

Using the Correlation Criterion to Position and Shape RBF Units for Incremental Modelling

Xun-xian Wang

Neural Computing Research Group, Aston University, Birmingham B4 7ET, U.K.

Sheng Chen*, Chris J. Harris

School of Electronics and Computer Science University of Southampton, Southampton SO17 1BJ, U.K.

Abstract: A novel technique is proposed for the incremental construction of sparse radial basis function (RBF) networks. The correlation between a RBF regressor and the training data is used as the criterion to position and shape the RBF node, and it is shown that this is equivalent to incrementally minimise the modelling mean square error. A guided random search optimisation method, called the repeated weighted boosting search, is adopted to append RBF nodes one by one in an incremental regression modelling procedure. The experimental results obtained using the proposed method demonstrate that it provides a viable alternative to the existing state-of-the-art modelling techniques for constructing parsimonious RBF models that generalise well.

Keywords: Correlation, optimisation, radial basis function network, regression, sparse modelling.

1 Introduction

A basic principle in nonlinear data modelling is that of ensuring the smallest possible model which explains the training data. This parsimonious principle is particularly relevant in the construction of radial basis function (RBF) networks. The key questions in constructing a RBF network model are how many RBF units to use, the positions (centres) of the RBF nodes, and the shapes (variances or covariance matrices) of the RBF nodes. An efficient algorithm for constructing sparse RBF models is the forward selection based on the orthogonal least squares (OLS) algorithm^[1-7]. Typically, a fixed common variance is used for every RBF regressor and the RBF centres are chosen from the training input data points. Alternatively, the sparse kernel modelling techniques^[8-14] have widely be adopted in practical applications. These state-of-the-art sparse kernel modelling techniques also typically use a common variance for all the kernel functions and consider all the training input data as candidate kernel centres.

In the above-mentioned methods, the value of the RBF variance used has an important influence on the model sparsity level and its generalisation capability^[15]. Since the construction algorithms themselves do not provide this RBF variance, it has to be learnt using some other means, such as via cross valida-

tion^[16,17]. A RBF network will have better modelling capability if each node has its own covariance matrix^[18]. A recent work^[19] presents a method of estimating the RBF covariance matrices based only on a single training data set. As usual, all the training input data points are considered as candidate RBF centres and the covariance matrix of each candidate RBF node is determined by maximising the correlation function between the RBF regressor and the training data. The OLS algorithm of [7] is then used to select a sparse RBF model by optimising the model generalisation performance directly.

A RBF network can also be constructed sequentially as each training data sample is collected, and a class of algorithms for growing and pruning RBF networks has been developed, see [20] and the references within. As a new data sample is acquired, a decision is made to determine whether a new RBF node is to be added. Note that this class of sequential learning algorithms also places the RBF centres at the training input data points. Rather than restricting the RBF positions at the training input samples, the clustering-based learning methods (e.g. [21-23]) can be used to construct sparse RBF networks. In these clustering based learning methods, the number of clusters or the model size must be learnt by other means, for example, via cross-validation^[23], and the RBF variances also need to be decided using some other appropriate techniques. Alternatively, all the parameters of the RBF network, the RBF centres, variances or covariance matrices, and weights, can be learnt together via a non-

Manuscript received June 11, 2005; revised January 11, 2006.

*Corresponding author. *E-mail address:* sqc@ecs.soton.ac.uk

linear optimisation^[24]. The optimisation process associated with this nonlinear learning approach, however, is highly complex and non-convex, and the genetic algorithm (GA) has been suggested to solve this type of nonlinear learning problems^[25], at the cost of an increased computational complexity.

We present a construction method for producing sparse RBF networks by appending RBF units one by one in an incremental modelling process. At each stage of the construction process, a RBF node is constructed by determining its RBF centre and diagonal covariance matrix through optimising the correlation between the RBF regressor and the training data. It is shown that this approach is equivalent to incrementally minimising the training mean square error (MSE). Since this optimisation task is non-convex, a guided global search algorithm, referred to as the repeated weighted boosting search (RWBS)^[26], is adopted to perform this modelling optimisation. Because RBF centres are not restricted to be the training input data and each RBF node has an individually tuned diagonal covariance matrix, our proposed method can produce very sparse RBF networks that generalise well and it offers a viable alternative to the existing state-of-the-art sparse modelling methods.

Our proposed incremental modelling method is very different from the cascade-correlation incremental learning^[27]. In the cascade-correlation method, regression units are constructed on a variable space of increasing dimension, namely, the inputs to a unit being the original inputs and the outputs of the previously selected units. This increases the dimension of the learning problem and hence the associated computational complexity. Our proposed method is a truly incremental modelling from the input space to the output space. It has a desired geometric property that a RBF unit is constructed to fit the peak (in the sense of correlation magnitude) of the current modelling residual at each stage. Another difference is that our method adopts an efficient global optimisation algorithm in the incremental modelling, while the cascade-correlation method uses the gradient-based optimisation at each modelling stage to determine the corresponding regression unit. To avoid the problem of local minima, several candidate units with random initialisations are actually optimised in the cascade-correlation method at each stage and the best candidate unit is then selected.

2 The proposed RBF network construction method

Consider the regression modelling problem of approximating the N pairs of training data, $\{(\mathbf{x}_t, y_t)\}_{t=1}^N$,

with the RBF network

$$y(\mathbf{x}) = \sum_{i=1}^{n_M} w_i g_i(\mathbf{x}) + e(\mathbf{x}) = \hat{y}(\mathbf{x}) + e(\mathbf{x}) \quad (1)$$

where \mathbf{x} is the m -dimensional input variable, $e(\mathbf{x})$ is the modelling error at \mathbf{x} , and

$$\hat{y}(\mathbf{x}) = \sum_{i=1}^{n_M} w_i g_i(\mathbf{x}) \quad (2)$$

is the RBF model output with $g_i(\bullet)$ for $1 \leq i \leq n_M$ denoting the RBF regressors, w_i for $1 \leq i \leq n_M$ the RBF weights and n_M the number of RBF nodes. We will consider the general RBF regressor of the form

$$g_i(\mathbf{x}) = K \left(\sqrt{(\mathbf{x} - \boldsymbol{\mu}_i)^T \boldsymbol{\Sigma}_i^{-1} (\mathbf{x} - \boldsymbol{\mu}_i)} \right) \quad (3)$$

where $\boldsymbol{\mu}_i$ is the centre vector of the i th RBF unit, the diagonal covariance matrix has the form $\boldsymbol{\Sigma}_i = \text{diag}\{\sigma_{i,1}^2, \dots, \sigma_{i,m}^2\}$, and $K(\bullet)$ is the chosen RBF or kernel function. A widely used $K(\bullet)$ is the Gaussian function

$$g_i(\mathbf{x}) = e^{-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_i)^T \boldsymbol{\Sigma}_i^{-1} (\mathbf{x} - \boldsymbol{\mu}_i)}. \quad (4)$$

In the standard RBF or kernel modelling^[2-13], the RBF centres $\boldsymbol{\mu}_i$ are placed at the training input points \mathbf{x}_k , and all the covariance matrices take the same form $\boldsymbol{\Sigma}_i = \text{diag}\{\sigma^2, \dots, \sigma^2\}$ with σ^2 being the chosen RBF variance. A sparse RBF network is then constructed using a chosen construction algorithm, for example, the OLS algorithm^[1] or the support vector machine (SVM) algorithm^[9].

In this paper, we introduce an incremental construction algorithm for the RBF network with the general RBF node defined in (3). Let us first introduce the following notation

$$\left. \begin{aligned} y_i^{(0)} &= y_i \\ y_i^{(k)} &= y_i^{(k-1)} - w_k g_k(\mathbf{x}_i) \end{aligned} \right\} 1 \leq i \leq N. \quad (5)$$

Obviously, $y_i^{(k)}$ is the modelling error at \mathbf{x}_i after the k th RBF unit has been fitted. At the k th stage of the incremental modelling, the RBF regressor $g_k(\mathbf{x})$ is fitted to the training data set $\{y_i^{(k-1)}, \mathbf{x}_i\}_{i=1}^N$.

A. Correlation criterion for fitting a RBF unit

The correlation function between the RBF unit $g_k(\mathbf{x})$ and the training data set $\{y_i^{(k-1)}, \mathbf{x}_i\}_{i=1}^N$, given as

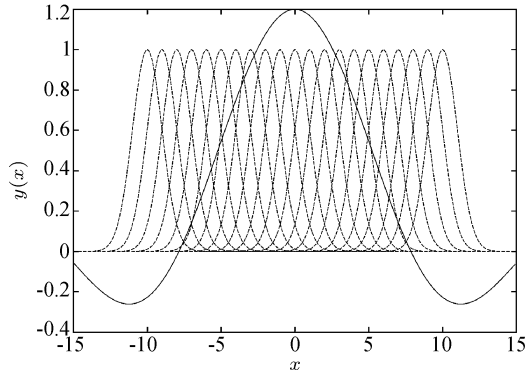
$$C_k(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = \frac{\sum_{i=1}^N g_k(\mathbf{x}_i) y_i^{(k-1)}}{\sqrt{\sum_{i=1}^N g_k^2(\mathbf{x}_i)} \sqrt{\sum_{i=1}^N (y_i^{(k-1)})^2}} \quad (6)$$

defines the “similarity” between the RBF regressor and the training data set. The larger value of $|C_k|$ is, the more similar they are.

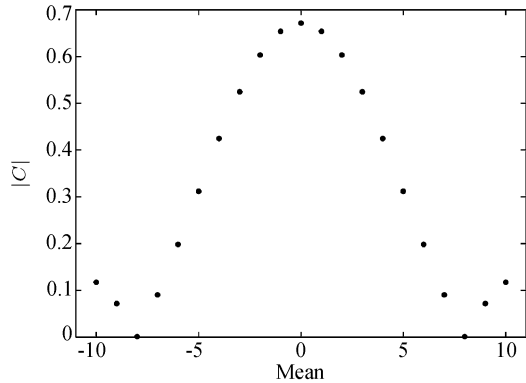
RBF positioning and shaping. The correlation C_k is a function of the RBF centre vector $\boldsymbol{\mu}_k$ and covariance matrix $\boldsymbol{\Sigma}_k$. Thus the correlation function (6) can be used for positioning and shaping the RBF unit $g_k(\bullet)$ by maximising $|C_k|$ with respect to $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$. An illustration is given in the one-dimensional space where the underlying data generation mechanism is given by

$$y(x) = 3.0 \frac{\sin(0.4x)}{x}. \quad (7)$$

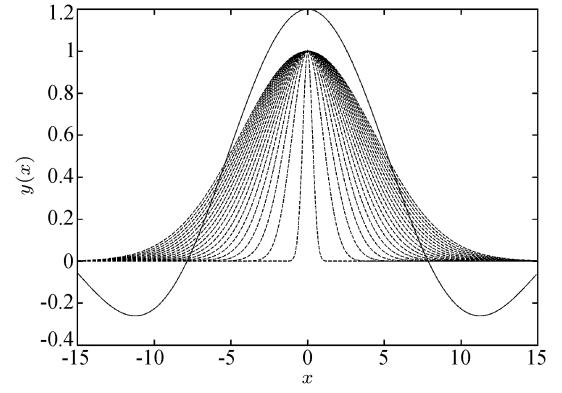
Fig. 1 (a) depicts the training data and the curves of the various Gaussian RBF units with different centres and a variance of 1.0, and Fig. 1 (b) shows the corresponding absolute correlation values between the training data and Gaussian RBF regressors; while Fig. 1 (c) depicts the training data and the Gaussian RBF units with zero centre and various values of variance, and Fig. 1 (d) plots the corresponding values of the correlation between the training data and RBF regressors. For this example, $|C_k(\mu_k, \sigma_k^2)|$ is maximised with $\mu_k = 0$ and $\sigma_k^2 = 12$.



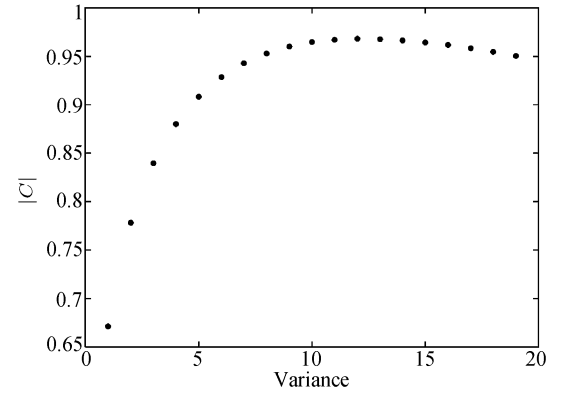
(a)



(b)



(c)



(d)

Fig. 1 Illustration of RBF positioning and shaping using the correlation criterion: (a) the solid curve is the training data set, and dashed curves are the Gaussian RBF units with different centres and a variance of 1.0, and (b) the corresponding values of the correlation between the training data and RBF units; while (c) the solid curve is the training data set, and dashed curves are the Gaussian RBF units with zero centre and various values of variance, and (d) the corresponding values of the correlation between the training data and RBF units

RBF weight calculation. After the determination of the k th regressor $g_k(\mathbf{x})$, the corresponding RBF weight w_k can be calculated by minimising the MSE for the k -unit RBF model with respect to w_k

$$\begin{aligned} \text{MSE}_k &= \frac{1}{N} \sum_{i=1}^N \left(y_i^{(k)} \right)^2 = \\ &= \frac{1}{N} \sum_{i=1}^N \left(y_i^{(k-1)} - w_k g_k(\mathbf{x}_i) \right)^2. \end{aligned} \quad (8)$$

This leads to the usual least squares solution

$$w_k = \frac{\sum_{i=1}^N y_i^{(k-1)} g_k(\mathbf{x}_i)}{\sum_{i=1}^N g_k^2(\mathbf{x}_i)}. \quad (9)$$

It can be shown that selecting RBF units by incrementally maximising $|C_k(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)|$ is identical to incrementally minimising the modelling MSE (8). In fact, substituting (5) into (8) with w_k given by (9) yields

$$\text{MSE}_k = \left(\frac{1}{N} \sum_{i=1}^N \left(y_i^{(k-1)} \right)^2 \right) (1 - C_k^2(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)). \quad (10)$$

Clearly maximising $|C_k(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)|$ is equivalent to minimising MSE_k with respect to $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$.

B. Incremental modelling to construct a RBF network

The proposed procedure for constructing a RBF network can now be summarised. Refer to the definition of the MSE (8) for the k -term RBF model. Give a preset modelling accuracy ξ_{MSE} , and set $k = 0$.

Do: $k = k + 1$

1) Determine the mean vector $\boldsymbol{\mu}_k$ and covariance matrix $\boldsymbol{\Sigma}_k$ of the k -th RBF unit by maximising the correlation criterion $|C_k(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)|$

2) Calculate the model weight w_k for the k -th RBF unit according to (9) and compute the modelling errors $y_i^{(k)} = y_i^{(k-1)} - w_k g_k(\mathbf{x}_i)$, $1 \leq i \leq N$

While $\text{MSE}_k < \xi_{\text{MSE}}$

The termination of the model construction process can alternatively be decided using cross validation^[16,17]. A simple method is to have a separate validation data set. The model construction is based on the training data set, while the performance of the selected model, the MSE (8), is monitored over the validation data set. The construction process is terminated when the MSE over the validation set stops improving. Alternatively, the Akaike information criteria^[28,29] and the optimal experimental design criteria^[6,30] can be employed to terminate the model construction procedure without the need to specify a modelling accuracy ξ_{MSE} .

C. Repeated weighted boosting search optimisation

It can be seen that at the k -th stage of construction, the task is to minimise the cost function

$$J(\mathbf{u}) = 1 - |C_k(\mathbf{u})| \quad (11)$$

where the parameter vector \mathbf{u} contains the centre vector $\boldsymbol{\mu}_k$ and covariance matrix $\boldsymbol{\Sigma}_k$ of the k -th RBF unit. This task may be carried out with a gradient based optimisation method^[27]. A gradient method however depends on the initial condition and may be trapped at some bad local minima. Alternatively, the global optimisation methods, such as the GA^[31,32] and adaptive simulated annealing (ASA)^[33,34], can be used. In this study, we employ a guided random search algorithm called the RWBS^[26] to perform this optimisation task. The RWBS algorithm is a simple yet efficient

global search algorithm. In several global optimisation applications investigated in [26], the RWBS algorithm achieved a similar convergence speed as the GA and ASA. The RWBS algorithm has additional advantages of requiring minimum programming effort and having very few algorithmic parameters that require to tune, compared with the GA or ASA. The detailed RWBS algorithm for fitting the k -th RBF unit is now summarised.

Specify the population size P_S , the number of generations in the repeated search N_G , and the accuracy for terminating the weighted boosting search ξ_B .

Outer loop: generations For $l = 1 : N_G$

Generation initialisation: Initialise the population by setting $\mathbf{u}_1^{(l)} = \mathbf{u}_{\text{best}}^{(l-1)}$ and randomly generating rest of the population members $\mathbf{u}_i^{(l)}$, $2 \leq i \leq P_S$, where $\mathbf{u}_{\text{best}}^{(l-1)}$ denotes the solution found in the previous generation. If $l = 1$, $\mathbf{u}_1^{(l)}$ is also randomly chosen

Weighted boosting search initialisation: Assign the initial distribution weightings $\delta_i(0) = \frac{1}{P_S}$, $1 \leq i \leq P_S$, for the population, and calculate the cost function value of each point $J_i = J(\mathbf{u}_i^{(l)})$, $1 \leq i \leq P_S$

Inner loop: weighted boosting search Set $t = 0$; For $t = t + 1$

Step 1: Boosting

1) Find

$$i_{\text{best}} = \arg \min_{1 \leq i \leq P_S} J_i \quad \text{and} \quad i_{\text{worst}} = \arg \max_{1 \leq i \leq P_S} J_i$$

Denote $\mathbf{u}_{\text{best}}^{(l)} = \mathbf{u}_{i_{\text{best}}}^{(l)}$ and $\mathbf{u}_{\text{worst}}^{(l)} = \mathbf{u}_{i_{\text{worst}}}^{(l)}$

2) Normalise the cost function values

$$\bar{J}_i = \frac{J_i}{\sum_{m=1}^{P_S} J_m}, \quad 1 \leq i \leq P_S$$

3) Compute a weighting factor β_t according to

$$\eta_t = \sum_{i=1}^{P_S} \delta_i(t-1) \bar{J}_i, \quad \beta_t = \frac{\eta_t}{1 - \eta_t}$$

4) Update the distribution weightings for $1 \leq i \leq P_S$

$$\delta_i(t) = \begin{cases} \delta_i(t-1) \beta_t^{\bar{J}_i}, & \text{for } \beta_t \leq 1 \\ \delta_i(t-1) \beta_t^{1-\bar{J}_i}, & \text{for } \beta_t > 1 \end{cases}$$

and normalise them

$$\delta_i(t) = \frac{\delta_i(t)}{\sum_{m=1}^{P_S} \delta_m(t)}, \quad 1 \leq i \leq P_S$$

Step 2: Parameter updating

1) Construct the $(P_S + 1)$ th point using the formula

$$\mathbf{u}_{P_S+1} = \sum_{i=1}^{P_S} \delta_i(t) \mathbf{u}_i^{(l)}$$

2) Construct the $(P_S + 2)$ th point using the formula

$$\mathbf{u}_{P_S+2} = \mathbf{u}_{\text{best}}^{(l)} + \left(\mathbf{u}_{\text{best}}^{(l)} - \mathbf{u}_{P_S+1} \right)$$

3) Compute the cost function values $J_i = J(\mathbf{u}_i)$, $i = P_S + 1, P_S + 2$, for these two points and find

$$i_* = \arg \min_{i=P_S+1, P_S+2} J_i$$

4) The pair $(\mathbf{u}_{i_*}, J_{i_*})$ then replaces $(\mathbf{u}_{\text{worst}}^{(l)}, J_{i_{\text{worst}}})$ in the population

If $\|\mathbf{u}_{P_S+1} - \mathbf{u}_{P_S+2}\| < \xi_B$, exit **inner loop**

End of inner loop

The solution found in the l -th generation is $\mathbf{u} = \mathbf{u}_{\text{best}}^{(l)}$

End of outer loop

This yields the solution $\mathbf{u} = \mathbf{u}_{\text{best}}^{(N_G)}$, i.e. $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$ of the k -th RBF unit.

The motivations and analysis of the RWBS algorithm as a global optimiser is detailed in [26]. To guarantee a global optimal solution as well as to achieve a fast convergence, P_S , N_G and ξ_B need to be set carefully. The appropriate values for these algorithmic parameters depend on the dimension of \mathbf{u} and how hard the objective function to be optimised, and generally they have to be found empirically. The elitist initialisation is very useful, as it keeps the information obtained by the previous search generation, which otherwise would be lost due to the randomly sampling initialisation. In the inner loop optimisation, there is no need for every members of the population to converge to a (local) minimum, and it is sufficient to locate where the minimum lies. Thus ξ_B can be set to a relatively large value. This makes the search efficient, achieving convergence with a small number of the cost function evaluations. A sufficiently large N_G should be used to ensure that the parameter space is sampled sufficiently.

3 Experimental results

Two simulated systems and two real-data sets were used to investigate the proposed RBF network construction method. Gaussian RBF units were used in all the examples. The RWBS algorithmic parameters, P_S , N_G and ξ_B , were set empirically. We also assumed that the desired modelling accuracy ξ_{MSE} could be chosen. We point out that other automatic termination criteria^[6] can be adopted as alternatives without the need of specifying ξ_{MSE} . The standard SVM algorithm with the ε -insensitive loss function^[9] was used as the benchmarker in the modelling experiments. The three learning parameters of the ε -insensitive SVM algorithm, the ε value and the value of the regularisation

parameter C as well as the RBF variance σ^2 , were determined empirically via cross validation.

Example 1. The 500 points of training data were generated from

$$y(x) = 0.1x + \frac{\sin x}{x} + \sin 0.5x + e \quad (12)$$

with equally spaced $x \in [-10, 10]$, where e was a Gaussian white noise with zero mean and variance 0.01. With the modelling accuracy set to $\xi_{\text{MSE}} = 0.012$, the proposed incremental RBF model construction procedure produced 6 Gaussian RBF units, as summarised in Table 1, and the construction process is also illustrated graphically in Fig. 2. The model output of the constructed 6-unit RBF model is superimposed on the noisy training data in Fig. 3 (a), and the modelling errors are shown in Fig. 3 (b). The MSE of this RBF model was 0.0119.

Table 1 Incremental construction of the RBF network for Example 1

step	centre	variance	weight	MSE
k	$\boldsymbol{\mu}_k$	σ_k^2	w_k	MSE_k
0	—	—	—	0.843 1
1	2.690 5	4.248 8	1.608 8	0.370 3
2	-4.083 7	2.185 3	-1.601 9	0.034 1
3	0.298 2	0.600 0	0.378 1	0.024 3
4	6.606 2	0.661 0	0.311 6	0.017 3
5	3.416 2	0.609 1	-0.224 2	0.013 8
6	-8.478 0	0.429 5	0.178 7	0.011 9

For the ε -SVM algorithm, it was found that $\sigma^2 = 4$, $\varepsilon = 0.245$ and $C = 10$ was appropriate, and the algorithm selected 15 support vectors (SVs) with the MSE value of 0.0126. The modelling performance of the resulting SVM model is shown in Fig. 4. The SVM algorithm achieves a similar modelling accuracy as the proposed incremental modelling method but it requires a much larger RBF network.

Example 2. This was a two-dimensional modelling problem involving data generated by the peaks function in Matlab. The peaks function is a function of two variables, obtained by translating and scaling Gaussian distributions. In the experiment, 625 training data points were produced with peaks (25,25) and a Gaussian white noise with variance 0.01 was added to the data generated. Fig. 5 (a) depicts the noisy training data. The modelling accuracy was set to $\xi_{\text{MSE}} = 0.05$. The proposed construction procedure, summarised in Table 2, constructed 9 Gaussian RBF units. The final modelling errors are shown in Fig. 5 (b), and the constructed 9-unit RBF model is depicted in Fig. 5 (c). The MSE value of the the resulting 9-unit RBF network was 0.048 1.

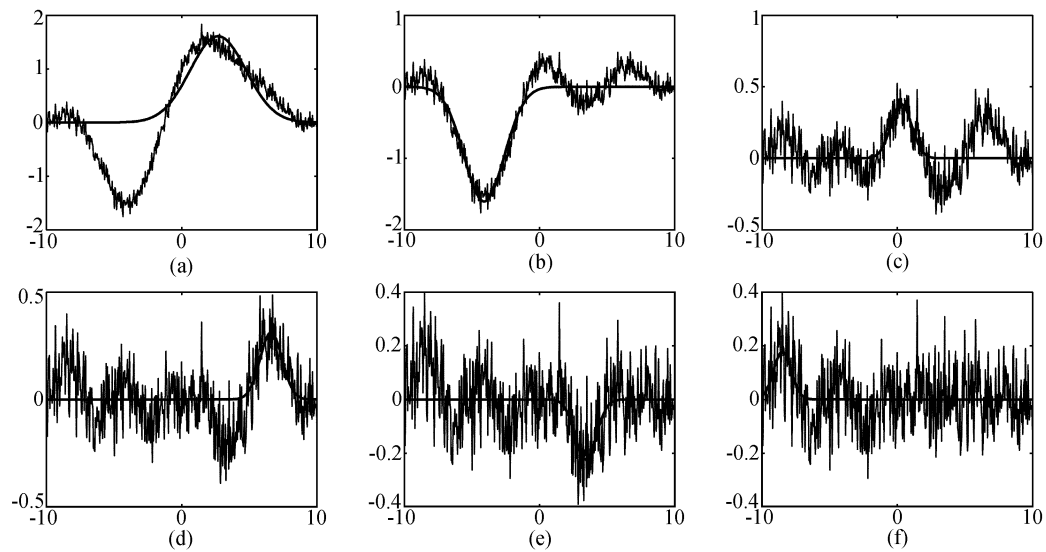


Fig. 2 Incremental construction of the RBF network for Example 1: in (a)–(f), the light curves are the modelling errors of the previous stage, $y_i^{(k-1)}$, and the dark curves are the fitted current RBF units, $w_k g_k(x_i)$, for $1 \leq k \leq 6$, respectively

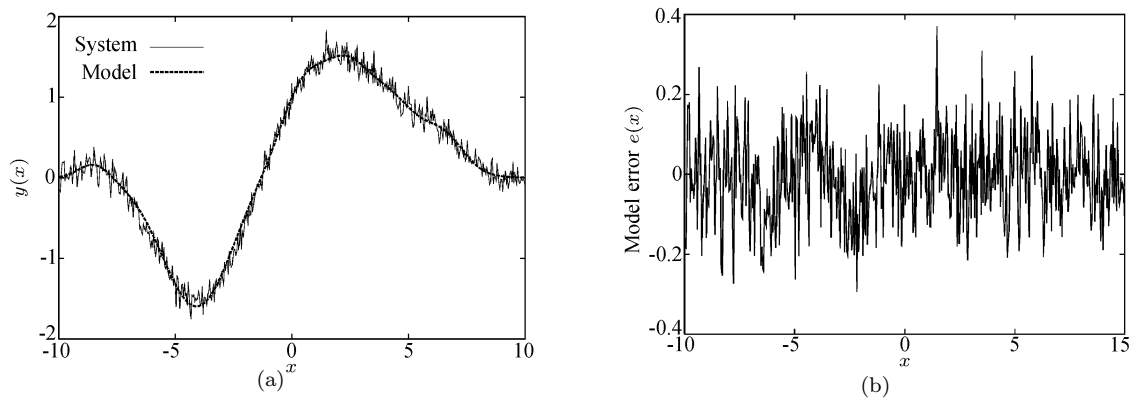


Fig. 3 Incremental construction of the RBF network for Example 1: (a) the solid curve is the noisy training data y_i and the dashed curve is the constructed 6-unit RBF model \hat{y} , and (b) the modelling error $e(x_i) = y_i - \hat{y}(x_i)$

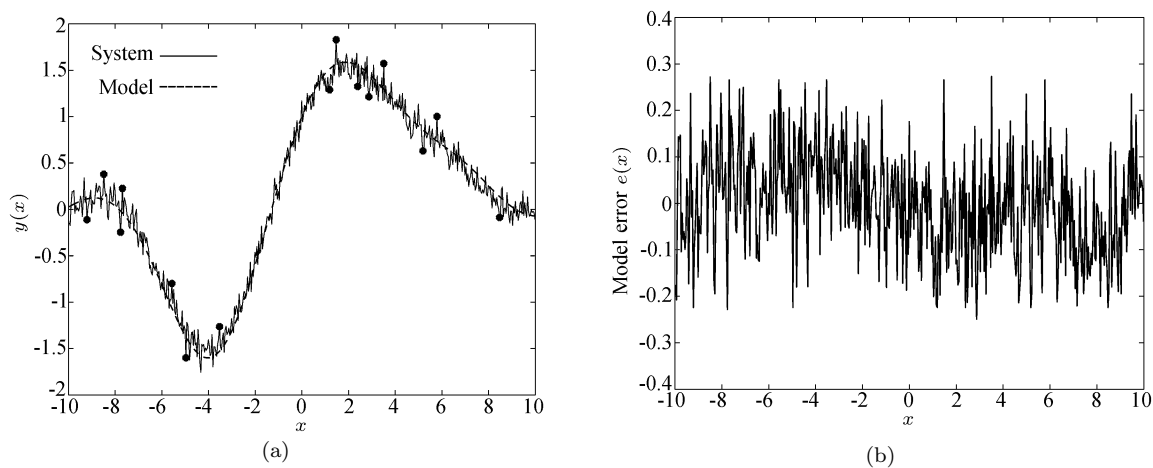


Fig. 4 SVM construction of the RBF network for Example 1: (a) the solid curve is the noisy training data y_i , the dashed curve is the constructed 15-unit RBF model \hat{y} and the points indicate the SVs, and (b) the modelling error $e(x_i) = y_i - \hat{y}(x_i)$

Table 2 Incremental construction of the RBF network for Example 2

step k	centre vector $\boldsymbol{\mu}_k$		covariance matrix $\boldsymbol{\Sigma}_k$		weight w_k	MSE	MSE $_k$
0	-		-		-	3.6323	
1	10.4505	15.4775	5.2804	2.6851	8.1114	1.6865	
2	11.0738	4.5417	3.0304	1.7384	-7.0448	0.7919	
3	14.6668	10.2624	2.0127	3.8346	3.8536	0.4679	
4	5.4481	11.2488	1.5582	2.2328	-3.4966	0.2888	
5	8.6135	8.1996	1.3214	1.6229	4.1949	0.0864	
6	11.7772	6.2996	2.1556	0.4471	-1.5106	0.0688	
7	6.7623	6.5534	0.8347	0.4666	1.2704	0.0609	
8	4.6504	9.1590	4.3879	0.7335	-0.7183	0.0536	
9	10.1679	1.9307	6.2416	0.5524	-0.6140	0.0481	

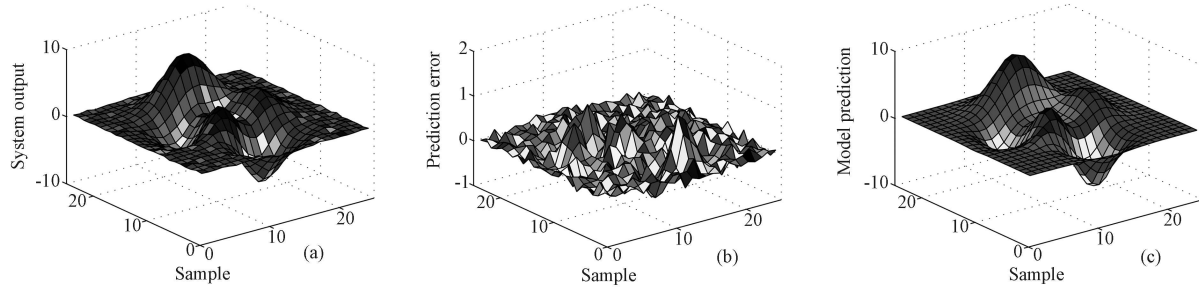


Fig. 5 Incremental construction of the RBF network for Example 2: (a) the noisy training data, (b) the modelling errors for the constructed 9-unit RBF model, and (c) the model outputs of the constructed 9-unit RBF model

For the SVM modelling, $\sigma^2 = 4$, $\varepsilon = 0.45$ and $C = 10$ were found to be appropriate, and the algorithm constructed a RBF model with a MSE value of 0.0471, which is similar to the MSE of the 9-unit RBF model constructed by the proposed algorithm. However, the constructed SVM model contained 42 SVs. The size of this SVM model is therefore more than four times of that produced by the proposed sparse construction algorithm. The modelling performance of this 42-unit RBF model is depicted in Fig. 6.

Example 3. This example modelled the relationship between the fuel rack position (input $u(t)$) and the engine speed (output $y(t)$) for a turbocharged, direct injection diesel engine operated at low engine speed. Detailed system description and experimental setup can be found in [35]. The data set, depicted in Fig. 7, contained 410 samples. The first 210 data points were used in training and the last 200 points in model validation. The previous study^[6] has shown that this data set can be modelled adequately as $y_i = F(\mathbf{x}_i) + e_i$ with $y_i = y(i)$, $\mathbf{x}_i = [y(i-1) u(i-1) u(i-2)]^T$, where $F(\bullet)$ describes the underlying system to be identified and e_i denotes the system noise. With the modelling accuracy of $\xi_{\text{MSE}} = 0.00055$, the proposed construction procedure produced 9 Gaussian RBF units, and the resulting model is listed in Table 3. The MSE values of this 9-unit RBF model over the training and testing sets were

0.000532 and 0.000558, respectively. Fig. 8 depicts the modelling performance for this 9-unit RBF model.

Using a cross validation, it was found that $\sigma^2 = 1.69$, $\varepsilon = 0.02$ and $C = 4$ were appropriate for the SVM algorithm, and the algorithm produced a RBF network with 89 SVs. The MSE values for this 89-unit RBF model were 0.000495 and 0.000524 over the training and test sets, respectively. Fig. 9 shows the modelling performance of this 89-unit RBF model. It can be seen that the proposed sparse construction algorithm is capable of producing a much sparser RBF network with the same excellent generalisation performance, in comparison with the SVM algorithm.

Example 4. This example constructed a model for the gas furnace data set (Series J in [36]). The data set contained 296 pairs of input-output points, where the input $u(t)$ was the coded input gas feed rate and the output $y(t)$ represented CO₂ concentration from the gas furnace. The input-output data are depicted in Fig. 10. The training data set was constructed with $y_i = y(i)$ and $\mathbf{x}_i = [y(i-1) y(i-2) y(i-3) u(i-1) u(i-2) u(i-3)]^T$ for $i = 4, 5, \dots, 296$. With the modelling accuracy set to $\xi_{\text{MSE}} = 0.054$, the proposed RBF construction procedure selected 18 Gaussian RBF regressors, as listed in Table 4. The modelling performance of this constructed 18-unit RBF model are shown in Fig. 11. The modelling MSE value was 0.0538.

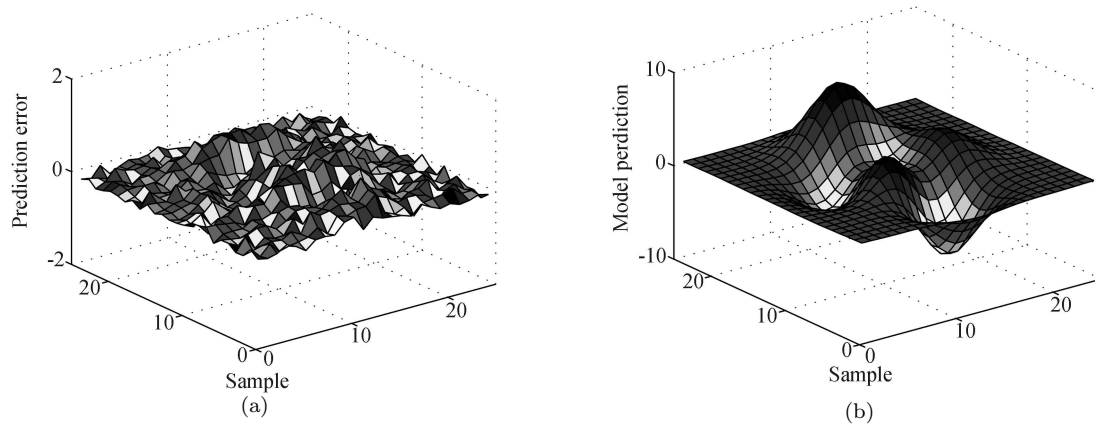


Fig. 6 SVM construction of the RBF network for Example 2: (a) the modelling errors for the constructed 42-unit RBF model, and (c) the model outputs of the constructed 42-unit RBF model

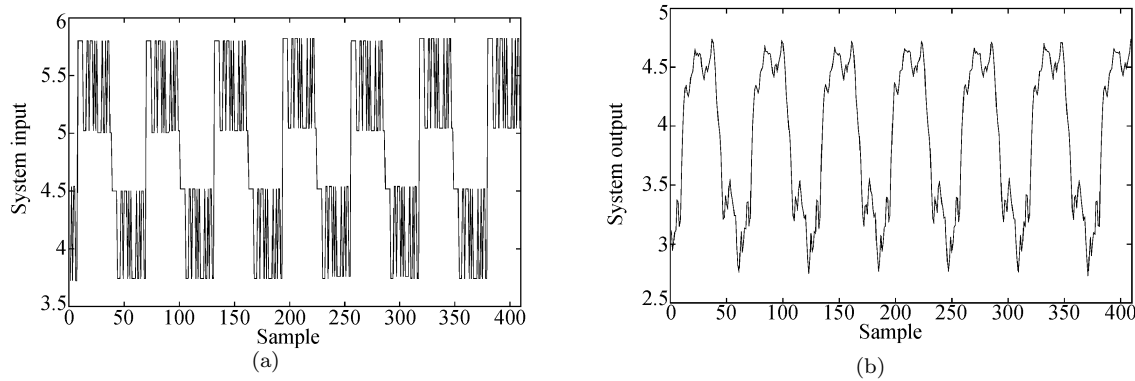


Fig. 7 The engine data set: (a) input $u(t)$ and (b) output $y(t)$

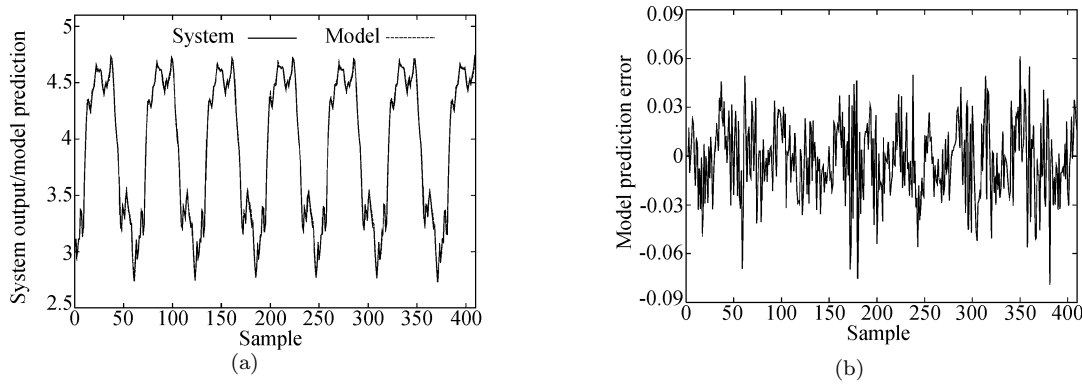


Fig. 8 Incremental construction of the RBF network for Example 3, the engine data set: (a) model prediction $\hat{y}(t)$ (dashed) superimposed on system output $y(t)$ (solid), and (b) model prediction error $e(t) = y(t) - \hat{y}(t)$. The proposed sparse construction procedure constructed a 9-unit RBF model

Table 3 Incremental construction of the RBF network for Example 3, the engine data set

step k	centre vector μ_k			covariance matrix Σ_k			weight w_k	MSE	$MSE_k \times 100$
0	-			-			-	1558.9	
1	5.000 2	5.535 1	5.540 9	6.651 4	18.615 3	15.654 7	4.637 6	0.480 5	
2	3.732 3	4.907 9	5.120 9	0.160 0	0.396 6	0.309 4	-0.282 7	0.351 5	
3	5.232 9	6.012 9	6.320 0	0.160 0	0.799 7	1.747 3	0.471 3	0.180 7	
4	2.247 2	4.193 5	4.473 9	0.160 0	0.160 0	0.216 2	0.942 4	0.107 6	
5	3.834 6	5.231 3	5.363 1	0.160 0	0.210 3	64.000 0	-0.093 9	0.076 9	
6	5.232 9	3.224 5	6.320 0	0.160 0	4.294 5	64.000 0	0.084 7	0.069 8	
7	2.247 2	6.320 0	6.320 0	0.160 0	0.842 7	1.272 6	1.445 0	0.063 6	
8	4.298 9	3.224 5	3.224 5	0.160 0	0.160 0	0.160 0	0.800 1	0.058 5	
9	3.548 0	3.837 2	6.302 3	0.160 0	0.160 0	0.250 6	-14.492 2	0.053 2	

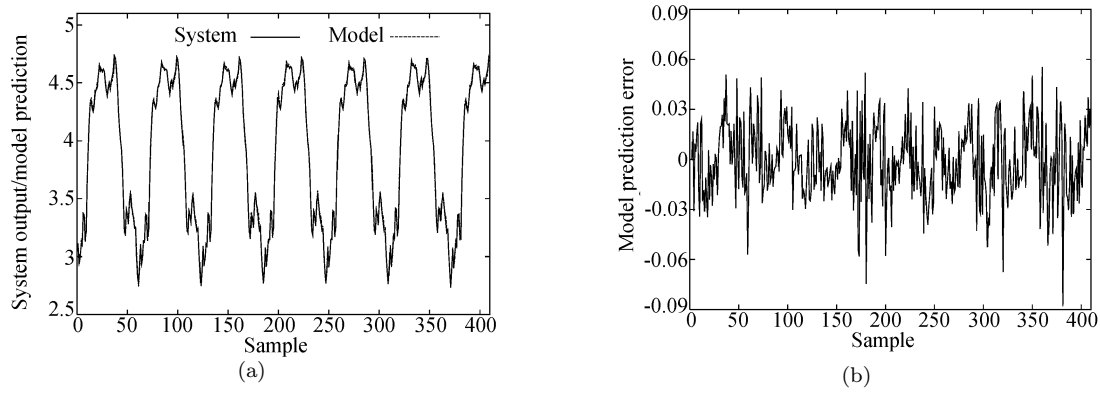


Fig. 9 SVM construction of the RBF network for Example 3, the engine data set: (a) model prediction $\hat{y}(t)$ (dashed) superimposed on system output $y(t)$ (solid), and (b) model prediction error $e(t) = y(t) - \hat{y}(t)$. The SVM algorithm constructed a 89-unit RBF model

Table 4 Incremental construction of the RBF network for Example 4, the gas furnace data set

k	centre vector μ_k						weight w_k	MSE $_k$
	diagonal covariance matrix Σ_k							
0	-						-	2.8443×10^3
1	61.0000	61.0000	50.9525	3.3340	-3.2160	-3.2160	59.276	0.8268
	6.0783	25.0000	25.0000	16.6549	2.4340	9.1178		
2	60.2279	45.1000	45.1000	3.3340	3.3340	3.3340	4.0451	0.5470
	0.0217	5.9837	2.8963	23.3208	19.4850	9.1388		
3	46.8987	58.9860	57.1320	-1.2067	-2.4251	0.3321	-71.404	0.2941
	0.0936	0.0762	6.2630	0.0322	6.7020	0.0262		
4	45.1000	45.1000	45.1000	3.3340	-3.2160	3.3340	2.4931	0.1389
	5.7572	0.0476	7.9013	19.0917	12.3381	10.1693		
5	57.9404	51.5699	47.5973	-3.2160	3.3340	3.3340	29.190	0.1138
	0.0293	22.1583	0.0476	11.1085	8.9510	18.3801		
6	58.0935	45.1000	45.1000	3.3340	3.3340	1.9810	58.659	0.1065
	0.0101	0.2753	3.8145	24.6911	0.0512	0.0164		
7	56.3757	49.5204	60.1222	2.2450	2.5864	-1.2469	21.472	0.0972
	25.0000	0.0100	25.0000	0.0100	0.0123	0.0100		
8	45.1000	45.1000	51.6922	-0.0144	-3.2160	3.3340	-5.2669	0.0804
	0.1052	1.5136	0.0501	0.0100	0.0550	11.9954		
9	46.0303	58.7537	54.9481	-2.6996	1.5488	-0.2903	-1.4557	0.0760
	25.0000	0.0385	25.0000	0.0100	25.0000	0.0100		
10	58.7624	55.1555	54.8745	2.5233	2.2267	-1.6638	46.918	0.0700
	0.0100	0.0100	0.1178	0.0113	25.0000	24.5476		
11	53.9952	54.8025	52.8236	-1.7173	3.1207	2.1370	-1.7379	0.0675
	25.0000	0.0100	0.0100	25.0000	25.0000	0.0100		
12	45.1000	47.1594	61.0000	3.3340	3.3340	3.3340	2.1679	0.0631
	25.0000	0.0620	25.0000	0.0100	0.0100	25.0000		
13	54.7481	48.7561	50.0140	-0.9630	0.8991	1.1693	48.259	0.0589
	0.0100	0.0100	0.0100	15.7145	0.0100	25.0000		
14	46.2857	46.4039	54.9041	-2.7555	-2.7218	2.6338	1.5930	0.0578
	25.0000	0.0992	0.0100	25.0000	25.0000	25.0000		
15	53.5265	49.5718	48.3381	1.9158	0.1292	1.7822	7.4529	0.0570
	0.0100	0.0100	25.0000	0.0100	0.0100	0.0100		
16	55.4362	54.9274	57.2423	0.1605	0.1057	1.8326	-0.9918	0.0550
	0.0100	0.0100	25.0000	0.0100	0.0100	0.0100		
17	53.7756	51.3648	51.6104	0.7165	-0.1010	0.6642	-0.0845	0.0542
	25.0000	0.0100	0.0100	25.0000	25.0000	25.0000		
18	45.1000	45.1000	45.1000	-3.2160	-3.2160	-3.2160	0.2511	0.0538
	1.0087	17.3402	25.0000	10.6542	25.0000	0.0100		

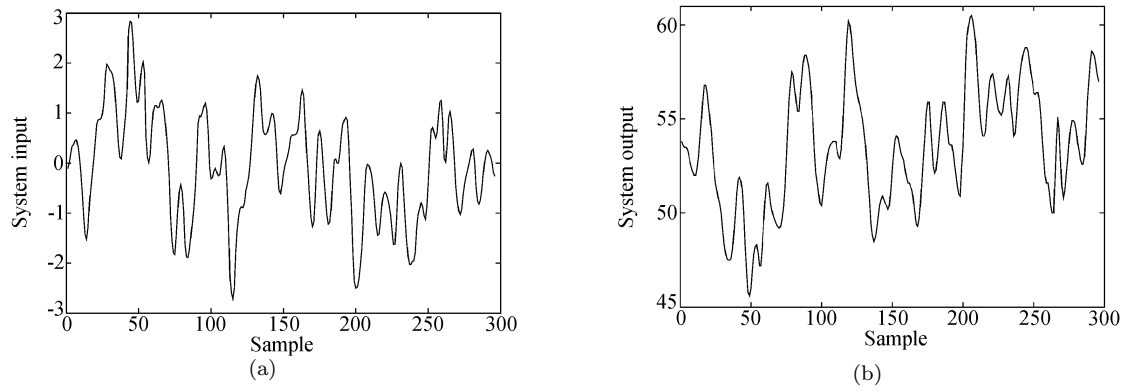


Fig. 10 The gas furnace data set: (a) input $u(t)$ and (b) output $y(t)$

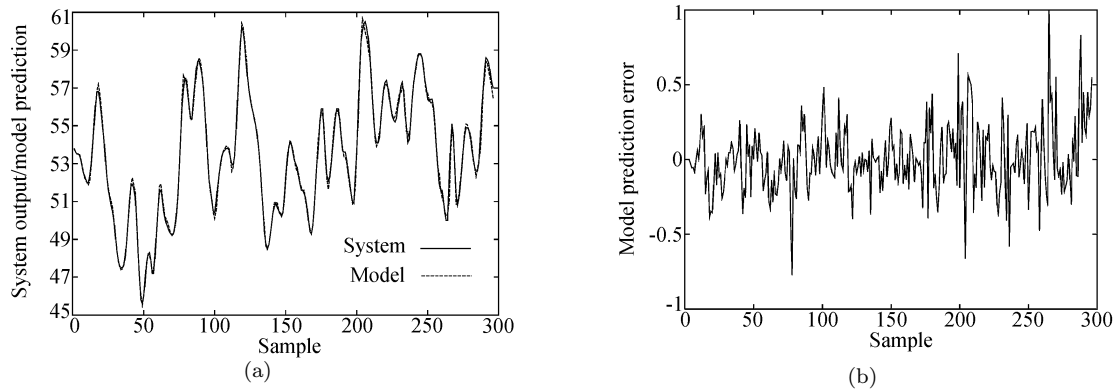


Fig. 11 Incremental construction of the RBF network for Example 4, the gas furnace data set: (a) model prediction $\hat{y}(t)$ (dashed) superimposed on system output $y(t)$ (solid), and (b) model prediction error $e(t) = y(t) - \hat{y}(t)$. The proposed sparse construction procedure constructed a 18-unit RBF model

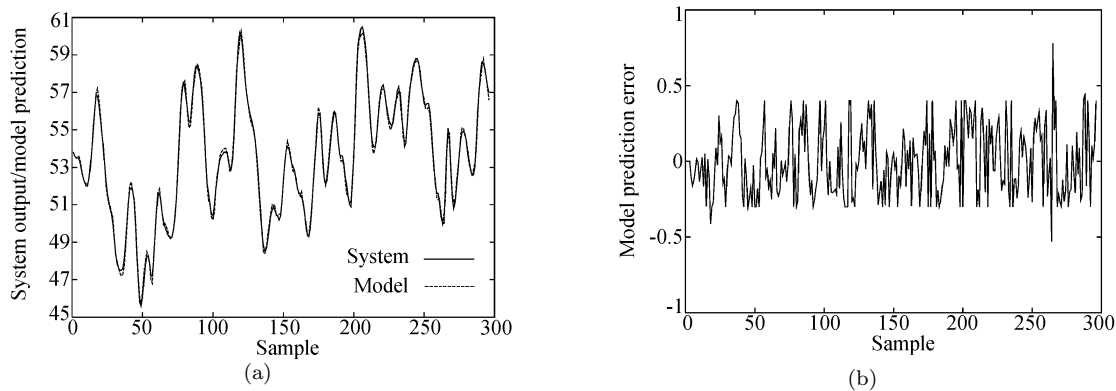


Fig. 12 SVM construction of the RBF network for Example 4, the gas furnace data set: (a) model prediction $\hat{y}(t)$ (dashed) superimposed on system output $y(t)$ (solid), and (b) model prediction error $e(t) = y(t) - \hat{y}(t)$. The SVM algorithm constructed a 61-unit RBF model

Next, the SVM algorithm was applied to this example. The appropriate values for the SVM's learning parameters were found to be $\sigma^2 = 20$, $\varepsilon = 0.35$ and $C = 300$. The RBF model constructed by the SVM algorithm contained 61 SVs with the modelling MSE value of 0.0520. Fig. 12 depicts the modelling perfor-

mance for the constructed 61-unit RBF model. Again, it is seen that both the proposed RBF construction method and SVM algorithm have similar modelling accuracy but the former can produce a much sparser RBF network.

4 Conclusions

A novel technique has been presented to construct sparse RBF network models for regression. Unlike most of the sparse kernel regression methods, which restrict RBF centres to the training input data and use a fixed common variance for all the RBF units, the proposed technique can tune the centre vector and diagonal covariance matrix of individual RBF unit to best fit the training data based on the correlation between the RBF regressor and the training data. This technique thus provides enhanced modelling capability. An efficient repeated weighted boosting search optimisation method has been employed based on the correlation criterion to append RBF units one by one in an incremental construction procedure. Using the standard SVM method as the benchmark, experiments have been conducted using several regression modelling examples, and it has been shown that the proposed RBF network construction method is capable of producing much sparser model representations with the same excellent generalisation performance in comparison with the SVM algorithm.

References

- [1] S. Chen, S. A. Billings, W. Luo. Orthogonal Least Squares Methods and Their Application to Non-Linear System Identification. *International Journal of Control*, vol. 50, no. 5, pp. 1873–1896, 1989.
- [2] S. Chen, C. F. N. Cowan, P. M. Grant. Orthogonal Least Squares Learning Algorithm for Radial Basis Function Networks. *IEEE Transactions on Neural Networks*, vol. 2, no. 2, pp. 302–309, 1991.
- [3] M. J. L. Orr. Regularization in the Selection of Radial Basis Function Centres. *Neural Computation*, vol. 7, no. 3, pp. 606–623, 1995.
- [4] S. Chen, E. S. Chng, K. Alkadhimi. Regularised Orthogonal Least Squares Algorithm for Constructing Radial Basis Function Networks. *International Journal of Control*, vol. 64, no. 5, pp. 829–837, 1996.
- [5] S. Chen, Y. Wu, B. L. Luk. Combined Genetic Algorithm Optimisation and Regularised Orthogonal Least Squares Learning for Radial Basis Function Networks. *IEEE Transactions on Neural Networks*, vol. 10, no. 5, pp. 1239–1243, 1999.
- [6] S. Chen, X. Hong, C. J. Harris. Sparse Kernel Regression Modelling Using Combined Locally Regularized Orthogonal Least Squares and D-Optimality Experimental Design. *IEEE Transactions on Automatic Control*, vol. 48, no. 6, pp. 1029–1036, 2003.
- [7] S. Chen, X. Hong, C. J. Harris, P. M. Sharkey. Sparse Modelling Using Orthogonal Forward Regression with PRESS Statistic and Regularization. *IEEE Transactions on Systems, Man and Cybernetics, Part B*, vol. 34, no. 2, pp. 898–911, 2004.
- [8] V. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, New York, 1995.
- [9] S. Gunn. Support Vector Machines for Classification and Regression. Technical Report, Information: Signals, Images, Systems (ISIS) Research Group, Department of Electronics and Computer Science, University of Southampton, UK, May 1998.
- [10] S. S. Chen, D. L. Donoho, M. A. Saunders. Atomic Decomposition by Basis Pursuit. *SIAM Review*, vol. 43, no. 1, pp. 129–159, 2001.
- [11] M. E. Tipping. Sparse Bayesian Learning and the Relevance Vector Machine. *Journal of Machine Learning Research*, vol. 1, pp. 211–244, 2001.
- [12] B. Schölkopf, A. J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, MA, 2002.
- [13] P. Vincent, Y. Bengio. Kernel Matching Pursuit. *Machine Learning*, vol. 48, no. 1, pp. 165–187, 2002.
- [14] G. R. G. Lanckriet, N. Cristianini, P. Bartlett, L. E. Ghaoui, M. I. Jordan. Learning the Kernel Matrix with Semidefinite Programming. *Journal of Machine Learning Research*, vol. 5, pp. 27–72, 2004.
- [15] T. Duddy. *Prior Knowledge for Time Series Modeling*. Ph.D. These, Department of Electronics and Computer Science, University of Southampton, U.K., 2000.
- [16] M. Stone. Cross Validation Choice and Assessment of Statistical Predictions. *Journal of Royal Statistics Society, Series B*, vol. 36, pp. 117–147, 1974.
- [17] R. H. Myers. *Classical and Modern Regression with Applications*. 2nd ed., PWS-KENT, Boston, 1990.
- [18] S. Lee, R. M. Kil. A Gaussian Potential Function Network with Hierarchically Self-Organizing Learning. *Neural Networks*, vol. 4, no. 2, pp. 207–224, 1991.
- [19] S. Chen, X. Hong, C. J. Harris, X. X. Wang. Identification of Nonlinear Systems Using Generalized Kernel Models. *IEEE Transactions on Control Systems Technology*, vol. 13, no. 3, pp. 401–411, 2005.
- [20] G. B. Huang, P. Saratchandran, N. Sundararajan. A Generalized Growing and Pruning RBF (GGAP-RBF) Neural Network for Function Approximation. *IEEE Transactions on Neural Networks*, vol. 16, no. 1, pp. 57–67, 2005.
- [21] J. Moody, C. J. Darken. Fast Learning in Networks of Locally-Tuned Processing Units. *Neural Computation*, vol. 1, pp. 281–294, 1989.
- [22] S. Chen, S. A. Billings, P. M. Grant. Recursive Hybrid Algorithm for Non-Linear System Identification Using Radial Basis Function Networks. *International Journal of Control*, vol. 55, pp. 1051–1070, 1992.
- [23] S. Chen. Nonlinear Time Series Modelling and Prediction Using Gaussian RBF Networks with Enhanced Clustering and RLS Learning. *Electronics Letters*, vol. 31, no. 2, pp. 117–118, 1995.
- [24] P. E. An, M. Brown, S. Chen, C. J. Harris. Comparative Aspects of Neural Network Algorithms for On-Line Modelling of Dynamic Processes. In *Proceedings of the Institution of Mechanical Engineers, Part I, Journal of Systems and Control Engineering*, vol. 207, pp. 223–241, 1993.
- [25] B. A. Whitehead, T. D. Choate. Evolving Space-Filling Curves to Distribute Radial Basis Functions Over an Input Space. *IEEE Transactions on Neural Networks*, vol. 5, no. 1, pp. 15–23, 1994.
- [26] S. Chen, X. X. Wang, C. J. Harris. Experiments with Repeating Weighted Boosting Search for Optimization in Signal Processing Applications. *IEEE Transactions on Systems, Man and Cybernetics, Part B*, vol. 35, no. 4, pp. 682–693, 2005.
- [27] S. E. Fahlman, C. Lebiere. The Cascade-Correlation Learning Architecture. *Neural Information Processing Systems 2*, D.S. Touretzky, Ed., Morgan-Kaufmann, San Francisco, CA, pp. 524–532, 1990.
- [28] H. Akaike. A New Look at the Statistical Model Identification. *IEEE Transactions on Automatic Control*, vol. AC-19, pp. 716–723, 1974.
- [29] I. J. Leontaritis, S. A. Billings. Model Selection and Validation Methods for Non-Linear Systems. *International Journal*

- of *Control*, vol. 45, no. 1, pp. 311–341, 1987.
- [30] X. Hong, C. J. Harris. Nonlinear Model Structure Design and Construction Using Orthogonal Least Squares and D-Optimality Design. *IEEE Transactions on Neural Networks*, vol. 13, no. 5, pp. 1245–1250, 2002.
- [31] D. E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison Wesley, Reading, MA, 1989.
- [32] K. F. Man, K. S. Tang, S. Kwong. *Genetic Algorithms: Concepts and Design*. Springer-Verlag, London, 1998.
- [33] L. Ingber. Simulated Annealing: Practice Versus Theory. *Mathematical and Computer Modeling*, vol. 18, no. 11, pp. 29–57, 1993.
- [34] S. Chen, B. L. Luk. Adaptive Simulated Annealing for Optimization in Signal Processing Applications. *Signal Processing*, vol. 79, no. 1, pp. 117–128, 1999.
- [35] S. A. Billings, S. Chen, R. J. Backhouse. The Identification of Linear and Non-Linear Models of a Turbocharged Automotive Diesel Engine. *Mechanical Systems and Signal Processing*, vol. 3, no. 2, pp. 123–142, 1989.
- [36] G. E. P. Box, G. M. Jenkins. *Time Series Analysis, Forecasting and Control*. Holden Day Inc., 1976.



Xun-xian Wang received his Ph.D. degree in the control theory and application field from Tsinghua University, Beijing, China, in July 1999.

From August 1999 to August 2001, he was a postdoctoral researcher in the State Key Laboratory of Intelligent Technology and Systems, Beijing, China. From September 2001 to December 2004, he was a research fellow at the University of Portsmouth, Portsmouth, U.K. From January 2005, he has been a research fellow at Neural Computing Research Group, Aston University, Birmingham, U.K.

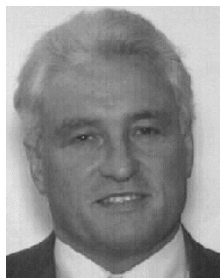
His main research interests include machine learning and neural networks, control theory and systems as well as robotics.



Sheng Chen received his Ph.D. degree in control engineering from the City University, London, U.K., in 1986. He joined the School of Electronics and Computer Science, University of Southampton, Southampton, U.K., in September 1999. He previously held research and academic appointments at the University of Sheffield, the University of Edinburgh, and the University of Portsmouth, all in U.K.

He has published over 260 research papers. His research works include wireless communications, machine learning and neural networks, finite-precision digital controller design, and evolutionary computation methods.

Professor Chen holds a higher doctorate degree, D.Sc, from the University of Southampton. In the database of the world's most highly cited researchers, compiled by Institute for Scientific Information (ISI) of the USA, Dr. Chen is on the list of the highly cited researchers in the engineering category.



Chris Harris received his Ph.D. degree from the University of Southampton, Southampton, U.K.

He previously held appointments at the University of Hull, the UMIST, the University of Oxford, and the University of Cranfield, all in U.K., as well as being employed by the U.K. Ministry of Defence. He returned to the University of Southampton as the Lucas Professor of Aerospace Systems Engineering in 1987 to establish the Advanced Systems Research Group and, more recently, Image, Speech and Intelligent Systems Group.

He has authored and co-authored 12 research books and over 400 research papers, and he is the associate editor of numerous international journals. His research interests include the general area of intelligent and adaptive systems theory and its application to intelligent autonomous systems such as autonomous vehicles, management infrastructures such as command & control, intelligent control, and estimation of dynamic processes, multi-sensor data fusion, and systems integration.

Professor Harris holds a higher doctorate degree, D.Sc, from the University of Southampton. Dr. Harris was elected to the Royal Academy of Engineering in 1996, was awarded the IEE Senior Achievement medal in 1998 for his work in autonomous systems, and the highest international award in IEE, the IEE Faraday medal, in 2001 for his work in intelligent control and neurofuzzy systems.