Struct Multidisc Optim (2009) 39:439–457
DOI 10.1007/s00158-008-0338-0

RESEARCH PAPER

Multiple surrogates: how cross-validation errors can help us to obtain the best predictor

Felipe A. C. Viana · Raphael T. Haftka · Valder Steffen Jr.

Received: 26 May 2008 / Revised: 5 September 2008 / Accepted: 2 November 2008 / Published online: 8 January 2009

Abstract Surrogate models are commonly used to replace expensive simulations of engineering problems. Frequently, a single surrogate is chosen based on past experience. This approach has generated a collection of papers comparing the performance of individual surrogates. Previous work has also shown that fitting multiple surrogates and picking one based on cross-validation errors (PRESS in particular) is a good strategy, and that cross-validation errors may also be used to create a weighted surrogate. In this paper, we discussed how PRESS (obtained either from the leave-one-out or from the *k*-fold strategies) is employed to estimate the RMS error, and whether to use the best PRESS solution or a weighted surrogate when a single surrogate is needed. We also studied the minimization of the integrated square error as a way to compute the weights of the weighted average surrogate. We found that it pays to generate a large set of different surrogates and then use PRESS as a criterion for selection. We found that (1) in general, PRESS is good for filtering out inaccurate surrogates; and (2) with sufficient number of points, PRESS may identify the best surrogate of the set. Hence the use of cross-validation errors for choosing a surrogate and for calculating the weights of weighted surrogates becomes more attractive in high dimensions (when a large number of points is naturally required). However, it appears that the potential gains from using weighted surrogates diminish substantially in high dimensions. We also examined the utility of using all the surrogates for forming the weighted surrogates versus using a subset of the most accurate ones. This decision is shown to depend on the weighting scheme. Finally, we also found that PRESS as obtained through the *k*-fold strategy successfully estimates the RMSE.

Keywords Multiple surrogate models · Weighted average surrogates · Cross-validation errors · Prediction sum of squares

F. A. C. Viana (✉) · R. T. Haftka
Department of Mechanical and Aerospace Engineering,
University of Florida, Gainesville, FL 32611, USA
e-mail: fchegury@ufl.edu

R. T. Haftka
e-mail: haftka@ufl.edu

V. Steffen Jr.
School of Mechanical Engineering,
Federal University of Uberlandia,
Uberlandia, MG 38400, Brazil
e-mail: vsteffen@mecanica.ufu.br

1 Introduction

Despite advances in computer throughput, the computational cost of complex high-fidelity engineering simulations often makes it impractical to rely exclusively on simulation for design optimization (Jin et al. 2001). In addition, these advances do not seem to affect the time for a state-of-the-art simulation, but instead to be used to add complexity to the modeling (Venkataraman and Haftka 2004). To reduce the computational cost, surrogate models, also known as meta-models, are often used in place of the actual simulation models. With advances in computer throughput, the cost of fitting a given surrogate drops in relation to the cost of simulations. Consequently more sophisticated and more expensive surrogates have become popular. Surrogates such as radial basis neural networks (Smith 1993; Cheng and Titterington 1994), kriging models (Sacks et al. 1989;

Lophaven et al. 2002) support vector regression (Smola and Scholkopf 2004; Clarke et al. 2005) that require optimization in the fitting process, increasingly replace the traditional polynomial response surfaces (Box et al. 1978; Myers and Montgomery 1995) that only require the solution of a system of linear equations.

It is difficult to predict the performance of a surrogate for a new problem. In most of the cases, the practice is to use a surrogate model of preference based on past experience. Alternatively, the generation of large and diverse set of surrogates reduces the chances of using poorly fitted surrogates. Zerpa et al. (2005) used multiple surrogates for optimization of an alkaline–surfactant–polymer flooding processes incorporating a local weighted average model of the individual surrogates. Goel et al. (2007) explored different approaches in which the weights associated with each surrogate model are determined based on the global cross-validation error measure called prediction sum of squares (PRESS). Acar and Rais-Rohani (2008) discussed variations in the choice weights using cross-validation errors, and studied weight selection via optimization. PRESS can also be used to identify the surrogate most likely to be the most accurate. Lin et al. (2002) and Meckesheimer et al. (2002) presented an investigation on the use of cross-validation for RMSE estimation.

The objectives of this paper are: (1) investigate the effectiveness of PRESS for selecting the best surrogate in a diverse set; and (2) explore two weighting schemes based on minimization of the mean integrated square error and their potential to improve upon the best PRESS surrogate. The rest of the paper is organized as follows. Sections 2 and 3 present the methodology of the investigation. Section 4 defines the numerical experiments used for the investigation and Section 5 presents results and discussion. Finally, the paper is closed by recapitulating salient points and concluding remarks. Three appendices present (1) an overview of the used surrogate modeling techniques; (2) basic concepts about boxplots; and (3) how reducing the cost of PRESS computation in kriging affects the accuracy of the RMSE estimation.

2 Background

Surrogates are fitted to function values at p points, which are known as the design of experiments (DOE). The accuracy of the surrogates is then evaluated on the entire domain, with one of the main error metrics being the root mean square error (RMSE).

2.1 Root mean square error

We denote by $y(\mathbf{x})$ the actual simulation at the point $\mathbf{x} = \left[x_1 \; x_2 \; \ldots \; x_{n_{\text{dv}}} \right]^T$, and by $e(\mathbf{x}) = y(\mathbf{x}) - \hat{y}(\mathbf{x})$ the error associated with the prediction of the surrogate model, $\hat{y}(\mathbf{x})$. The actual root mean square error (RMSE) in the design domain with volume V is given by:

$$\text{RMSE}_{\text{actual}} = \sqrt{\frac{1}{V} \int_V e^2(\mathbf{x}) \, \mathrm{d}\mathbf{x}} \, . \tag{1}$$

In this paper, when we check the accuracy of a surrogate, we compute the RMSE by Monte Carlo integration at a large number of p_{test} test points:

$$\text{RMSE} = \sqrt{\frac{1}{p_{\text{test}}} \sum_{i=1}^{p_{\text{test}}} e_i^2} \, , \tag{2}$$

where $e_i = y_i - \hat{y}_i$ is the error associated with the prediction, $\hat{y}_i$, compared to the actual simulation, y_i, in the i-th test point.

We use five different Latin hypercube designs (Mckay et al. 1979), created by the MATLAB Latin hypercube function *lhsdesign*, set with the "maxmin" option with ten iterations. The RMSE is taken as the mean of the values for the five designs, and the uncertainty about this value is approximately standard deviation divided by $\sqrt{5}$.[1]

For comparing surrogates based on the data only at the p points of the design of experiments (DOE), we use cross-validation errors.

2.2 Cross-validation errors

A cross-validation error is the error at a data point when the surrogate is fitted to a subset of the data points not including that point. When the surrogate is fitted to all the other $p - 1$ points, (so-called leave-one-out strategy), we obtain the vector of cross-validation errors, $\tilde{\mathbf{e}}$. This vector is also known as the PRESS vector (PRESS stands for prediction sum of squares). Figure 1 illustrates the cross-validation errors for a polynomial response surface and kriging surrogates.

The RMSE is estimated from the PRESS vector:

$$\text{PRESS}_{\text{RMS}} = \sqrt{\frac{1}{p} \tilde{\mathbf{e}}^T \tilde{\mathbf{e}}}. \tag{3}$$

[1] We performed a comparison with the RMSE computation using the whole set of points. We found that the uncertainty can be also adequately estimated from the standard deviation of the entire set divided by the square root of the number of points.

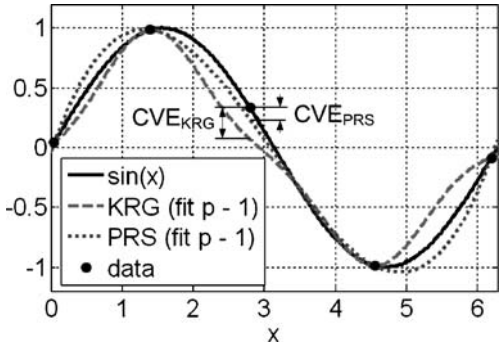


Fig. 1 The cross-validation error (*CVE*) at the third point of the DOE is exemplified by fitting a polynomial response surface (*PRS*) and kriging model (*KRG*) to five data points of the function sin(**x**). After repeating this process for all p points, the square root of the mean square error is used as an estimator of the *RMSE*

The leave-one-out strategy is computationally expensive for large number of points. We then use a variation of the k-fold strategy (Kohavi 1995) to overcome this problem. According to the classical k-fold strategy, after dividing the available data (p points) into p/k clusters, each fold is constructed using a point randomly selected (without replacement) from each of the clusters. Of the k folds, a single fold is retained as the validation data for testing the model, and the remaining $k-1$ folds are used as training data. The cross-validation process is then repeated k times with each of the k folds used exactly once as validation data. Note that k-fold turns out to be the leave-one-out when $k = p$. We implemented the strategy by: (1) extracting p/k points of the set using a "maximin" criterion (maximization of the minimum inter-distance); (2) removing these points from the set and repeating step (1) with the remaining points. Each set of extracted points is used for validation and the remaining for fitting. This process is repeated k times, each of it with a different validation set. The k-fold strategy is that it can be applied to any surrogate technique. For KRG models, less computationally expensive way of computing PRESS is to freeze the correlation parameters to their value for the original surrogate. We have studied this approach and the results (shown in Appendix 3) show that there is substantial loss of correlation between $\text{PRESS}_{\text{RMS}}$ and RMSE.

3 Ensemble of surrogates

3.1 Selection based on PRESS

Since $\text{PRESS}_{\text{RMS}}$ is an estimator of the RMSE, one possible way of using multiple surrogates is to select the model with best (i.e., smallest) PRESS value (BestPRESS surrogate). Because the quality of fit depends on the data points, the BestPRESS surrogate may vary from DOE to DOE. This strategy may include surrogates based on the same methodology, such as different instances of kriging (e.g., kriging models with different regression and/or correlation functions). The main benefit from a diverse and large set is the increasing chance of avoiding (1) poorly fitted surrogates and (2) DOE dependence of the performance of individual surrogates. Obviously, the success when using BestPRESS relies on the diversity of the set of surrogates and on the quality of the PRESS estimator.

3.2 Weighted average surrogate

Alternatively, a weighted average surrogate (WAS) intends to take advantage of n surrogates in the hope of canceling errors in prediction through proper weighting selection in the linear combination of the models:

$$\hat{y}_{\text{WAS}}(\mathbf{x}) = \sum_{i=1}^{n} w_i(\mathbf{x})\,\hat{y}_i(\mathbf{x}) = \mathbf{w}^T(\mathrm{x})\,\hat{\mathrm{y}}(\mathbf{x}) \;, \tag{4}$$

$$\sum_{i=1}^{n} w_i(\mathbf{x}) = \mathbf{1}^T\mathbf{w}(\mathbf{x}) = 1 \;, \tag{5}$$

where $\hat{y}_{\text{WAS}}(\mathbf{x})$ is the predicted response by the WAS model, $w_i(\mathbf{x})$ is the weight associated with the ith surrogate at $\mathbf{x}$, and $\hat{y}_i(\mathbf{x})$ is the predicted response by the ith surrogate. Furthermore, so that if all surrogates provide the same prediction, so would $\hat{y}_{\text{WAS}}(\mathbf{x})$.

The weights and the predicted responses can be written in the vector form as $\mathbf{w}(\mathbf{x})$ and $\hat{\mathbf{y}}(\mathbf{x})$. In this work we study weight selection based on PRESS, which is a global measure, so $\mathbf{w}(\mathbf{x}) = \mathbf{w}$, $\forall \mathbf{x}$.

In a specific DOE, when considering a set of surrogates, if not all are used, we assume surrogates are added to the ensemble one at a time based on the rank given by the $\text{PRESS}_{\text{RMS}}$. Then, the first one to be picked is the BestPRESS.

3.2.1 Heuristic computation of the weights

Goel et al. (2007) proposed a heuristic scheme for calculation of the weights, namely the PRESS weighted average surrogate (PWS). In PWS, the weights are computed as:

$$w_i = \frac{w_i^*}{\sum_{j=1}^{n} w_j^*} \;, \qquad w_i^* = \left(E_i + \alpha\, E_{\text{avg}}\right)^{\beta} \;,$$

$$E_{\text{avg}} = \frac{1}{n}\sum_{i=1}^{n} E_i \;, \qquad \beta < 0,\ \alpha < 1, \tag{6}$$

where E_i is given by the $\text{PRESS}_{\text{RMS}}$ of the ith surrogate.

The two parameters α and β, control the importance of averaging and importance of individual PRESS, respectively. Goel et al. (2007) suggested $\alpha = 0.05$ and $\beta = -1$.

3.2.2 Computation of the weights for minimum RMSE

Using an ensemble of neural networks, Bishop (1995) proposed a weighted average surrogate obtained by approximating the covariance between surrogates from residuals at training or test points. Here, as in Acar and Rais-Rohani (2008), we opt instead for basing Bishop's approach on minimizing the mean square error (MSE):

$$\mathrm{MSE}_{\mathrm{WAS}} = \frac{1}{V}\int_V e^2_{\mathrm{WAS}}(\mathbf{x})\, d\mathbf{x} = \mathbf{w}^T\mathbf{C}\mathbf{w}\,, \tag{7}$$

where $e_{\mathrm{WAS}}(\mathbf{x}) = y(\mathbf{x}) - \hat{y}_{\mathrm{WAS}}(\mathbf{x})$ is the error associated with the prediction of the WAS model, and the integral, taken over the domain of interest, permits the calculation of the elements of $\mathbf{C}$ as:

$$c_{ij} = \frac{1}{V}\int_V e_i(\mathbf{x})\, e_j(\mathbf{x})\, d\mathbf{x}\,, \tag{8}$$

where $e_i(\mathbf{x})$ and $e_j(\mathbf{x})$ are the errors associated with the prediction given by the surrogate model i and j, respectively.

$\mathbf{C}$ plays the same role as the covariance matrix in Bishop's formulation. However, we approximate $\mathbf{C}$ by using the vectors of cross-validation errors, $\tilde{\mathbf{e}}$,

$$c_{ij} \simeq \frac{1}{p}\tilde{\mathbf{e}}_i^T\tilde{\mathbf{e}}_j\,, \tag{9}$$

where p is the number of data points and the subindexes i and j indicate different surrogate models.

Given the $\mathbf{C}$ matrix, the optimal weighted surrogate (OWS) is obtained from minimization of the *MSE* as:

$$\min_{\mathbf{w}} \mathrm{MSE}_{\mathrm{WAS}} = \mathbf{w}^T\mathbf{C}\mathbf{w}\,, \tag{10}$$

subject to:

$$\mathbf{1}^T\mathbf{w} = 1\,. \tag{11}$$

The solution is obtained using Lagrange multipliers, as:

$$\mathbf{w} = \frac{\mathbf{C}^{-1}\mathbf{1}}{\mathbf{1}^T\mathbf{C}^{-1}\mathbf{1}}\,. \tag{12}$$

The solution may include negative weights as well as weights larger than one. Allowing this freedom was found to amplify errors coming from the approximation of $\mathbf{C}$ matrix (9). One way to enforce positivity is to solve (12) using only the diagonal elements of $\mathbf{C}$, which are more accurately approximated than the off-diagonal terms. We denote this approach $\mathrm{OWS}_{\mathrm{diag}}$. It is worth observing that when $\alpha = 0$ and $\beta = -2$, PWS gives the same weights of $\mathrm{OWS}_{\mathrm{diag}}$. We also studied the possibility of adding the constraint $w_i \geq 0$ to the optimization problem; however it was not sufficient to overcome the effect of poor approximations of the $\mathbf{C}$ matrix. OWS is recommended for the cases when an accurate approximation of $\mathbf{C}$ is available and $\mathrm{OWS}_{\mathrm{diag}}$ for a less accurate one.

Either when employing BestPRESS or one of the above mentioned WAS schemes, the computational cost of using an ensemble of surrogates depends on the calculation of the cross-validation errors. Since each surrogate of the ensemble is fit as many times as the number of points in the data set, the larger the set the higher the cost.

3.2.3 Should we use all surrogates?

When forming a weighted surrogate, we may use all m surrogates we have created; or alternatively, we may use just a subset of the n best ones ($n \leq m$). With an exact $\mathbf{C}$ matrix, there is no reason not to use them all. However, with the approximation of (9), it is possible that adding inaccurate surrogates will lead to loss of accuracy. When we use a partial set, we will add the surrogates according to the rank given by $\mathrm{PRESS}_{\mathrm{RMS}}$. BestPRESS (surrogate with lowest $\mathrm{PRESS}_{\mathrm{RMS}}$) is the first one to be picked; which corresponds to $n = 1$. When the worst $\mathrm{PRESS}_{\mathrm{RMS}}$-ranked surrogate is added to the ensemble, $n = m$.

This can be shown to be the best approach with an exact over diagonal $\mathbf{C}$ matrix.

Note that when $n < m$ (i.e., not all surrogates are used) it is expected that the set of surrogates in the ensemble can change with the DOE, since the performance of individual surrogates may vary from one DOE to another.

3.2.4 Combining accuracy and diversity

For both selection and combination, the best case scenario would be to have a set of surrogates that are different in terms of prediction values $\left(\hat{y}(\mathbf{x})\right)$ but similar in terms of prediction accuracy (RMSE). This would increase the chance that *WAS* would allow error cancellation. In the present work, even though we generated a substantial number of surrogates with different statistical models, loss functions, and shape functions, we usually missed this goal. That is, comparably accurate surrogates were often highly correlated. Future

Table 1 Information about the set of 24 basic surrogates (see Appendix 1 for a review on the surrogate models)

Surrogates	Modeling technique	Details
1 KRG-Poly0-Exp	Kriging model	Poly0, Poly1, and Poly2 indicate zero, first, and second order polynomial regression model, respectively. Exp and Gauss indicate general exponential and Gaussian correlation model, respectively. In all cases, $\theta_0 = 10 \times \mathbf{1}_{ndv \times 1}$, and $10^{-2} \leq \theta_i \leq 200$, $i = 1, 2, \ldots, ndv$.
2 KRG-Poly0-Gauss		We chose 6 different Kriging surrogates by varying the regression and correlation models.
3 KRG-Poly1-Exp		
4 KRG-Poly1-Gauss		
5 KRG-Poly2-Exp		
6 KRG-Poly2-Gauss		
7 PRS2	Polynomial response surface	Full model of degree 2.
8 RBNN	Radial basis neural network	Goal $= (0.05\overline{y})^2$ and Spread $= 1/3$.
9 SVR-ANOVA-E-Full	Support vector regression	ANOVA, ERBF, GRBF and Spline indicate the kernel function (ERBF and GRBF kernel functions were set with $\sigma = 0.5$). E and Q indicate the loss function as ε-insensitive and quadratic, respectively. Full and Short refer to different values for the regularization parameter, C, and for the insensitivity, ε. Full adopts $C = \infty$ and $\varepsilon = 1 \times 10^{-4}$, while Short01 and Short02 uses the selection of values according to Cherkassky and Ma (2004). For Short01 $\varepsilon = {\sigma_y}/{\sqrt{k}}$ and for Short02 $\varepsilon = 3\sigma_y \sqrt{{\ln(k)}/{k}}$; and for both $C = 100 \max\left(\left|\overline{y} + 3\sigma_y\right|, \left|\overline{y} - 3\sigma_y\right|\right)$, where $\overline{y}$ and σ_y are the mean value and the standard deviation of the function values at the design data, respectively. We chose 16 different SVR surrogates by varying the kernel function, the loss function (ε-insensitive or quadratic) and the SVR parameters (C and ε) define these surrogates.
10 SVR-ANOVA-E-Short01		
11 SVR-ANOVA-E-Short02		
12 SVR-ANOVA-Q		
13 SVR-ERBF-E-Full		
14 SVR-ERBF-E-Short01		
15 SVR-ERBF-E-Short02		
16 SVR-ERBF-Q		
17 SVR-GRBF-E-Full		
18 SVR-GRBF-E-Short01		
19 SVR-GRBF-E-Short02		
20 SVR-GRBF-Q		
21 SVR-Spline-E-Full		
22 SVR-Spline-E-Short01		
23 SVR-Spline-E-Short02		
24 SVR-Spline-Q		

For support vector regression, details about the kernels are given in Table 8

Table 2 Selection of surrogates according to different criteria

Surrogate	Definition
BestRMSE	Most accurate surrogate for a given DOE (basis of comparison for all other surrogates).
BestPRESS	Surrogate with lowest $\text{PRESS}_{\text{RMS}}$ for a given DOE.
PWS	Weighted surrogate with heuristic computation of weights.
$\text{OWS}_{\text{ideal}}$	Most accurate weighted surrogate based on the true **C** matrix. Depending on the surrogate selection, $\text{OWS}_{\text{ideal}}$ may be less accurate than BestRMSE.
OWS	Weighted surrogate based on an approximation of the **C** matrix using cross-validation errors.
OWS_{diag}	Weighted surrogate based on the main diagonal elements of the approximated **C** matrix (using cross-validation errors).

BestRMSE and $\text{OWS}_{\text{ideal}}$ are defined based on testing points; all others are obtained using data points

research may need to consider the problem of how to generate a better set of basic surrogates.

4 Numerical experiments

4.1 Basic surrogates and derived surrogates

Table 1 gives details about the 24 different basic surrogates used during the investigation (see Appendix 1 for a short theoretical review). The toolbox of Lophaven et al. (2002), the native neural networks Matlab toolbox (Mathworks contributors 2004), and the code developed by Gunn (1997) were used to execute the KRG, RBNN, and SVR algorithms, respectively. The SURROGATES toolbox of Viana and Goel (2007) is used to execute the PRS and WAS algorithms and also for an easy manipulation of all these different codes.

No attempt was made to improve the predictions of any surrogate by fine tuning their respective parameters (such as the initial θ parameter in kriging models).

Table 2 summarizes the surrogates that can be selected from the set using different criteria. These include the best choice of surrogate that would be selected if we had perfect knowledge of the function (BestRMSE), the surrogate selected based on the lowest PRESS error (BestPRESS), as well as the various weighted surrogates. Of these, the ideal weighted surrogate has weights based on perfect knowledge, and it provides a bound to the gains possible by using WAS.

4.2 Performance measures

The numerical experiments are intended to (1) measure how efficient is $\text{PRESS}_{\text{RMS}}$ as an estimator of the RMSE (and consequently, how good is $\text{PRESS}_{\text{RMS}}$ for identifying the surrogate with the smallest RMSE), and (2) explore how much the RMSE can be further reduced by the WAS. The first objective is quantified by comparing the correlation between $\text{PRESS}_{\text{RMS}}$ and RMSE across the 24 surrogates. For both objectives, we compare each basic surrogate, BestPRESS, and WAS model with the best surrogate of the set in a specific DOE (BestRMSE). We define %difference such as the percent gain by choosing a specific model over BestRMSE:

$$\%\text{difference} = 100\frac{\text{RMSE}_{\text{BestRMSE}} - \text{RMSE}_{\text{Surr}}}{\text{RMSE}_{\text{BestRMSE}}}. \quad (13)$$

where $\text{RMSE}_{\text{BestRMSE}}$ is the RMSE of the best surrogate of that specific DOE (i.e., BestRMSE) and $\text{RMSE}_{\text{Surr}}$ is the RMSE of the surrogate we are interested in (it can be either a single or a WAS model). When %difference > 0 there is a gain in using the specific surrogate, and when %difference < 0 there is a loss.

For each basic surrogate and also for BestPRESS, it is expected that %difference ≤ 0, which means that there may be losses and the best case scenario is when one of the basic surrogates (hopefully BestPRESS) coincides with BestRMSE. When considering BestPRESS, the smaller the loss the better is the ability of $\text{PRESS}_{\text{RMS}}$ to select the best surrogate of the set. For the WAS, in a particular DOE, we add surrogates according to the rank given by the $\text{PRESS}_{\text{RMS}}$ value (i.e., we always start from BestPRESS). Thus, the %difference may start negative, and as we increase the number of surrogates

Table 3 Parameters used in Hartman function

$$\text{Hartman3 } \mathbf{B} = \begin{bmatrix} 3.0 & 10.0 & 30.0 \\ 0.1 & 10.0 & 35.0 \\ 3.0 & 10.0 & 30.0 \\ 0.1 & 10.0 & 35.0 \end{bmatrix}$$

$$\mathbf{D} = \begin{bmatrix} 0.3689 & 0.1170 & 0.2673 \\ 0.4699 & 0.4387 & 0.7470 \\ 0.1091 & 0.8732 & 0.5547 \\ 0.03815 & 0.5743 & 0.8828 \end{bmatrix}$$

$$\text{Hartman6 } \mathbf{B} = \begin{bmatrix} 10.0 & 3.0 & 17.0 & 3.5 & 1.7 & 8.0 \\ 0.05 & 10.0 & 17.0 & 0.1 & 8.0 & 14.0 \\ 3.0 & 3.5 & 1.7 & 10.0 & 17.0 & 8.0 \\ 17.0 & 8.0 & 0.05 & 10.0 & 0.1 & 14.0 \end{bmatrix}$$

$$\mathbf{D} = \begin{bmatrix} 0.1312 & 0.1696 & 0.5569 & 0.0124 & 0.8283 & 0.5886 \\ 0.2329 & 0.4135 & 0.8307 & 0.3736 & 0.1004 & 0.9991 \\ 0.2348 & 0.1451 & 0.3522 & 0.2883 & 0.3047 & 0.6650 \\ 0.4047 & 0.8828 & 0.8732 & 0.5743 & 0.1091 & 0.0381 \end{bmatrix}$$

Table 4 Specifications for the DOEs, and test points for the test functions

Test problem	No. of design variables	No. of points for fitting	No. of points for test (in each of the 5 DOEs)
Branin–Hoo	2	12, 20, and 42	2,000
Camelback	2	12	2,000
Hartman3	3	20	2,000
Hartman6	6	56	2,000
Extended Rosenbrock	9	110 and 220	2,500
Dixon–Price	12	182	4,000

in the ensemble, it is expected that %difference turns to positive, which express the potential of WAS to be better than the best surrogate of the set.

4.3 Test functions

To test the effectiveness of the various approaches, we employ a set of analytical functions widely used as benchmark problems in optimization (e.g. Dixon and Szegö 1978; Lee 2007). These are:

1. Branin–Hoo function (two variables):

$$y(\mathbf{x}) = \left(x_2 - \frac{5.1x_1^2}{4\pi^2} + \frac{5x_1}{\pi} - 6\right)^2 + 10\left(1 - \frac{1}{8\pi}\right)\cos(x_1) + 10\,, \quad -5 \le x_1 \le 10, \quad 0 \le x_2 \le 15\,. \tag{14}$$

2. Camelback function (two variables):

$$y(\mathbf{x}) = \left(\frac{x_1^4}{3} - 2.1x_1^2 + 4\right)x_1^2 + x_1x_2 + \left(4x_2^2 - 4\right)x_2^2\,, \quad -3 \le x_1 \le 3, \quad -2 \le x_2 \le 2\,. \tag{15}$$

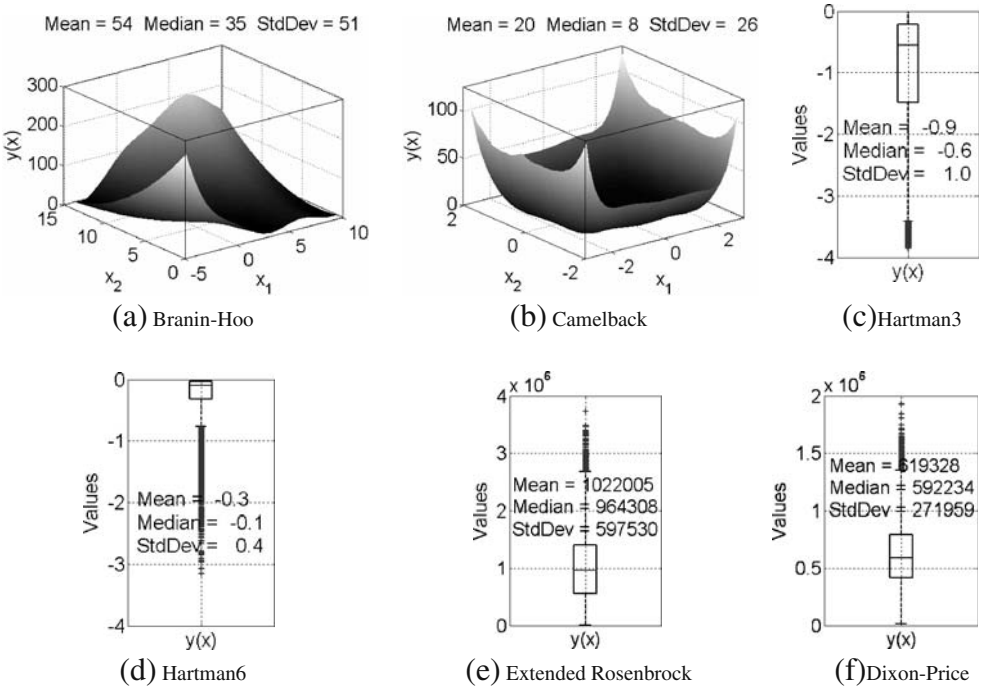
3. Hartman functions (three and six variables):

$$y(\mathbf{x}) = -\sum_{i=1}^{q} a_i \exp\left(-\sum_{j=1}^{m} b_{ij}\left(x_j - d_{ij}\right)^2\right), \quad 0 \le x_j \le 1\,, \quad j = 1,\ 2,\ \ldots,\ m. \tag{16}$$

We use two instances: Hartman3 ($m = 3$) and Hartman6 ($m = 6$), with 3 and with 6 variables, respectively. For both $q = 4$ and $\mathbf{a} = \left[1.0\ 1.2\ 3.0\ 3.2\right]$. Other parameters are given in Table 3.

4. Extended Rosenbrock function (nine variables):

$$y(\mathbf{x}) = \sum_{i=1}^{m-1}\left[(1 - x_i)^2 + 100\left(x_{i+1} - x_i^2\right)^2\right], \quad -5 \le x_i \le 10\,, \quad i = 1,\ 2,\ \ldots,\ m = 9. \tag{17}$$



(a) Branin-Hoo (b) Camelback (c) Hartman3

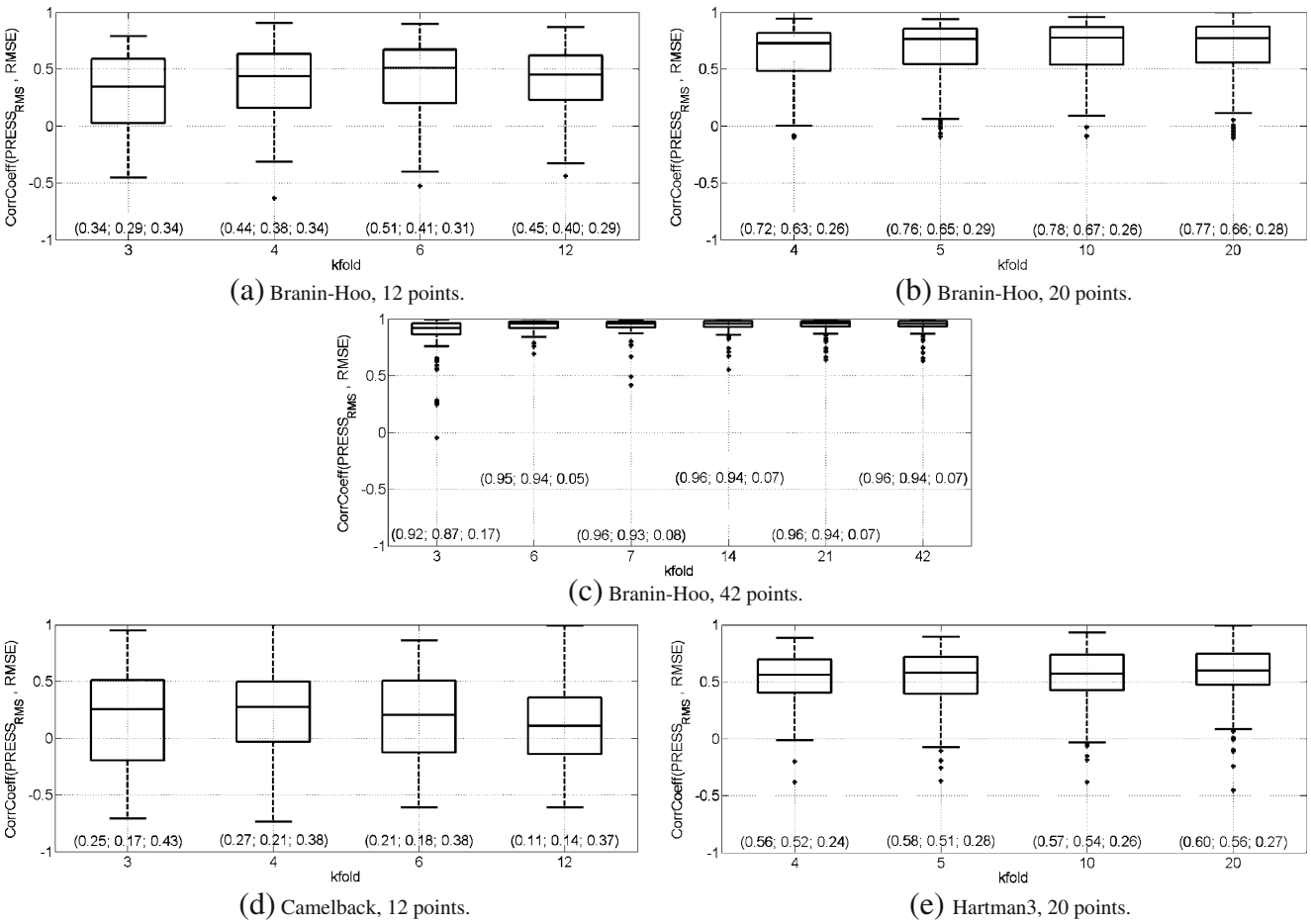
(d) Hartman6 (e) Extended Rosenbrock (f) Dixon-Price

Fig. 2 Plot of test functions. See Appendix 2 for details about boxplots

5. Dixon–Price function (12 variables):

$$y(\mathbf{x}) = (x_1 - 1)^2 + \sum_{i=2}^{m} i\left[2x_i^2 - x_{i-1}\right]^2,$$
$$-10 \le x_i \le 10, \quad i = 1, 2, \ldots, m = 12. \tag{18}$$

As stated before, the quality of fit depends on the training data (function values at DOE). As a consequence, performance measures may vary from DOE to DOE. Thus, for all test problems, a set of 100 different DOEs were used as a way of averaging out the DOE dependence of the results. They were created by the MATLAB Latin hypercube function *lhsdesign*, set with the "maxmin" option with 1,000 iterations. Table 4 shows details about the data set generated for each test function. Naturally, the number of points used to fit surrogates increase with dimensionality. We also use the Branin–Hoo and the extended Rosenbrock functions to investigate what happens in low-and high-dimensions, respectively, if we can afford more points. For the computation of the cross-validation errors, in most cases we use the leave-one-out strategy ($k = p$ in the k-fold strategy, see Section 2). However, for the Dixon–Price function, due to the high cost of the leave-one-out strategy for the 24 surrogates for all 100 DOEs; we adopted the k-fold strategy with $k = 14$, instead. This means that the surrogate is fitted 14 times, each time with 13 points left out (that is to the remaining 169 points). For all problems, we use five different Latin hypercube designs for evaluating the accuracy of the surrogate by Monte Carlo integration of the RMSE. These DOEs are also created by the MATLAB



(a) Branin-Hoo, 12 points.

(b) Branin-Hoo, 20 points.

(c) Branin-Hoo, 42 points.

(d) Camelback, 12 points.

(e) Hartman3, 20 points.

Fig. 3 Boxplot of the correlation coefficient between the vectors of $\text{PRESS}_{\text{RMS}}$ and RMSE (24 surrogates each) for the low-dimensional problems (this coefficient is computed for all 100 DOEs). In parenthesis, it is the median, mean, and standard deviation of the correlation coefficient, respectively. $\text{PRESS}_{\text{RMS}}$ is calculated via k-fold strategy (which reduces to the leave-one-out strategy when k is equal to the number of points used for fitting). It can be seen that except when we use the smallest value of k, there is no significant disadvantage comparing the k-fold and the leave-one-out strategies. See Appendix 2 for details about boxplots

Latin hypercube function *lhsdesign*, but set with the "maxmin" option with ten iterations. The RMSE is taken as the mean of the values for the five DOEs

Figure 2 illustrates the complexity of the problems. For the two-dimensional cases, plots using the test points reveal the presence of high gradients. For all other cases, the test points were used to obtain box plots of the functions, which show variation in the values of the functions by more than one order of magnitude.

5 Results and discussion

We begin with a short study on the use of the k-fold strategy. When the number of folds is small, the accuracy of the surrogate fit by omitting this fold can be much poorer than that of the original surrogate, so that $\text{PRESS}_{\text{RMS}}$ is likely to be much larger than RMSE. Since we mostly used a number of points which is twice that of the polynomial coefficients, two folds would fit a polynomial with the same number of points as the number of coefficients, which is likely to be much less accurate than using all the points. So in this study, we varied the values of k, starting from the smallest value of k that divides the p points into more than two folds For example, if $p = 12$, this value is $k = 3$ (which generates folds of four points each). Then we use all possible values of k up to p (when the k-fold turns into the leave-one-out strategy). Due to the computational cost of the PRESS errors, we divided the test problems into two sets:

1. Low-dimensional problems (Branin–Hoo, Camelback and Hartman3). We compute the correlations between RMSE and $\text{PRESS}_{\text{RMS}}$ for different k values. For a given DOE, the correlation is computed between the vectors of RMSE and $\text{PRESS}_{\text{RMS}}$ values for the different surrogates. The correlation measures the ability of PRESS to substitute for the exact RMSE for choosing the best surrogate (the closer the correlation is to 1 the better). This is repeated for all DOEs.
2. High-dimensional problems (Hartman6, extended Rosenbrock, and Dixon–Price). The cost of performing PRESS for several k values is high. To keep low computational cost, we do the study only for the least expensive surrogate, i.e., the PRS (degree = 2) and we calculate the ratio between $\text{PRESS}_{\text{RMS}}$ and RMSE for each DOE (the closer the ratio is to 1 the better).

Figure 3 shows that in low-dimensions the use of the k-fold does not drastically affect the correlation between RMSE and $\text{PRESS}_{\text{RMS}}$; even though the smallest value of k always presents the worst correlation coefficient between $\text{PRESS}_{\text{RMS}}$ and RMSE. It means that the

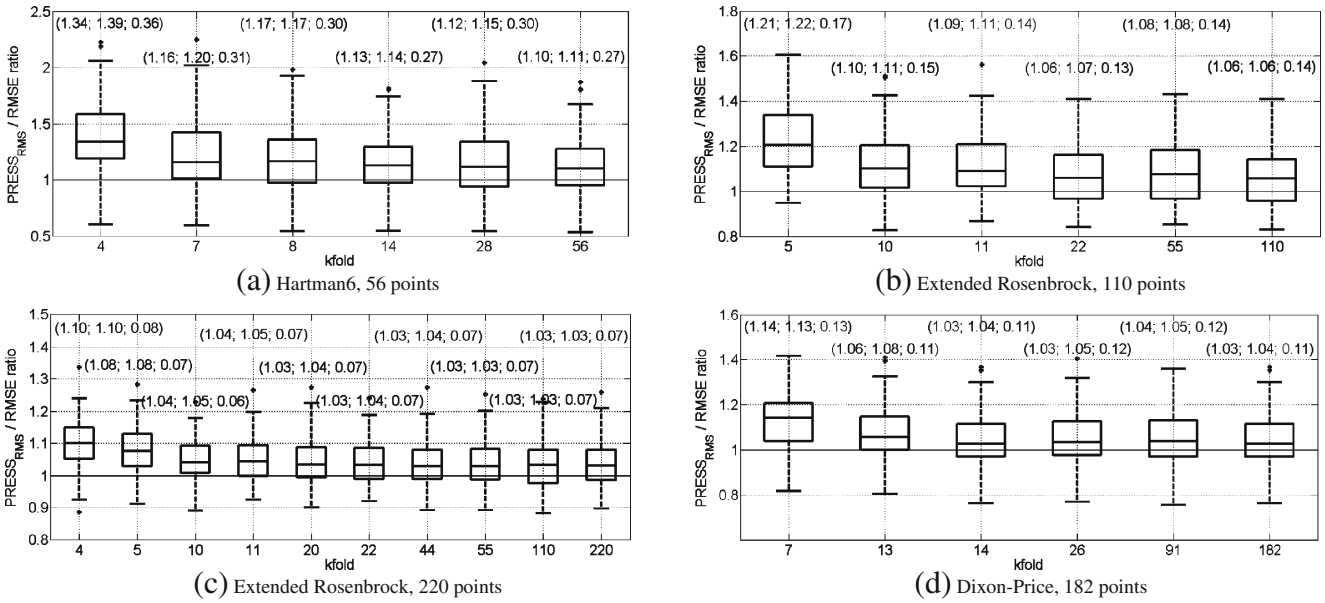


Fig. 4 Boxplot of the $\text{PRESS}_{\text{RMS}}$/RMSE ratio for PRS (degree = 2) for the low-dimensional problems (this ratio is computed for all 100 DOEs). $\text{PRESS}_{\text{RMS}}$ is calculated via k-fold strategy (which equals to the leave-one-out strategy when k is equal to the number of points used for fitting). In parenthesis, it is the median, mean, and standard deviation of the ratio, respectively. As expected, the ratio becomes better as k approaches the number of points. In these cases, it does not pay much to perform more than 30 fits for the cross-validations (i.e. $k > 30$). See Appendix 2 for details about boxplots

Table 5 Frequency, in number of DOEs (out of 100), of best RMSE and $PRESS_{RMS}$ for each basic surrogate

Surrogate	Branin–Hoo (12 points)		Branin–Hoo (20 points)		Branin–Hoo (42 points)		Camelback (12 points)		Hartman3 (20 points)		Hartman6 (56 points)		Extended Rosenbrock (110 points)		Extended Rosenbrock (220 points)		Dixon–Price (182 points)	
	RMSE	$PRESS_{RMS}$	RMSE	$PRESS_{RMS}$	RMSE	$PRESS_{RMS}$	RMSE	$PRESS_{RMS}$	RMSE	$PRESS_{RMS}$	RMSE	$PRESS_{RMS}$	RMSE	$PRESS_{RMS}$	RMSE	$PRESS_{RMS}$	RMSE	$PRESS_{RMS}$
1	0	0	0	0	0	0	5	**15**	4	2	1	3	0	0	1	0	0	0
2	1	1	**58**	**79**	**16**	**31**	1	6	**38**	**11**	7	2	0	0	0	0	0	0
4	3	0	**14**	**9**	**46**	**28**	0	0	5	2	0	0	0	0	1	0	0	0
5	0	1	0	0	0	0	**16**	3	0	0	0	0	**18**	**32**	**34**	**39**	**10**	**21**
6	2	2	**23**	4	**37**	**41**	**30**	6	1	1	0	0	**77**	**41**	**35**	**26**	**74**	**40**
7	0	1	0	0	0	0	0	**18**	0	0	0	0	**4**	**23**	0	1	**16**	**39**
8	1	3	0	0	0	0	12	3	**36**	**31**	**17**	6	0	0	0	0	0	0
9	**42**	**19**	0	**5**	1	0	7	**11**	0	4	0	0	0	0	0	0	0	0
10	0	3	0	0	0	0	0	6	0	2	0	0	0	0	0	0	0	0
11	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	5	0	0	0	0	2	11	0	1	0	1	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	**6**	8	1	3	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	5	0	1	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0
17	11	**32**	0	0	0	0	0	3	5	4	**14**	**18**	0	0	0	0	0	0
18	0	1	0	0	0	0	0	3	3	**13**	**50**	**53**	0	0	0	0	0	0
20	**16**	**18**	0	1	0	0	0	5	2	9	10	**9**	0	0	0	0	0	0
21	**24**	8	5	2	0	0	**26**	10	0	1	0	0	0	2	**21**	12	0	0
22	0	1	0	0	0	0	0	0	0	3	0	4	0	1	6	**14**	0	0
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	8	0	0
24	0	4	0	0	0	0	1	0	0	1	0	0	1	1	0	0	0	0
Top 3 surrogates	**82**	**69**	**95**	**93**	**99**	**100**	**72**	**44**	**80**	**55**	**81**	**80**	**99**	**96**	**90**	**79**	**100**	**100**

The numbers indicate the identity as in Table 1. It can be seen that (1) for both RMSE and $PRESS_{RMS}$, the best surrogate depends on the problem and also on the DOE; and (2) especially in the case of the high-dimensional problems; the best three surrogates according to both RMSE and $PRESS_{RMS}$ tend to be the same (bold face). As the number of points increases most surrogates fall behind, i.e. the advantage of the top surrogates is more pronounced

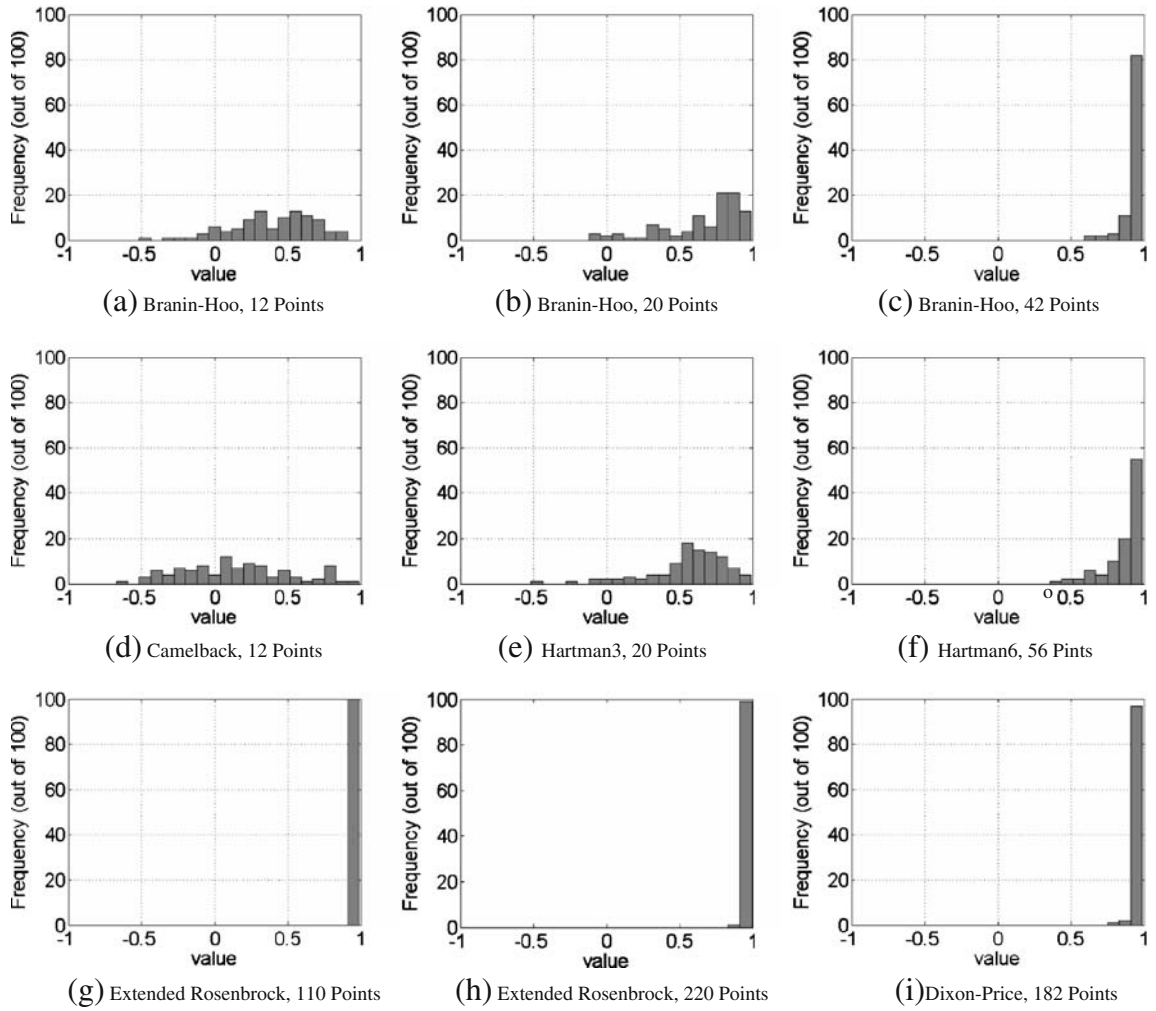


Fig. 5 Correlation between $\text{PRESS}_{\text{RMS}}$ and *RMSE*. In a given DOE, the correlation is computed between the sets of $\text{PRESS}_{\text{RMS}}$ values and RMSE. The correlation appears to improve with the number of points

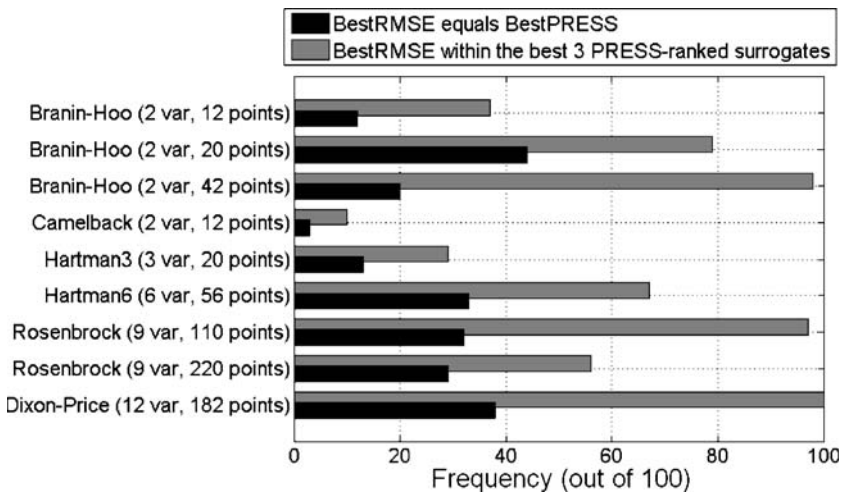


Fig. 6 Frequency of success in selecting BestRMSE (out of 100 experiments). The success of using PRESS for surrogates selection increases with the number of points

poor correlation has to do with the number of points used to fit the set of surrogates. This is clearly seen in the Branin–Hoo example, Fig. 3a–c, where more points improve the correlation. Figure 4 illustrates that in high-dimensions, an increasing of the value of k improves the quality of the information given by the cross-validation errors. The best scenario is when $k = p$. However, the ratio seems to be acceptable when the each fold has around 10% of the p points (i.e., $k = 8$ when $p = 56$; $k = 11$ when $p = 110$; and $k = 14$ when $p = 182$). This agrees with Meckesheimer et al. (2002).

Back to the discussion about $\text{PRESS}_{\text{RMS}}$ as a criterion for surrogate selection, Table 5 shows the frequency of best RMSE and the $\text{PRESS}_{\text{RMS}}$ for each of the surrogates in all test problems. It can be observed that the best surrogate depends (1) on the problem, i.e. no single surrogate or even modeling technique is always the best; and (2) on the DOE, i.e. for the same problem, the surrogate that performs the best can vary from DOE to DOE. In addition, as the number of points increases, there is a better agreement between RMSE and $\text{PRESS}_{\text{RMS}}$. Particularly, the top three surrogates (bold face in Table 5) are identified better. However, for the Branin–Hoo problem, we note deterioration in identifying the best surrogate when going from 20 to 42 points. This is because for high density of

Table 6 %difference in the RMSE, defined in (13), of the best three basic surrogates (according to how often they have the best $\text{PRESS}_{\text{RMS}}$, see Table 5) and *BestPRESS* for each test problem

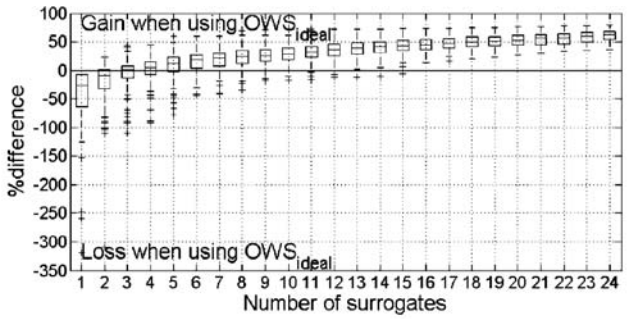
Problem	Surrogate	Freq. of best $\text{PRESS}_{\text{RMS}}$	Median	Mean	SD
Branin–Hoo (12 points)	9	19	−3	−21	36
	17	32	−18	−27	33
	20	18	−17	−19	18
	BestPRESS	–	−26	−43	55
Branin–Hoo (20 points)	2	79	0	−13	31
	4	9	−14	−29	55
	9	5	−152	−189	148
	BestPRESS	–	−3	−31	61
Branin–Hoo (42 points)	2	31	−12	−26	57
	4	28	−3	−21	56
	6	41	−21	−42	55
	BestPRESS	–	−18	−37	51
Camelback (12 points)	1	15	−40	−42	30
	7	18	−8	−12	13
	9	11	−24	−31	25
	BestPRESS	–	−29	−35	29
Hartman3 (20 points)	2	11	−12	−22	26
	8	31	−5	−33	206
	18	13	−23	−26	17
	BestPRESS	–	−21	−28	31
Hartman6 (56 points)	17	18	−7.9	−10.2	9.7
	18	53	−0.1	−1.7	2.6
	20	9	−3.5	−4.0	3.4
	BestPRESS	–	−1.9	−5.2	8.2
Extended Rosenbrock (110 points)	5	32	−0.09	−0.47	0.85
	6	41	0.00	−0.19	1.02
	7	23	−0.13	−0.52	0.94
	BestPRESS	–	−0.02	−1.03	4.15
Extended Rosenbrock (220 points)	5	39	−1.28	−3.55	7.85
	6	26	−2.10	−6.55	15.11
	22	14	−3.75	−9.19	17.54
	BestPRESS	–	−2.80	−7.17	16.49
Dixon–Price (182 points)	5	21	0.00	0.00	0.00
	6	40	0.00	−0.01	0.05
	7	39	0.00	0.00	0.00
	BestPRESS	–	0.00	0.00	0.03

For the basic surrogates, the numbers indicate the identity as in Table 1. The negative sign of %difference indicates a loss in terms of RMSE for the specific surrogate compared with the best surrogate. The loss decreases with increasing number of points. At low number of points, BestPRESS may not be even as good as the third best surrogate. In high dimension, BestPRESS is as good as the best of the three

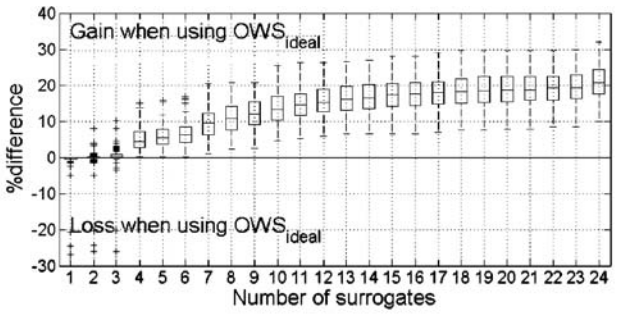
points, the trend for kriging becomes less important, so surrogates 2, 4, and 6 have very similar performance. In addition, for KRG surrogates in high dimensions, the correlation model is less important since points are so sparse, hence several almost identical surrogates.

Figure 5 shows histograms of the correlations between RMSE and $\text{PRESS}_{\text{RMS}}$. Figure 6 complements Fig. 5 and shows that as the number of points increases, there is a better agreement between the selections given by RMSE and $\text{PRESS}_{\text{RMS}}$. Particularly, the top three surrogates are identified better. For the extended Rosenbrock with 220 points, the number of surrogates that are almost equally accurate is larger than 3. This explains why the number of times that BestRMSE is within the best 3 PRESS-ranked surrogates drops when compared to the case with 110 points. Altogether, when there are few points (low-dimensional problems, i.e., two and three variables), $\text{PRESS}_{\text{RMS}}$ is good for filtering out bad surrogates; when there are more points (high-dimensional problems, i.e., six, nine and 12 variables) and $\text{PRESS}_{\text{RMS}}$ can also identify the subset of the best surrogates.

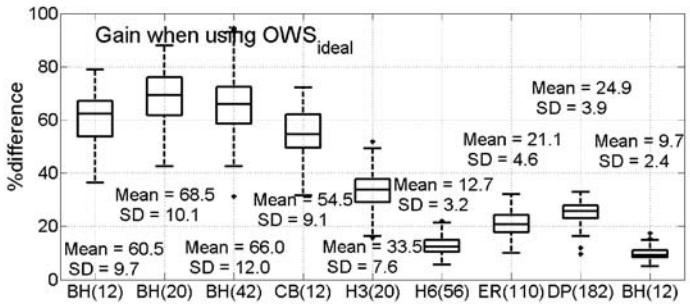
Table 6 provides the mean, median and standard deviation of the %difference in the RMSE for the best three surrogates and BestPRESS. Since a single fixed surrogate cannot beat the best surrogate of the set



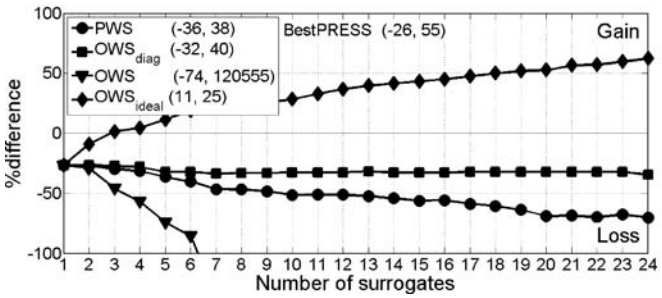
(a) OWS_{ideal} for Branin-Hoo, 12 Points



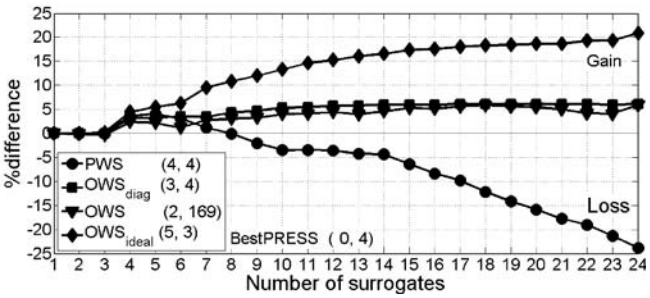
(b) OWS_{ideal} for Extended Rosenbrock, 110 Points



(c) OWS_{ideal} for all test problems (using all 24 surrogates)



(d) Median of the *%difference* of WAS for Branin-Hoo, 12 Points



(e) Median of the *%difference* of WAS for Extended Rosenbrock, 110 Points

Fig. 7 %difference in the RMSE, defined in (13), when using weighted average surrogates. **a**, **b** illustrates the effect of adding surrogates to the ensemble (picked one at a time according to the $\text{PRESS}_{\text{RMS}}$ ranking) for the Branin–Hoo and Extended Rosenbrock functions (best cases in low and high dimensions, respectively). **c** shows that the gain in using $\text{OWS}_{\text{ideal}}$ decreases to around 10% for problems in high-dimension (BH, CB, H3, H6, ER, and DP are the initials of the test problem and in parenthesis it is the number of points on the DOEs). For **d**, **e**, in parenthesis, it is the median and standard deviation when using 5 surrogates (after which there is no improvement in practice). While in theory the surrogate with best RMSE (BestRMSE) can be beaten ($\text{OWS}_{\text{ideal}}$), in practice, the quality of information given by the cross-validation errors makes it very difficult in low dimensions. In high dimension the quality permits gain, but it is very limited. See Appendix 2 for details about boxplots

(which in fact varies from DOE to DOE), there cannot be any gain. As the loss approaches zero, BestPRESS becomes more reliable in selecting the best surrogate of the set, instead of being a hedge against selecting an inaccurate surrogate. It can be observed that: (1) in low dimensions, the poor quality of information given by $PRESS_{RMS}$ makes some surrogates outperform BestPRESS (i.e., %difference closer to zero, or smaller loss); and (2) in high dimension, the quality of information given by $PRESS_{RMS}$ is better and as a consequence, BestPRESS becomes hard to beat. For Dixon–Price for example, the best three surrogates practically coincide with BestPRESS, so the %difference is close to zero for them.

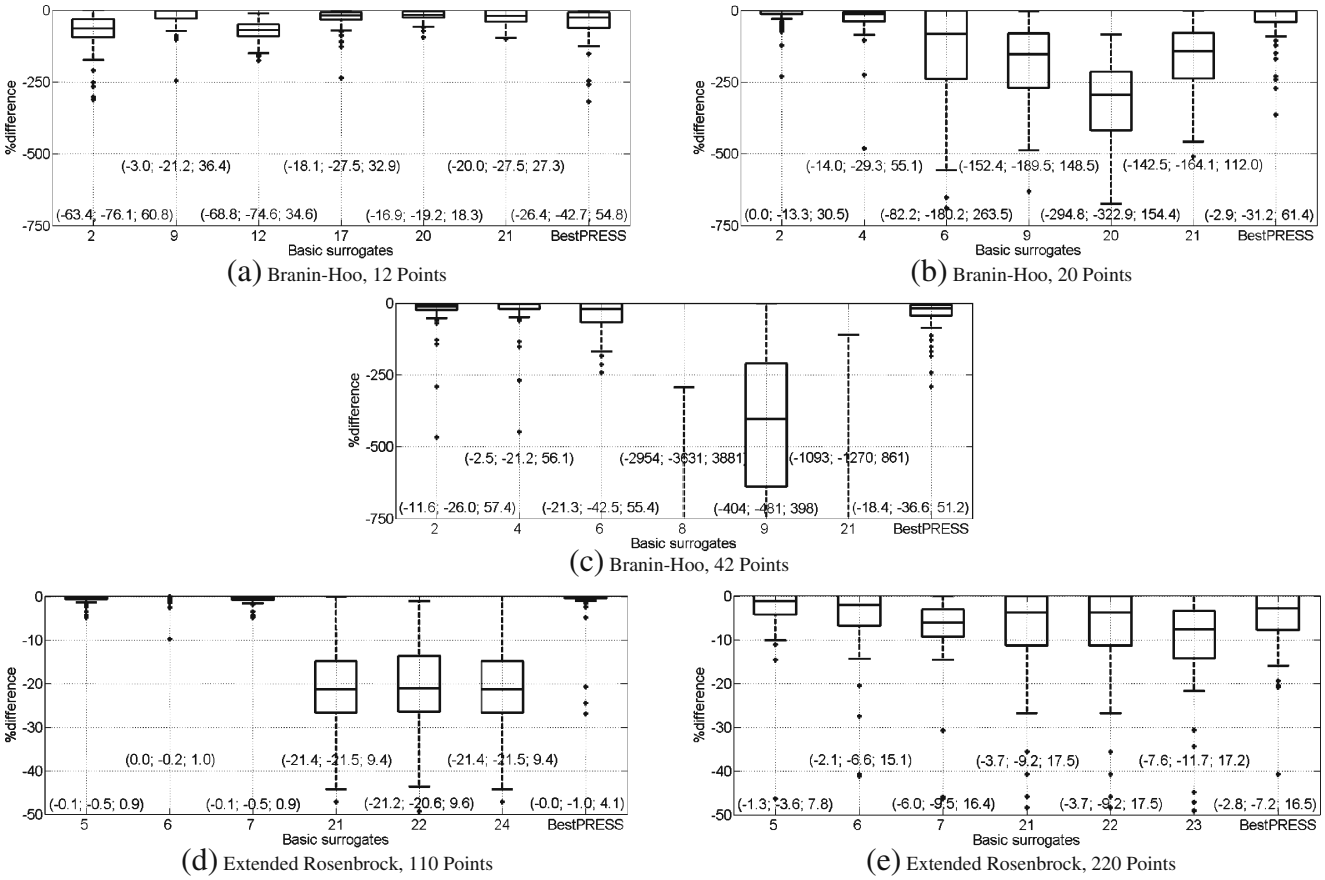
Next, we study if we can do any better by using a weighted average surrogate rather than BestPRESS. For a given DOE, surrogates are added according to rank based on $PRESS_{RMS}$. We start the study considering best possible performance of OWS, i.e., based on exact computation of the **C** matrix (which may not be possible in real world applications, but it is easy to compute in our set of analytical functions). Figure 7a, b exemplify what ideally happens with the %difference in the RMSE as we add surrogates to the WAS for the Branin–Hoo and Extended Rosenbrock functions fitted with 12 and 110 points, respectively. It is seen that we can potentially gain by adding more surrogates, but even the ideal potential gain levels off after a while. Disappointingly, Fig. 7c shows that the maximum possible gain decreases with dimensionality. Keeping the Branin–Hoo and extended Rosenbrock functions, Fig. 7d, e compare the ideal gain with the gain obtained with information based on cross-validation errors. It can be observed that while in theory (that is with OWS_{ideal}), BestPRESS as well as the surrogate with best RMSE (BestRMSE) can be beaten, in practice none of the *WAS* schemes is able to substantially improve the results of BestPRESS, i.e., more than 10% gain. In both low and high-dimensions, the best scenario is given by OWS_{diag}, which appears to tolerate well the use of a large number of surrogates. PWS is not able to handle the addition of poorly fitted surrogates. For this reason, the remainder of the paper does not include PWS. OWS is unstable in low dimensions while presenting small gains and the risk of losses in high dimensions.

Table 7 summarizes the information about the %difference in the RMSE for all test problems. It is then clear that in low-dimension very little can be done to improve BestPRESS (see mean and median). For the high-dimensional problems, OWS_{diag} seems to handle the uncertainty on the cross-validation errors better than OWS. However, the gains are limited to between

Table 7 %difference in the RMSE, defined in (13), of the *WAS* schemes and BestPRESS for each test problem when using all 24 surrogates

Test problem		Branin–Hoo (12 points)	Branin–Hoo (20 points)	Branin–Hoo (42 points)	Camelback (12 points)	Hartman3 (20 points)	Hartman6 (56 points)	Extended Rosenbrock (110 points)	Extended Rosenbrock (220 points)	Dixon–Price, 24 surrogates (182 points)	Dixon–Price, 21 surrogates (182 points)
OWS_{ideal}	Median	62	69	66	54	34	12.3	20.8	25.8	9.2	30.5
	Mean	61	68	66	55	34	12.7	21.1	24.9	9.7	30.7
	SD	10	10	12	9	7	3.3	4.6	3.9	2.4	2.5
BestPRESS	Median	−26	−3	−18	−29	−21	−1.9	0.0	−3.2	0.00	0.0
	Mean	−43	−31	−37	−35	−28	−5.2	−1.0	−7.6	0.00	−2.6
	SD	55	61	51	29	31	8.2	4.2	16.6	0.03	5.2
OWS_{diag}	Median	−35	−24	−34	−23	−18	−2.7	6.1	11.4	−7.0	4.4
	Mean	−43	−41	−51	−42	−24	−3.9	5.3	9.1	−7.3	4.1
	SD	39	55	61	131	26	5.2	4.2	11.8	4.9	4.2
OWS	Median	−119	−249	−61	−84	−182	−31	6	15.6	−9	17.2
	Mean	−144	−356	−104	−181	−205	−37	−13	13.6	−153	15.5
	SD	99	374	151	769	120	24	182	13.5	1396	10.7

For the Branin–Hoo function, the number of points used for fitting the surrogates is varied. For the Dixon–Price function, both the full set of surrogates and a partial set obtained when the first three best surrogates were left out are considered and used. The sign of %difference indicates that there is loss (−) or gain (+) in using the specific WAS scheme when compared with the best surrogate of the set in terms of RMSE



(a) Branin-Hoo, 12 Points

(b) Branin-Hoo, 20 Points

(c) Branin-Hoo, 42 Points

(d) Extended Rosenbrock, 110 Points

(e) Extended Rosenbrock, 220 Points

Fig. 8 %difference in the RMSE, defined in (13), for the overall best six surrogates and BestPRESS. In *parenthesis*, it is the median, mean, and standard deviation. Values closer to zero indicate better fits. Numbers on the abscissa indicate the surrogate number in Table 1. It can be seen that BestPRESS has performance comparable with the best three surrogates of the set

10% and 20%. As noted earlier, the behavior for the Branin–Hoo function when going from 12 to 42 points is anomalous because surrogates 2, 4 and 6 become very similar and dependent on the DOE. While PRESS is less reliable in identifying the most accurate surrogate, this is less important, because as can be seen from Fig. 8, the differences are miniscule for these surrogates. The results of BestPRESS for the extended Rosenbrock function shown in Table 7 are also counter-intuitive. Figure 8 shows that unlike the case of 220 points, for 110 points the first three surrogates are equally much better than the remaining surrogates. This makes selection less risky for 110 points.

6 Conclusions

In this paper, we have explored the use of multiple surrogates for the minimum RMSE in meta-modeling. We explored (1) the generation of a large set of surrogates and the use of PRESS errors as a criterion for surrogate selection; and (2) a weighted average surrogate based on the minimization of the integrated square error (in contrast to heuristic schemes).

The study allows the conclusion that the benefits of both strategies depend on dimensionality and number of points:

- With sufficient number of points, $\text{PRESS}_{\text{RMS}}$ becomes very good for ranking the surrogates according to prediction accuracy. In general, $\text{PRESS}_{\text{RMS}}$ is good for filtering out inaccurate surrogates, and as the number of points increases $\text{PRESS}_{\text{RMS}}$ can also identify the best surrogate of the set (or another equally accurate surrogate).
- As the dimension increases the possible gains from a weighted surrogate diminish even though our ability to approximate the ideal weights improves. In the two dimensional problems, $\text{OWS}_{\text{ideal}}$ presented a median gain of 60%; while in practice, there was a loss of 25% and no improvement over BestPRESS. For the extended Rosenbrock function (nine variables), $\text{OWS}_{\text{ideal}}$ presented a median

gain of 20%; while in practice, there was a gain of just 6% (for both OWS_{diag} and OWS; however, the former presents better performance when considering the mean and standard deviation).

Therefore, we can say that using multiple surrogates and $PRESS_{RMS}$ for identifying the best surrogate is a good strategy that becomes ever more useful with increasing number of points. On the other hand, the use of weighted average surrogate does not seem to have the potential of substantial error reductions even with large number of points that improve our ability to approximate the ideal weighted surrogate.

Additionally, we can point out the following findings:

- For large number of points, $PRESS_{RMS}$ as obtained through the k-fold strategy successfully estimates the RMSE (as shown in Fig. 5i). In the set of test problems, there is very little improvement when performing more than 30 fits for the cross-validations (i.e. $k > 30$).
- At high point density (possible in low dimensions) the choice of the trend function (or regression function as it is also sometimes called) for kriging is not important. At very low point density (characteristic of high dimensions) the choice of correlation function is not important. Both situations lead to the emergence of multiple almost identical kriging surrogates.
- When using a WAS, OWS_{diag} seems to be the best choice, unless the large number of points allows the use of OWS. In the cases that we have studied, OWS_{diag} was able to stabilize the addition of lousy surrogates up to the point of using all created surrogates.
- While we have been able to generate a large number of diverse surrogates, we were less successful in generating a substantial number of diverse surrogates with comparable high accuracy. This challenge is left for future research.

Acknowledgements This research was supported by National Science Foundation, USA (grant no. 0423280) and Coordenação de Aperfeiçoamento de Pessoal de Nível Superior, Brazil (proc. no. 3327/06-0). Authors gratefully acknowledge these grants.

Appendix 1. Surrogate modeling techniques

The principal features of the surrogate modeling techniques used in this study are described in the following sections.

1.1 Kriging (KRG)

KRG is named after the pioneering work of the South African mining engineer D.G. Krige. It estimates the value of a function as a combination of a known function $f_i(\mathbf{x})$ (e. g., a linear model such as a polynomial trend) and departures (representing low and high frequency variation components, respectively) of the form:

$$\hat{y}(\mathbf{x}) = \sum_{i=1}^{m} \beta_i f_i(\mathbf{x}) + z(\mathbf{x}) \ , \tag{19}$$

where $z(\mathbf{x})$ is assumed to be a realization of a stochastic process $Z(\mathbf{x})$ with mean zero, process variance σ^2, and spatial covariance function given by:

$$\mathrm{cov}\left(Z(\mathbf{x}_i), Z(\mathbf{x}_j)\right) = \sigma^2 R\left(\mathbf{x}_i, \mathbf{x}_j\right) \ , \tag{20}$$

where $R(\mathbf{x}_i, \mathbf{x}_j)$ is the correlation between $\mathbf{x}_i$ and $\mathbf{x}_j$.

The conventional KRG models interpolate training data. This is an important characteristic when dealing with noisy data. In addition, KRG is a flexible technique since different instances can be created by choosing different pairs of $f_i(\mathbf{x})$ and correlation functions. Finally, complexity and the lack of commercial software may hinder this technique from popularity in the near term (Simpson et al. 1998).

The Matlab code developed by Lophaven et al. (2002) was used to execute the KRG algorithm. More details about KRG are provided in Sacks et al. (1989), Simpson et al. (1998) and Lophaven et al. (2002).

1.2 Polynomial response surface (PRS)

The PRS approximation is one of the most well established meta-modeling techniques. In PRS modeling, a polynomial function is used to approximate the actual function. A second-order polynomial model can be expressed as:

$$\hat{y}(\mathbf{x}) = \beta_0 + \sum_{i=1}^{m} \beta_i x_i + \sum_{i=1}^{m} \sum_{j=1}^{m} \beta_{ij} x_i x_j \ , \tag{21}$$

The set of coefficients can be obtained by least squares and according to the PRS theory are unbiased and have minimum variance. Another characteristic is that it is possible to identify the significance of different design factors directly from the coefficients in the normalized regression model (in practice, using t-statistics). In spite of the advantages, there is always a

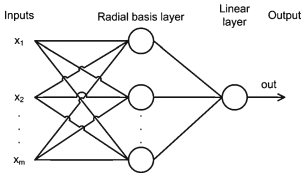


Fig. 9 Radial basis neural network architecture

drawback when applying PRS to model highly nonlinear functions. Even though higher-order polynomials can be used, it may be difficult to take sufficient sample data to estimate all of the coefficients in the polynomial equation.

See Box et al. (1978) and Myers and Montgomery (1995) for more details about PRS.

1.3 Radial basis neural networks (RBNN)

RBNN is an artificial neural network which uses radial basis functions as transfer functions. RBNN consist of two layers: a hidden radial basis layer and an output linear layer, as shown in Fig. 9. The output of the network is thus:

$$\hat{y}(\mathbf{x}) = \sum_{i=1}^{N} a_i \rho(\mathbf{x}, \mathbf{c}_i) , \tag{22}$$

where N is the number of neurons in the hidden layer, $\mathbf{c}_i$ is the center vector for neuron i, and a_i are the weights of the linear output neuron. The norm is typically taken to be the Euclidean distance and the basis function is taken to be the following:

$$\rho(\mathbf{x}, \mathbf{c}_i) = \exp\left(-\beta \|\mathbf{x} - \mathbf{c}_i\|^2\right) , \tag{23}$$

where $\exp(\cdot)$ is the exponential function.

RBNN may require more neurons than standard feedforward/backpropagation networks, but often they can be designed in a fraction of the time it takes to train standard feedforward networks. A crucial drawback for some applications is the need of many training points.

The native neural networks Matlab toolbox (Mathworks contributors 2004) was used to execute the RBNN algorithm. RBNN is comprehensively presented in Smith (1993) and Cheng and Titterington (1994).

1.4 Support vector regression (SVR)

SVR is a particular implementation of support vector machines (SVM). In SVR, the aim is to find $\hat{y}(\mathbf{x})$ that has at most a deviation of magnitude ε from each of the training data. Mathematically, the SVR model is given by:

$$\hat{y}(\mathbf{x}) = \sum_{i=1}^{p} \left(a_i - a_i^*\right) K(\mathbf{x}_i, \mathbf{x}) + b , \tag{24}$$

where $K(\mathbf{x}_i,\mathbf{x})$ is the so-called kernel function, $\mathbf{x}_i$ are different points of the original DOE and $\mathbf{x}$ is the point of the design space in which the surrogate is evaluated. Parameters a_i, a_i^*, and b are obtained during the fitting process.. Table 8 lists the kernel functions used in this work.

During the fitting process, SVR minimizes an upper bound on the expected risk unlike empirical risk minimization techniques, which minimize the error on the training data. This is done by using alternative loss functions. Figure 10 shows two of the most common possible loss functions. Figure 10a corresponds to the conventional least squares error criterion. Figure 10b illustrates the loss function used in this work, which is given by the following equation:

$$\text{Loss}(\mathbf{x}) = \begin{cases} \varepsilon, \text{ if } \left|y(\mathbf{x}) - \hat{y}(\mathbf{x})\right| \le \varepsilon \\ \left|y(\mathbf{x}) - \hat{y}(\mathbf{x})\right| , \text{otherwise} \end{cases} . \tag{25}$$

Table 8 Example of kernel functions

Gaussian radial basis function (GRBF)	$K(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{\|\mathbf{x}-\mathbf{x}'\|^2}{2\sigma^2}\right)$
Exponential radial basis function (ERBF)	$K(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{\|\mathbf{x}-\mathbf{x}'\|}{2\sigma^2}\right)$
Splines	$K(\mathbf{x}, \mathbf{x}') = 1 + \langle \mathbf{x}, \mathbf{x}' \rangle +$ $\frac{1}{2}\langle \mathbf{x}, \mathbf{x}' \rangle \min(\mathbf{x}, \mathbf{x}') - \frac{1}{6}(\min(\mathbf{x}, \mathbf{x}'))^3$
	$K(\mathbf{x}, \mathbf{x}') = \prod_i K_i(x_i, x_i')$
Anova-spline (Anova)	$K_i(x_i, x_i') = 1 + x_i x_i' + (x_i x_i')^2 +$ $(x_i x_i')^2 \min(x_i, x_i') - x_i x_i' (x_i + x_i') (\min(x_i, x_i'))^2 +$ $\frac{1}{3}(x_i^2 + 4 x_i x_i' + x_i'^2)(\min(x_i, x_i'))^3 +$ $-\frac{1}{2}(x_i + x_i')(\min(x_i, x_i'))^4 +$ $\frac{1}{5}(\min(x_i, x_i'))^5$

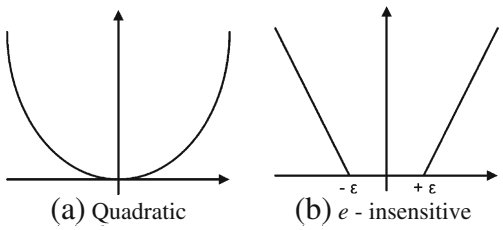


Fig. 10 Loss functions

The implication is that in SVR the goal is to find a function that has at most ε deviation from the training data. In other words, the errors are considered zero as long as they are less than ε.

Besides ε, the fitting the SVR model has a regularization parameter, C. Parameter C determines the compromise between the complexity and the degree to which deviations larger than ε are tolerated in the optimization formulation. If C is too large, the tolerance is small and the tendency is to have the most possible complex SVR model. An open issue in SVR is the choice of the values of parameters for both kernel and loss functions.

The Matlab code developed by Gunn (1997) was used to execute the SVR algorithm. To learn more about SVR see Gunn (1997), Clarke et al. (2005) and Smola and Scholkopf (2004).

Finally, SURROGATES ToolBox is used for an easy manipulation of all these different models. SURROGATES ToolBox, developed by Viana and Goel (2007), integrates several open-source tools providing a general-purpose MATLAB library of multidimensional function approximation methods.

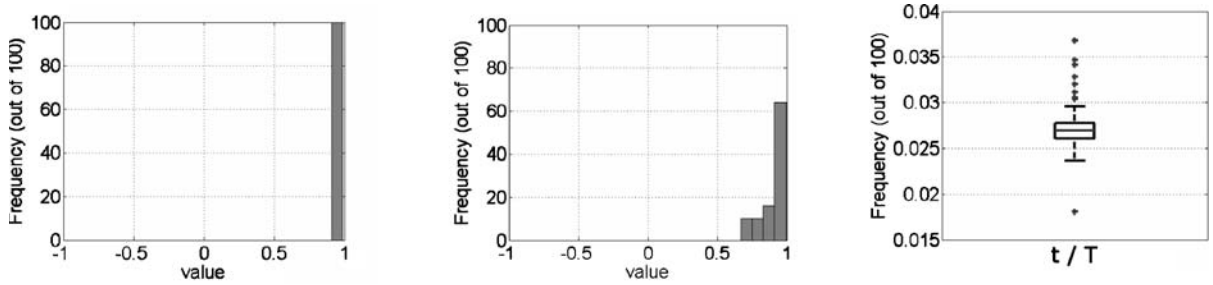
Appendix 2. Box plots

In a box plot, the box is defined by lines at the lower quartile (25%), median (50%), and upper quartile (75%) values. Lines extend from each end of the box and outliers show the coverage of the rest of the data. Lines are plotted at a distance of 1.5 times the interquartile range in each direction or the limit of the data, if the limit of the data falls within 1.5 times the interquartile range. Outliers are data with values beyond the ends of the lines by placing a "+" sign for each point.

Appendix 3. Reducing the costs of PRESS computation in kriging

As the number of point increases, the computation of PRESS via the leave-one-out approach may become prohibitive. One way to tackle this issue is to use the k-fold strategy (see section about cross-validation errors). However, for kriging, another alternative is to avoid the costly estimation of the correlation parameters and keep them the same as those found for the model fit to all data. We used the extended Rosenbrock function to study the potential of the second approach.

Figure 11a shows the correlation coefficient between the vector of $\text{PRESS}_{\text{RMS}}$, obtained with repeated computation of the correlation parameters, and RMSE, but only for the KRG models (total of six, shown in Table 1). Figure 11b shows the same plot but with correlation parameters kept the same as for the original fit (avoiding the costly optimization). Finally, Fig. 11c



(a) Correlation between $PRESS_{RMS}$ and $RMSE$ for the leave-one-out with computation of correlation coefficients.

(b) Correlation between $PRESS_{RMS}$ and $RMSE$ for the leave-one-out with frozen computation of correlation coefficients.

(c) Ratio of computational time of PRESS computation with frozen correlation parameters (t) to repeated computation of correlations (T).

Fig. 11 Performance of PRESS computation: work horse versus freezing correlation parameters. t and T have mean values of 23 and 850 s, respectively. Savings in time may hurt the correlation with the RMSE

compares computational times. It can be seen that the savings in time may hurt the correlation with RMSE. Ultimately, this would also translate in poorer data for the computation of the weights in a WAS strategy.

References

Acar E, Rais-Rohani M (2008) Ensemble of metamodels with optimized weight factors. In: 49th AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics, and materials, Schaumburg, IL, AIAA 2008-1884

Bishop CM (1995) Neural networks for pattern recognition. Oxford University Press, Oxford, pp 364–371

Box GEP, Hunter WG, Hunter JS (1978) Statistics for experimenters. Wiley, New York

Cheng B, Titterington DM (1994) Neural networks: a review from a statistical perspective. Stat Sci 9:2–54

Cherkassky V, Ma Y (2004) Practical selection of SVM parameters and noise estimation for SVM regression. Neural Netw 17(1):113–126

Clarke SM, Griebsch JH, Simpson TW (2005) Analysis of support vector regression for approximation of complex engineering analyses. J Mech Des 127:1077–1087

Dixon LCW, Szegö GP (1978) Towards global optimization 2. North-Holland, Amsterdam, The Netherlands

Goel T, Haftka RT, Shyy W, Queipo NV (2007) Ensemble of surrogates. Struct Multidisc Optim 33:199–216

Gunn SR (1997) Support vector machines for classification and regression. Technical Report, Image Speech and Intelligent Systems Research Group, University of Southampton, UK

Jin R, Chen W, Simpson TW (2001) Comparative studies of metamodelling techniques under multiple modeling criteria. Struct Multidisc Optim 23:1–13

Kohavi R (1995) A study of cross-validation and bootstrap for accuracy estimation and model selection. In: Proceedings of the fourteenth international joint conference on artificial intelligence, pp 1137–1143

Lee J (2007) A novel three-phase trajectory informed search methodology for global optimization. J Glob Optim 38(1):61–77

Lin Y, Mistree F, Tsui K-L, Allen JK (2002) Metamodel validation with deterministic computer experiments. In: 9th AIAA/ISSMO symposium on multidisciplinary analysis and optimization, Atlanta, GA, AIAA, AIAA-2002-5425

Lophaven SN, Nielsen HB, Søndergaard J (2002) DACE—A MATLAB kriging toolbox. Technical Report IMM-TR-2002-12, Informatics and Mathematical Modelling. Technical University of Denmark

Mathworks Contributors (2004) MATLAB® The language of technical computing, version 7.0 Release 14, The MathWorks Inc.

Mckay MD, Beckman RJ, Conover WJ (1979) A comparison of three methods for selecting values of input variables from a computer code. Technometrics 21:239–245

Meckesheimer M, Barton RR, Simpson TW, Booker A (2002) Computationally inexpensive metamodel assessment strategies. AIAA J 40(10):2053–2060

Myers RH, Montgomery DC (1995) Response surface methodology: process and product optimization using designed experiments. Wiley, New York

Sacks J, Welch WJ, Mitchell TJ, Wynn HP (1989) Design and analysis of computer experiments. Stat Sci 4(4):409–435

Simpson TW, Peplinski J, Koch PN, Allen JK (1998) On the use of statistics in design and the implications for deterministic computer experiments. In: Proceedings of the design theory and methodology (DTM'97), Paper No. DETC97/DTM-3881. ASME

Smith M (1993) Neural networks for statistical modeling. Von Nostrand Reinhold, New York

Smola AJ, Scholkopf B (2004) A tutorial on support vector regression. Stat Comput 14:199–222

Venkataraman S, Haftka RT (2004) Structural optimization complexity: what has Moore's law done for us? Struct Multidisc Optim 28:375–387

Viana FAC, Goel T (2007) SURROGATES ToolBox, http://fchegury.googlepages.com

Zerpa L, Queipo NV, Pintos S, Salager J (2005) An optimization methodology of alkaline–surfactant–polymer flooding processes using field scale numerical simulation and multiple surrogates. J Pet Sci Eng 47:197–208