionary algorithms scale to large problems, In *IEEE Congress on Evolutionary Computation*, 2007.
- [34] P. Ranjan and D. Bingham, Sequential experiment design for contour estimation from complex computer codes, *Technometrics* **50** (2008), 527–541.
- [35] S.S. Rao, *Engineering Optimization: Theory and Practice*, Wiley-Interscience, third edition, 1996.
- [36] T. Ray, T. Kang and S.K. Chye, An evolutionary algorithm for constrained optimization, In *Proceedings of the Genetic and Evolutionary Computing Conference*, Las Vegas, 2000.
- [37] R.G. Regis, Stochastic radial basis function algorithms for large-scale optimization involving expensive black-box objective and constraint functions, *Computers and Operations Research* **38** (2011), 837–853.
- [38] R.G. Regis and C.A. Shoemaker, Constrained global optimization of expensive black box functions using radial basis functions, *Journal of Global Optimization* **31** (2005), 151–171.
- [39] J. Sacks, W.J. Welch, T.J. Mitchell and H.P. Wynn, Design and analysis of computer experiments, *Statistical Science* **4** (1989), 409–435.
- [40] M.J. Sasena, *Flexibility and Efficiency Enhancements for Constrained Global Design Optimization with Kriging*, PhD thesis, University of Michigan, 2002.
- [41] M.J. Sasena, P. Papalambros and P. Goovaerts, Exploration of metamodeling sample criteria for constrained global optimization, *Engineering Optimization* **34** (2002), 263–278.
- [42] M. Schonlau, *Computer Experiments and Global Optimization*, PhD thesis, University of Waterloo, 1997.

- [43] O. Schütze, A. Lara and C.A. Coello Coello, On the influence of the number of objectives on the hardness of a multiobjective optimization problem, *IEEE Transactions on Evolutionary Computation*, 2010.
- [44] A. Sóbester, S.J. Leary and A.J. Keane, On the design of optimization strategies based on global response surface approximation models, *Journal of Global Optimization* **33** (2005), 31–59.
- [45] N. Srinivas and K. Deb, Multiobjective optimization using nondominated sorting in genetic algorithms, *Journal of Evolutionary Computation* **2**(3) (1995), 221–248.
- [46] D.J.J. Toal, N.W. Bressloff and A.J. Keane, Kriging hyperparameter tuning strategies, *AIAA Journal* **46** (2008), 1240–1252.
- [47] A. Waldock and D. Corne, Multiple objective optimization applied to route planning, In *Proceedings of the 13th annual conference on Genetic and evolutionary computation*, Dublin, July 2011.
- [48] D. Yuret and M. Maza, Dynamic hill climbing: Overcoming the limitations of optimization techniques, In *Proceedings of the 2<sup>nd</sup> Turkish Symposium on Artificial Intelligence and ANN*, 1993.

AUTHOR COPY