Struct Multidisc Optim (2014) 49:969–978
DOI 10.1007/s00158-013-1028-0

RESEARCH PAPER

Simultaneous optimization of shape parameters and weight factors in ensemble of radial basis functions

Erdem Acar

Received: 22 March 2013 / Revised: 27 October 2013 / Accepted: 18 November 2013 / Published online: 17 December 2013

Abstract Radial basis functions (RBFs) are approximate mathematical models that can mimic the behavior of fast changing responses. Different formulations of RBFs can be combined in the form of an ensemble model to improve prediction accuracy. The conventional approach in constructing an RBF ensemble is based on a two-step procedure. In the first step, the optimal values of the shape parameters of each stand-alone RBF model are determined. In the second step, the shape parameters are fixed to these optimal values and the weight factors of each stand-alone RBF model in the ensemble are optimized. In this paper, simultaneous optimization of shape parameters and weight factors is proposed as an alternative to this two-step procedure for further improvement of prediction accuracy. Gaussian, multiquadric and inverse multiquadric RBF formulations are combined in the ensemble model. The efficiency of the proposed method is evaluated through example problems of varying dimensions from two to twelve. It is found that the proposed method improves the prediction accuracy of the ensemble compared to the conventional two-step procedure for the example problems considered.

Keywords Ensemble · Radial basis function · Shape parameter · Weight factor

E. Acar (✉)
Department of Mechanical Engineering, TOBB University of Economics and Technology, Söğütözü Cad. No: 43, Söğütözü, Ankara 06560, Turkey
e-mail: acar@etu.edu.tr

1 Introduction

Radial basis functions (RBFs) are approximate mathematical models used as surrogates for fast changing and computationally expensive simulations. RBFs have many attractive features including (i) their capability of accurately modelling arbitrary functions, (ii) their capability of handling scattered training points in multiple dimensions, and (iii) their relatively simple implementation compared to Kriging and neural networks (Mullur and Messac 2005). Due to these capabilities, RBFs have been used in many engineering applications. Hardy (1971, 1990, 1992) used RBFs to predict potential or temperature on the Earth's surface at some desired points. Arad et al. (1994) used RBFs for image warping of facial expressions. Tu and Barton (1997) used RBFs as surrogates for electronic circuit simulation models. Zala and Barrodale (1999) used RBFs to warp aerial photographs to orthomaps. Kremper et al. (2002) used RBFs in neuro-physic applications to classify neural signals. Papila et al. (2002) used RBFs for design optimisation of propulsion system and turbo-machinery components. Reddy and Ganguli (2003) used RBFs to predict structural damage in helicopter rotor blades. Wuxing et al. (2004) used RBFs for gear fault classification. Sonar et al. (2006) used RBFs for predicting the surface roughness in a turning process. Zhang et al. (2006) used RBFs for optimising a microelectronic packaging system. Young et al. (2007) used RBFs to predict responses of control systems used in aircraft. Sjögren (2009) used RBFs for multi-objective design of antennas.

The accuracy of the RBF models depends heavily on the shape parameters. Several methods have been suggested in the literature for selecting the shape parameters. Hardy (1971), Franke (1982), Kansa (1990a, b) and Fasshauer (2002) proposed empirical formulations for selecting good values for the shape parameters. Carlson and Foley (1991)

Table 1 Parameters used in Hartman-3 function, $j = 1, 2, 3$

i	a_{ij}			c_i	p_{ij}		
1	3.0	10.0	30.0	1.0	0.3689	0.1170	0.2673
2	0.1	10.0	35.0	1.2	0.4699	0.4387	0.7470
3	3.0	10.0	30.0	3.0	0.1091	0.8732	0.5547
4	0.1	10.0	35.0	3.2	0.03815	0.5743	0.8828

and then Foley (1994), with an improved procedure, proposed computing the RBF shape parameters by minimizing mean square error evaluated at a set of test points. As the use of test points is computationally prohibitive when the responses are calculated through time-consuming analysis models (e.g., high-fidelity finite element simulations), Rippa (1999) proposed computing the RBF shape parameters by minimizing mean square of cross validation error (CVE) evaluated at training points. Evaluation of CVE becomes computationally costly when the number of training points is large. In that case, numerically more efficient techniques (Wang 2004; Roque and Ferreira 2010) can be used to compute CVE.

Alternative to using a single RBF model, different stand-alone RBF models can be combined to construct an ensemble model. The idea of using an ensemble model can be traced to the development of ensembles of neural networks by Perrone and Cooper (1993) with further refinement by Bishop (1995). The resulting ensemble model takes advantage of the prediction ability of each stand-alone model to increase the prediction accuracy. Similar to combining neural networks in an ensemble model, other surrogate models such as RBFs can also be combined to form an ensemble (Tumer et al. 1998; Gutta et al. 2000; Hernàndez-Espinosa et al. 2004). The conventional approach in constructing an RBF ensemble is based on a two-step procedure (Gutta et al. 2000). In the first step, the optimal values of the shape parameters of each RBF model are determined. In the second step, the shape parameters are fixed at the previously found values and the optimal values of the weight factors of each RBF model are found. Instead of determining the optimal values of shape parameters and weight factors in two steps, it may be more advantageous to unify these two steps to perform simultaneous optimization of shape parameters and weight factors.

In this paper, simultaneous optimization of shape parameters and weight factors in ensemble of RBFs is proposed. Gaussian, multiquadric and inverse multiquadric stand-alone RBF models are combined in the ensemble. The paper is organized as follows. Section 2 provides a brief description of RBFs. Section 3 presents the current practice in choosing shape parameters and weight factors in ensemble of RBFs. A new method for determining the shape parameters and weight factors is proposed in Section 4. Six benchmark mathematical problems and two structural mechanics problems used to measure the accuracy of the proposed method are presented in Section 5. The numerical procedure followed in this study is detailed in Section 6. The results of test problems are presented and discussed in Section 7, followed by concluding remarks given in Section 8.

2 Radial basis functions (RBFs)

Radial basis functions were originally developed to approximate multivariate functions based on scattered data (Buhmann 2003). For a data set consisting of the values of input variables and response values at N training points, the true function $y(\boldsymbol{x})$ can be approximated as

$$\hat{y}(\mathbf{x}) = \sum_{k=1}^{N} \lambda_k \phi \left(\|\mathbf{x} - \mathbf{x}_k\| \right) \tag{1}$$

where $\mathbf{x}$ is the vector of input variables, $\mathbf{x}_k$ is the value of $\mathbf{x}$ at the k^{th} training point, $r = \|\mathbf{x} - \mathbf{x}_k\| = \sqrt{(\mathbf{x} - \mathbf{x}_k)^T (\mathbf{x} - \mathbf{x}_k)}$ is the Euclidean norm representing the radial distance from the prediction point $\mathbf{x}$ to the training point $\mathbf{x}_k$, ϕ is a radially symmetric basis function, and λ_k are the unknown interpolation coefficients. Equation (1) represents a linear combination of a finite number of radially

Table 2 Parameters used in Hartman-6 function, $j = 1, \cdots, 6$

i	a_{ij}						c_i	p_{ij}					
1	10.0	3.0	17.0	3.5	1.7	8.0	1.0	0.1312	0.1696	0.5569	0.0124	0.8283	0.5886
2	0.05	10.0	17.0	0.1	8.0	14.0	1.2	0.2329	0.4135	0.8307	0.3736	0.1004	0.9991
3	3.0	3.5	1.7	10.0	17.0	8.0	3.0	0.2348	0.1451	0.3522	0.2883	0.3047	0.6650
4	17.0	8.0	0.05	10.0	0.1	14.0	3.2	0.4047	0.8828	0.8732	0.5743	0.1091	0.0381

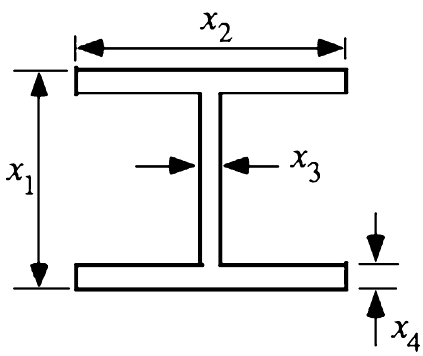


Fig. 1 The cross-section of the four variable beam problem

symmetric basis functions. The most popular RBF formulations include the Gaussian formulation with $\phi(r) = e^{-cr^2}$, the multiquadric formulation with $\phi(r) = \sqrt{r^2 + c^2}$, and the inverse multiquadric formulation with $\phi(r) = \frac{1}{\sqrt{r^2+c^2}}$ (in all of these formulations $c > 0$). The shape parameter c plays an important role in the accuracy of the RBF models.

Given the locations of training points $\mathbf{x}_k$ and calculated responses at training points $y(\mathbf{x}_k)$, the unknown interpolation coefficients $\boldsymbol{\lambda}$ are found by minimizing the residual R as

$$R = \sum_{j=1}^{N} \left[\mathrm{y}\left(\mathbf{x}_j\right) - \sum_{i=1}^{N} \lambda_i \phi\left(\left\|\mathbf{x}_j - \mathbf{x}_i\right\|\right) \right]^2 \tag{2}$$

Equation (2) can be expressed in matrix from as

$$[A]\{\boldsymbol{\lambda}\} = \{y\} \tag{3}$$

where $[A] = [\phi\|x_j - x_i\|]$, $i = 1 \in N$, $j = 1 \in N$, $\{\boldsymbol{\lambda}\}^T = \{\lambda_1, \lambda_2, \ldots\lambda_N\}^T$, and $\{y\}^T = \{y(x_1), y(x_2), \ldots, y(x_N)\}^T$. The unknown interpolation coefficient vector $\boldsymbol{\lambda}$ is obtained by solving (3).

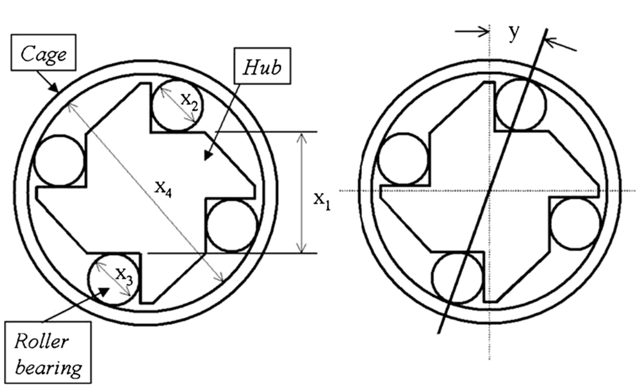


Fig. 2 The clutch assembly (Courtesy of Lee and Kwak 2006)

3 Ensemble of radial basis functions (ERBF)

An ensemble of radial basis functions (ERBF) can be constructed by using a weighted average of different stand-alone RBF models as

$$\hat{y}_{ens}(x) = \sum_{i=1}^{M} w_i \hat{y}_i(x) \tag{4}$$

where $\hat{y}_{ens}$ is the prediction of the ensemble, M is the number of stand-alone RBF models used, w_i is the weight factor for the $\mathrm{i^{th}}$ stand-alone RBF model and $\hat{y}_i$ is the prediction of the $\mathrm{i^{th}}$ stand-alone RBF model. The weight factors satisfy

$$\sum_{i=1}^{M} w_i = 1 \tag{5}$$

The conventional approach in constructing an ERBF is based on a two-step procedure. In the first step, the optimal values of the shape parameters, c_i, of each stand-alone RBF model are determined such that the prediction accuracy of each stand-alone RBF model is maximized. The shape parameters are usually found from solving the optimization problem in (6) so that the mean square cross validation error is minimized.

$$\text{Find} c_i,\ i = 1 \in M \tag{6.1}$$

$$\min \quad MSE_{CV}\left\{\hat{y}_i\left(c_i, \boldsymbol{x}^k\right), y\left(x^k\right), k = 1 \in N\right\} \tag{6.2}$$

Here MSE_{CV} is the mean square cross validation error calculated at the training points, and it is calculated from

$$MSE_{CV} = \frac{1}{N}\sum_{k=1}^{N}\left(y^k - \hat{y}^{(k)}\right)^2 \tag{7}$$

where N is the number of training points, y^k is the true response at $\boldsymbol{x}_k$ and $\hat{y}^{(k)}$ is the corresponding predicted value from the stand-alone RBF model constructed using all except the k^{th} training point. Computational cost of MSE_{CV} evaluation increases as the number of training points increases. An efficient method for cross validation error computation proposed by Rippa (1999) is used in this paper so that MSE_{CV} is calculated from

$$MSE_{CV} = \sum_{k=1}^{N}\left(\boldsymbol{\lambda}_k / \mathbf{A}_{kk}^{-1}\right)^2 \tag{8}$$

Table 3 Mean and standard deviation of the geometric variables

Variable	Mean	Standard deviation
x_1	55.29	0.0793
x_2	22.86	0.0043
x_3	22.86	0.0043
x_4	101.60	0.0793

Table 4 Summary of training and test data used in each problem

Problem	Number of variables	Number of points in a training set	Number of points in a test set	Number of training and test set
Branin-Hoo	2	20	1,000	1,000
Camelback	2	20	1,000	1,000
Hartman-3	3	30	1,000	1,000
Four variable beam	4	40	1,000	600
Fortini's clutch	4	40	1,000	600
Hartman-6	6	60	1,000	400
Extended Rosenbrock	9	110	1,000	100
Dixon-Price	12	182	1,000	25

where $\boldsymbol{\lambda}$ and $\mathbf{A}$ are defined earlier in (3). If needed, numerically more efficient techniques (e.g., Wang 2004; Roque and Ferreira 2010) can also be used to compute MSE_{CV}.

In the second step, the shape parameters of each stand-alone RBF model are fixed at the previously found optimal values and the weight factors, w_i, for the stand-alone RBF models are usually chosen such that the mean square cross validation error is minimized.

$$\text{Find} \quad w_i, i = 1 \in M \tag{9.1}$$

$$\min \quad MSE_{CV}\left\{\hat{y}_{ens}\left(w_i, \hat{y}_i\left(x^k\right)\right), y\left(x^k\right), k = 1 \in N\right\} \tag{9.2}$$

$$\text{s.t.} \quad \sum_{i=1}^{M} w_i = 1 \tag{9.3}$$

4 Proposed method for constructing an ERBF

The conventional approach in constructing an RBF ensemble has a shortcoming that the shape parameters for stand-alone RBF models may not be optimal for the ensemble when stand-alone RBF models are combined to form an ensemble. The selection of shape parameters together with the weight factors is a better strategy. That is, the conventional two-step approach can be modified to a unified step such that the shape parameters as well as weight factors can be optimized simultaneously. In this study, it is proposed that the shape parameters and weight factors can be determined by solving the following optimization problem

$$\text{Find} \quad \{c_i, w_i\}, \quad i = 1 \in M \tag{10.1}$$

$$\min \quad MSE_{CV}\left\{\hat{y}_{ens}\left(w_i, \hat{y}_i\left(c_i, \boldsymbol{x}^k\right)\right), y\left(x^k\right), k=1 \in N\right\} \tag{10.2}$$

$$\text{s.t.} \quad \sum_{i=1}^{M} w_i = 1 \tag{10.3}$$

The proposed method increases the prediction accuracy of the constructed ensemble at the expense of increased dimensionality of the optimization problem and computational cost. The conventional approach requires solution of M number of one-dimensional optimization problems followed by a single M-dimensional optimization problem. However, the proposed method requires solution of a single 2M-dimensional optimization problem.

The inflation of the computational cost in the proposed method compared to the conventional approach is mainly due to the necessity for calculating cross validation errors repeatedly for each stand-alone RBF model as the shape parameter changes during optimization. In conventional approach, on the other hand, the cross validation errors for each stand-alone RBF model are fixed as the shape parameters are fixed.

Table 5 RMSE$_{CV}$ (average over 1,000 different training sets) and RMSE (average over 1,000 different training and test sets) of stand-alone and ensemble models for the Branin-Hoo problem

	RBF_G	RBF_M	RBF_I	ERBF_C	ERBF_P
RMSE$_{CV}$	1.26(0.39)	1.00(0.38)	1.03(0.40)	0.95(0.41)	**0.93** (0.41)
RMSE	1.17(0.37)	**1.00**(0.67)	1.12(1.35)	1.11(1.31)	1.05(0.64)

The smallest error value in each category is shown in bold for ease of comparison. The numbers in parenthesis are the coefficient of variation of the corresponding quantity

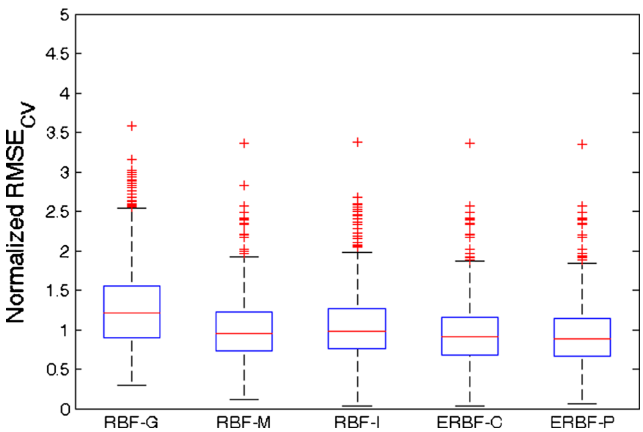


Fig. 3 Boxplots of RMSE$_{CV}$ over 1,000 training sets for the Branin-Hoo problem

5 Example problems

Overall eight example problems are used. The first six examples are widely used mathematical benchmark problems in the literature. The remaining two examples are structural mechanics problems, where the responses are described by analytic functions.

5.1 Mathematical problems

The mathematical benchmark problems are defined by the following analytical functions:

- Branin-Hoo function (two-variable)

$$y\left(x_1, x_2\right) = \left(x_2 - \frac{5.1x_1^2}{4\pi^2} + \frac{5x_1}{\pi} - 6\right)^2 + 10\left(1 - \frac{1}{8\pi}\right)\cos\left(x_1\right) + 10 \quad (11)$$

where $x_1 \in [-5, 10]$, and $x_2 \in [0, 15]$.

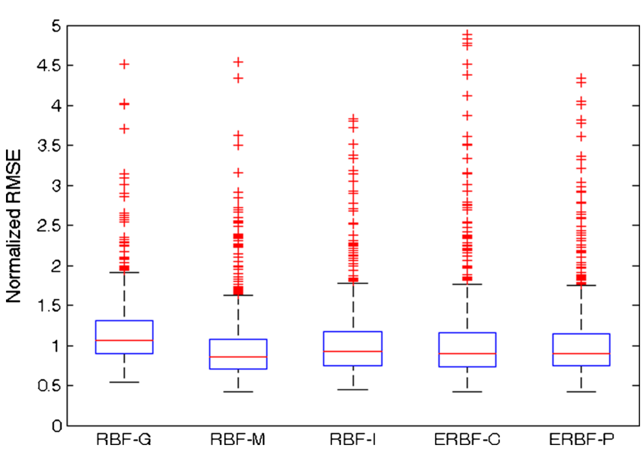


Fig. 4 Boxplots of RMSE over 1,000 training and test sets for the Branin-Hoo problem

- Camelback function (two-variable)

$$y\left(x_1, x_2\right) = \left(4 - 2.1x_1^2 + \frac{x_1^4}{3}\right)x_1^2 + x_1x_2 + \left(-4 + 4x_2^2\right)x_2^2 \quad (12)$$

where $x_1 \in [-3, 3]$, and $x_2 \in [-2, 2]$.

- Hartman function (three and six-variable)

$$y\left(\mathbf{x}\right) = -\sum_{i=1}^{m} c_i \exp\left[-\sum_{j=1}^{n} a_{ij}\left(x_j - p_{ij}\right)^2\right] \quad (13)$$

where $x_i \in [0, 1]$. Both three-variable ($n = 3$) and six-variable ($n = 6$) models of this function are considered, where m is taken four. The values of function parameters c_i, a_{ij} and p_{ij}, taken from Goel et al. (2007), are provided in Tables 1 and 2.

- Extended Rosenbrock function (nine-variable)

$$y\left(\mathbf{x}\right) = \sum_{i=1}^{m-1}\left[\left(1 - x_i\right)^2 + 100\left(x_{i+1} - x_i^2\right)^2\right] \quad (14)$$

where $x_i \in [-5, 10]$. Here nine-variable ($m = 9$) model of this function is considered.

- Dixon-Price function (twelve-variable)

$$y\left(\mathbf{x}\right) = \left(x_1 - 1\right)^2 + \sum_{i=2}^{m} i\left(2x_i^2 - x_{i-1}\right)^2 \quad (15)$$

where $x_i \in [-10, 10]$. Here twelve-variable ($m = 12$) model of this function is considered.

5.2 Structural mechanics problems

The structural mechanics problems are the following beam and clutch examples:

- Beam design (four-variable)

This four-variable I-beam (see Fig. 1) problem is taken from Messac and Mullur (2008). The beam is simply supported at both ends, and is subjected to an applied concentrated load. The beam length is $L = 2$ m and the applied load is $P = 600$ kN. The critical response for this problem is the maximum bending stress developed in the beam, which is calculated from

$$\sigma_{\max} = \frac{\frac{P}{2}\frac{x_1}{2}}{I},\ I = \frac{1}{12}\left[x_2x_1^3 - \left(x_2 - x_3\right)\left(x_1 - 2x_4\right)^3\right] \quad (16)$$

The ranges of the design variables are $0.1m \leq x_1, x_2 \leq 0.8m$ and $0.009m \leq x_3, x_4 \leq 0.05m$ as specified in Messac and Mullur (2008).

- Fortini's clutch (four-variable)

Table 6 Optimum shape parameters for RBF models and weight factors in ensemble models for the Branin-Hoo problem

	Conventional ensemble			Proposed ensemble		
	RBF_G	RBF_M	RBF_I	RBF_G	RBF_M	RBF_I
Shape parameter	6.07	1.05	1.28	4.98	1.00	1.24
Weight factor	0.11	0.70	0.19	0.13	0.67	0.20

Average values over 1,000 different training sets are provided

The other structural mechanics problem is taken from Lee and Kwak (2006). This overrunning clutch assembly, depicted in Fig. 2, is known as Fortini's clutch. The contact angle y is given in terms of the geometric variables x_1 through x_4 as

$$y = \arccos\left[\frac{x_1 + 0.5\,(x_2 + x_3)}{x_4 - 0.5\,(x_2 + x_3)}\right] \tag{17}$$

The problem specified in Lee and Kwak (2006) is a reliability assessment problem for the clutch. The mean and standard deviations of the geometric variables are provided in Table 3. The ranges for these variables are taken as ± five times standard deviations away from the mean values.

6 Numerical procedure

For all example problems, Latin hypercube sampling (LHS) technique is used to select the locations of the training points such that the minimum distance between the design points is maximized. The MATLAB® routine "lhsdesign" and "maximin" criterion with a maximum of 100 iterations is used to obtain the locations of the training points. Random sampling is used to generate 1,000 test points for a specified training set.

To reduce the effect of random sampling, a varying number of different training sets are used for the example problems (see the last column of Table 4). The low computational cost allowed considering repetitive training and test sets. Hence, all the stand-alone RBF models and ensemble models are constructed multiple times with the error estimate being the average value corresponding to multiple versions (replicates) of the same model. In addition, for each training set, a different set of test points is used to reduce the bias in error estimation. To keep the computational cost affordable, the number of training sets is reduced as the number of variables is increased. There exists no globally accepted method to determine the number of training points. The most commonly followed two approaches are: (i) using ten times the number of variables, and (ii) using twice the number of coefficients in a full quadratic PRS. In this work, the number of points corresponding to both approaches is computed and the larger value is used.

The shape parameters of stand-alone RBFs and weight factors in the ensemble are calculated through optimization where the "*fmincon*" function (optimizer) of MATLAB based on sequential quadratic programming algorithm is used. Since *fmincon* is a gradient-based optimizer and the objective function being minimized is not necessarily convex, a multiple starting point strategy is used to increase the probability for the solution to converge to a global optimum.

7 Results and discussion

As noted earlier, the shape parameters of stand-alone RBFs and the weight factors in the ensemble model are selected such that the mean square cross validation error is minimized. Therefore, the root mean square error (RMSE) is chosen as the error metric of interest. The error values are normalized with respect to the most accurate stand-alone RBF model to provide a better comparison of different models. The stand-alone RBF models and ERBF models are designated using the following abbreviated symbols. The

Table 7 p-values of the two-sample t-test to determine the significance of the differences between the optimum values of the shape parameters and the weight factors for RBF models for the conventional ensemble and the proposed ensemble

	RBF_G	RBF_M	RBF_I
Shape parameter	1.89×10^{-14}	0.584	0.744
Weight factor	0.051	0.174	0.911

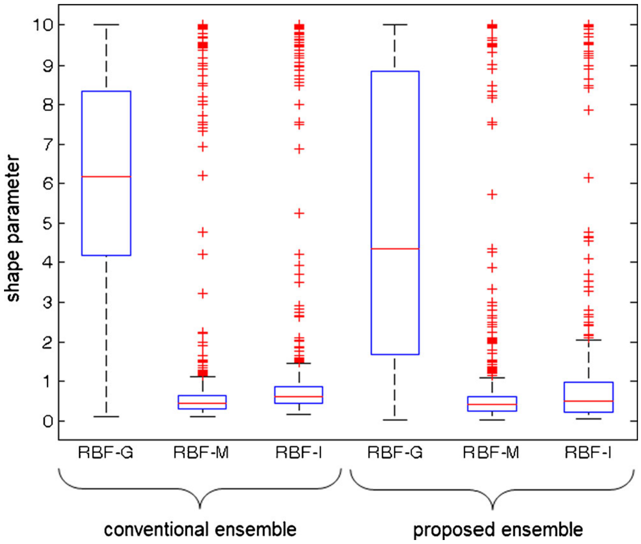


Fig. 5 Boxplots of shape parameters of stand-alone RBF models over 1,000 training sets for the Branin-Hoo problem

stand-alone RBF models with Gaussian, multiquadratic and inverse multiquadratic formulations are denoted by RBF_G, RBF_M and RBF_I, respectively. The ERBF model generated using conventional two-step procedure is denoted by ERBF_C. The ERBF model generated using proposed method is denoted by ERBF_P.

7.1 Branin-Hoo problem

Table 5 provides the root mean square cross validation error (RMSE_{CV}) and RMSE (calculated at test points) of all stand-alone RBF models as well as ERBF models for the Branin-Hoo problem. The most accurate model in terms of RMSE_{CV} metric is ERBF_P (the proposed ensemble), whereas the most accurate model in terms of RMSE metric is RBF_M, and ERBF_P is the second most accurate model. The proposed ensemble, ERBF_P, is more accurate than the conventional ensemble, ERBF_C, in terms of both RMSE_{CV} and RMSE metrics. The accuracy improvement that can be obtained using the proposed ensemble instead of the conventional ensemble is 1.6 % in terms of RMSE_{CV} and 5.6 % in terms of RMSE.

The mean and the coefficient of variation (COV) values in Table 5 are calculated based on 1,000 different training and test sets, so the mean values over the selected population sample have COV of $1/\sqrt{1000}$ times that of the native COV. For instance, the COV of the mean $RMSE_{CV}$ for RBF_G model is $0.39/\sqrt{1000} = 0.012$. This number provides an estimate of the standard error in the prediction of mean GMSE over 1,000 training sets, which is fairly small in this case.

Figures 3 and 4 show the boxplots for the error metrics RMSE_{CV} and RMSE, respectively, corresponding to the stand-alone and ensemble models for the Branin-Hoo problem. The boxplots provide a graphical depiction of how the normalized value of each metric varies over the range of training and test sets used. The bottom and top of each box represent the lower and upper quartile values, respectively, with the interior line representing the median. The broken line (whiskers) extending from each end of the box indicates the extent of the remaining data relative to the lower and upper quartiles. Here, the maximum whisker length is set at 1.5 times the inter-quartile range, and the data beyond this limit (if present) are characterized as outliers and represented by the + symbols.

Table 6 presents the average values (over 1,000 different training sets) of the optimum values of the shape parameters of the RBF models and the weight factors in the ensemble models. To determine whether the difference between the optimal values of the shape parameters and the weight factors in the ensemble models are significant for the conventional ensemble and the proposed ensemble, independent two-sample t-test is performed. The *p*-values are given in Table 7. The critical value of the *p*-value is typically chosen as 0.05. If the *p*-value is smaller than the critical value, then the difference is determined to be significant; otherwise the difference is not significant. For the optimal values of the shape parameters, Table 7 shows that the difference for the RBF_G is significant, whereas the differences for the RBF_M and RBF_I are not significant. For the weight factors in the ensemble models, Table 7 shows that the differences are not significant for all the RBF models. This indicates that the conventional two-step ensemble construction provides near optimal solutions. Similarly, the

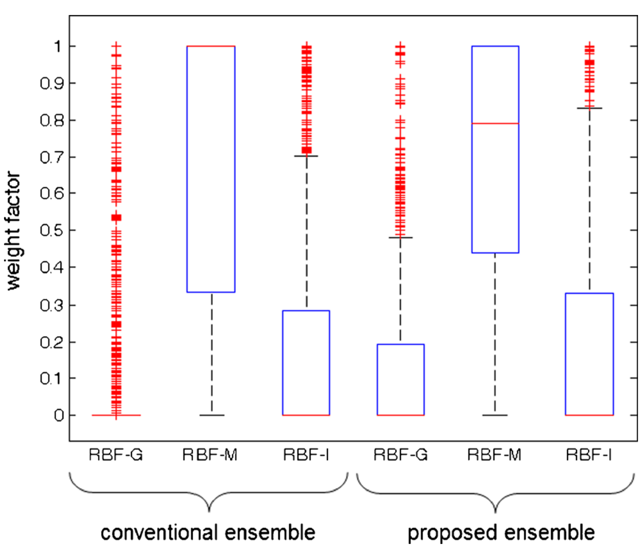


Fig. 6 Boxplots of weight factors of stand-alone RBF models over 1,000 training sets for the Branin-Hoo problem

Table 8 RMSE$_{CV}$ (average over 1,000 different training sets) of stand-alone and ensemble models for all example problems

	RBF_G	RBF_M	RBF_I	ERBF_C	ERBF_P
Branin-Hoo	1.26	1.00	1.03	0.95	**0.93**
Camelback	1.31	1.00	1.08	0.94	**0.93**
Hartman-3	1.00	1.11	1.02	0.96	**0.95**
Four variable beam	1.26	1.00	1.07	1.00	**0.99**
Fortini's clutch	1.21	1.00	1.06	1.00	**0.99**
Hartman-6	1.00	1.06	1.03	**0.99**	**0.99**
Extended Rosenbrock	1.32	1.00	1.14	1.00	**0.99**
Dixon-Price	1.13	1.00	1.05	1.00	**0.99**

The smallest error value in each category is shown in bold for ease of comparison

optimum values of the weight factors in the ensemble models are slightly different for the conventional ensemble and proposed ensemble. Boxplots for the shape parameters and weight factors, respectively, corresponding to the stand-alone and ensemble models for the Branin-Hoo problem are given in Figs. 5 and 6.

7.2 Other example problems

Table 8 provides RMSE$_{CV}$ of all stand-alone RBF models as well as ERBF models for all the example problems considered. RMSE$_{CV}$ of both ERBF models are smaller than RMSE$_{CV}$ of all stand-alone RBF models, and RMSE$_{CV}$ of ERBF_P (the proposed ensemble) is smaller than that of ERBF_C (the conventional ensemble). Amongst the stand-alone RBF models, RBF_M is the most accurate model for six example problems, while RBF_G and RBF_I models are the most accurate models for one example problem each. In addition, it is observed that the accuracy gain due to the use of ensemble models is larger for low dimensional problems than high dimensional problems.

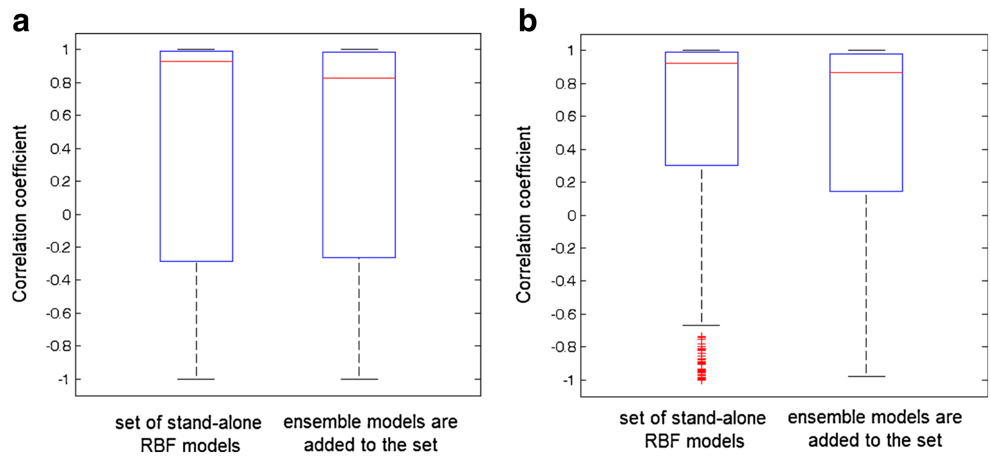
Table 9 presents RMSE of all stand-alone RBF models as well as ERBF models for all the example problems considered. It is found that the proposed ensemble is usually the second most accurate model whereas the most accurate model was one of the stand-alone RBF models (RBF_M for six example problems, RBF_G and RBF_I for one example problem each). This is the major benefit of using ensemble models over stand-alone models as the ensemble provides protection against using a poor stand-alone model. Comparison of results presented in Tables 8 and 9 reveals that the RMSE$_{CV}$ performs well for ranking the approximation models according to their prediction accuracy for a set of stand-alone models, but it does not perform well if the ensemble models are added to the set. This explains why ERBF models that outperform stand-alone RBF models in terms of RMSE$_{CV}$ does not necessarily outperform stand-alone models in terms of RMSE. Boxplots for the correlation coefficient between RMSECV and RMSE for Branin-Hoo and Hartman-6 problems (chosen as representative examples) are shown in Fig. 7.

Table 9 RMSE (average over 1,000 different training and test sets) of stand-alone and ensemble models for all example problems

	RBF_G	RBF_M	RBF_I	ERBF_C	ERBF_P
Branin-Hoo	1.17	**1.00**	1.12	1.11	1.05
Camelback	1.21	**1.00**	1.13	1.08	1.06
Hartman-3	**1.00**	1.11	1.03	1.02	1.02
Four variable beam	1.16	**1.00**	1.04	1.01	1.01
Fortini's clutch	1.16	**1.00**	1.05	1.01	1.01
Hartman-6	**1.00**	1.05	1.03	1.01	1.01
Extended Rosenbrock	1.32	**1.00**	1.14	**1.00**	**1.00**
Dixon-Price	1.14	1.00	1.06	1.00	**0.99**

The smallest error value in each category is shown in bold for ease of comparison

Fig. 7 Boxplots of correlation coefficient between $RMSE_{CV}$ and RMSE over 1,000 training and test sets for the **a** Branin-Hoo and **b** Hartman-6 problems



8 Concluding remarks

The conventional approach in constructing an ensemble model is based on a two-step procedure. In the first step, the optimal values of the model parameters for each stand-alone model are determined. In the second step, the model parameters are kept frozen to these optimal values and the weight factors of each stand-alone model in the ensemble model are optimized. In this paper, these two steps are unified and simultaneous optimization of model parameters and weight factors is proposed. Ensemble of RBF models with Gaussian, multiquadric and inverse multiquadric RBF formulations are considered, and the effectiveness of the proposed method is tests through example problems of varying dimensions. From the results of this study, the following conclusions could be drawn.

- The prediction accuracy of the proposed ensemble construction method was better than or equal to that the conventional two-step ensemble construction approach for all problems in terms of both cross-validation error and test point errors.
- The proposed method increased the prediction accuracy of the constructed ensemble at the expense of increased dimensionality of the optimization problem and computational cost. The conventional approach required solution of M number of one-dimensional optimization problems followed by a single M-dimensional optimization problem, where M is the number of stand-alone models in the ensemble. However, the proposed method required solution of a single 2M-dimensional optimization problem. The main computational cost driver was the necessity of calculating cross validation errors repeatedly for each stand-alone RBF model as the shape parameter changed during optimization. In conventional approach, on the other hand, the cross validation errors for each stand-alone RBF model were fixed as the shape parameters were fixed at the second step.
- When the prediction accuracy was evaluated using cross-validation errors ($RMSE_{CV}$ metric), the most accurate model was the proposed ensemble, followed by the conventional ensemble. The accuracy gain over the most accurate stand-alone model ranged from 1 % to 7 % for the example problems considered. This accuracy gain can be increased by increasing the number of stand-alone models in the ensemble (Acar and Solanki 2009). RBF models with linear, cubic, thin-plate spline and compactly supported formulations can be added to the ensemble.
- The reduction of $RMSE_{CV}$ obtained from ERBF over stand-alone RBF was larger for low dimensional problems than high dimensional problems. The error reduction was 7 % for two-variable problems, 5 % for three-variable problems and only 1 % for four and larger variable problems. Therefore, it could be concluded that the additional computational cost of the proposed ERBF construction procedure is more legitimate for low dimensional problems, and the use of the best stand-alone RBF model is more suitable for high dimensional problems.
- When the prediction accuracy was evaluated using error at test points (RMSE metric), the most accurate model was usually one of the stand-alone RBF models. The proposed ensemble was usually the second most accurate model. The reason for this result was that the capability of $RMSE_{CV}$ for ranking the approximation models according to their prediction accuracy was satisfactory for a set of stand-alone models, but not that satisfactory if the ensemble models are added to the set.
- Even though the proposed ensemble was usually the second most accurate model, it could protect against

using the poor RBF formulation (e.g., using multiquadratic formulation instead of Gaussian formulation for the three-variable Hartman problem) and thereby leading to a robust approximation.

- The differences between the optimum values of the shape parameters for RBF models for the conventional ensemble and proposed ensemble were not substantial. This indicates that the conventional two-step approach provides near optimal solutions. Similarly, the optimum values of the weight factors in the two ensemble models were only slightly different.
- Amongst the stand-alone RBF models, RBF with multiquadratic formulation was found to be the most accurate model for six example problems, while RBF with Gaussian formulation and RBF with inverse multiquadratic formulation were the most accurate models for one example problem each.

Acknowledgments The author gratefully acknowledges the funding provided by The Scientific and Technological Research Council of Turkey (TÜBİTAK), under award 109M537.

References

Acar E, Solanki K (2009) Improving the accuracy of vehicle crashworthiness response predictions using an ensemble of metamodels. Intern J Crashworthiness 14(1):49–61

Arad N, Dyn N, Reisfeld D, Yeshurun Y (1994) Image warping by radial basis functions: applications to facial expressions. Comput Vis Graph Image Process 56(2):161–172

Bishop CM (1995) Neural Networks for pattern recognition. Oxford University Press, New York, pp 364–369

Buhmann MD (2003) Radial basis functions: theory and implementations. Cambridge University Press, New York

Carlson RE, Foley TA (1991) The parameter R2 in multiquadric interpolation. Comput Math Appl 21:29–42

Fasshauer GE (2002) Newton iteration with multiquadrics for the solution of nonlinear PDEs. Comput Math Appl 43(3–5):423–438

Foley TA (1994) Near optimal parameter selection for multiquadric interpolation. J Appl Sci Comput 1:54–69

Franke R (1982) Scattered data interpolation: tests of some methods. Math Comput 38(157):181–200

Goel T, Haftka RT, Shyy W, Queipo NV (2007) Ensemble of surrogates. Struct Multidiscip Optim 33(3):199–216

Gutta S, Huang J, Philips J, Wechsler H (2000) Mixture of experts for classification of gender, ethnic origin, and pose of human faces. IEEE Trans Neural Netw 11(4):948–960

Hardy RL (1971) Multiquadric equations of topography and other irregular surfaces. J Geophys Res 76:1905–1915

Hardy RL (1990) Theory and applications of the multiquadric-biharmonic method. Comput Math Appl 19(8–9):163–208

Hardy RL (1992) A contribution of the multiquadric method: interpolation of potential inside the Earth: advances in the theory and applications of radial basis functions. Comput Math Appl 24(17):81–97

Hernàndez-Espinosa C, Fernàndez-Redondo M, Torres-Sospedra J (2004) Experiments on ensembles of radial basis functions. Artificial Intelligence and Soft Computing - ICAISC 2004. Lect Notes Comput Sci 3070:197–202

Kansa EJ (1990a) Multiquadrics. A scattered data approximation scheme with applications to computational fluid-dynamics. i. Surface approximations and partial derivative estimates. Comput Math Appl 19:127–145

Kansa EJ (1990b) Multiquadrics. A scattered data approximation scheme with applications to computational fluid-dynamics. ii. Solutions to parabolic, hyperbolic and elliptic partial differential equations. Comput Math Appl 19(8–9):147–161

Kremper A, Schanze T, Eckhorn R (2002) Classification of neural signals by a generalized correlation classifier based on radial basis functions. J Neurosci Methods 116:179–187

Lee SH, Kwak BM (2006) Response surface augmented moment method for efficient reliability analysis. Struct Saf 28:261–272

Messac A, Mullur AA (2008) A computationally efficient metamodeling approach for expensive multiobjective optimization. Optim Eng 9:37–67

Mullur AA, Messac A (2005) Extended radial basis functions: more flexible and effective metamodeling. AIAA J 43(6):1306–1315

Papila N, Shyy W, Griffin L, Dorney DJ (2002) Shape optimization of supersonic turbines using global approximation methods. J Propuls Power 18(3):509–518

Perrone MP, Cooper LN (1993) When networks disagree: ensemble methods for hybrid neural networks. In: Mammone, RJ (ed) Artificial neural networks for speech and vision, pp 126–142

Reddy RRK, Ganguli R (2003) Structural damage detection in a helicopter rotor blade using radial basis function neural networks. Smart Mater Struct 12:232–241

Rippa S (1999) An algorithm for selecting a good value for the parameter c in radial basis function interpolation. Adv Comput Math 11:193–210

Roque CMC, Ferreira AJM (2010) Numerical experiments on optimal shape parameters for radial basis functions. Numer Methods Partial Differ Equ 26(3):675–689

Sjögren D (2009) Statistical methods for improving surrogate models in antenna optimization, Master's Thesis, Chalmers University of Technology, Sweden

Sonar DK, Dixit US, Ojha DK (2006) The application of a radial basis function neural network for predicting the surface roughness in a turning process. Int J Adv Manuf Technol 27:661–666

Tu CH, Barton RR (1997) Production yield estimation by the metamodel method with a boundary-focused experiment design, Design Theory and Methodology Conference, Sacramento, CA, DETC97/DTM3870

Tumer K, Ramanujan N, Ghosh J, Richards-Kortum R (1998) Ensembles of radial basis function networks for spectroscopic detection of cervical precancer. IEEE Trans Biomed Eng 45(9):953–961

Wang BP (2004) Parameter optimization in multiquadric response surface approximations. Struct Multidiscip Optim 26:219–223

Wuxing L, Tseb PW, Guicaia Z, Tielina S (2004) Classification of gear faults using cumulants and the radial basis function network. Mech Syst Signal Process 18(2):381–389

Young A, Cao C, Patel V, Hovakimyan N, Lavertsky E (2007) Adaptive control design methodology for nonlinear-in-control systems in aircraft application. J Guid Control Dyn 30(6):1770–1782

Zala CA, Barrodale I (1999) Warping aerial photographs to orthomaps using thin plate splines. Adv Comput Math 11:211–227

Zhang T, Choi KK, Rahman S, Cho K, Baker P, Shakil M, Heitkamp D (2006) A hybrid surrogate and pattern search optimization method and application to microelectronics. Struct Multidiscip Optim 32(4):327–345