Neurocomputing 74 (2011) 1564–1571

Contents lists available at ScienceDirect

Neurocomputing

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

Grey-box radial basis function modelling

Sheng Chen [a,*], Xia Hong [b], Chris J. Harris [a]

[a] School of Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, UK
[b] School of Systems Engineering, University of Reading, Reading RG6 6AY, UK

ARTICLE INFO

Article history:
Received 28 September 2010
Received in revised form
23 December 2010
Accepted 2 January 2011
Communicated by K. Li
Available online 21 March 2011

Keywords:
Data modelling
Radial basis function network
Black-box model
Grey-box model
Orthogonal least squares algorithm
Symmetry
Boundary value constraint

ABSTRACT

A fundamental principle in data modelling is to incorporate available *a priori* information regarding the underlying data generating mechanism into the modelling process. We adopt this principle and consider grey-box radial basis function (RBF) modelling capable of incorporating prior knowledge. Specifically, we show how to explicitly incorporate the two types of prior knowledge: (i) the underlying data generating mechanism exhibits known symmetric property, and (ii) the underlying process obeys a set of given boundary value constraints. The class of efficient orthogonal least squares regression algorithms can readily be applied without any modification to construct parsimonious grey-box RBF models with enhanced generalisation capability.

1. Introduction

The radial basis function (RBF) network has found wide-ranging applications in diverse fields of engineering [1–17], and the class of orthogonal least squares (OLS) regression algorithms [18–22] offers powerful and efficient tools for constructing parsimonious RBF models that generalise well. This approach is equally applicable to the supervised regression [18–22] and classification [23–25] as well as the unsupervised probability density function estimation [26–28]. Like many other data modelling approaches, the RBF model constitutes a black-box data modelling approach. Adopting a black-box modelling is appropriate if no *a priori* information exists regarding the underlying data generating mechanism. However, if there are known prior knowledge concerning the underlying process, they should be incorporated into the model structure explicitly. The use of prior knowledge in data modelling often leads to enhanced modelling performance. A general discussion on learning from known prior knowledge or hints is given in [29]. A few works have exploited the symmetric properties of some underlying systems in regression applications [30,31] as well as in classification problems [32,33].

System identification has a long history of investigating grey-box based techniques, and some studies on how to incorporating *a priori* system knowledge into the model structure can be found in [34–39]. For the linear system identification, the work [34] has shown how to translate crucial physical knowledge, such as process stability and sign of stationary gains, into linear inequality constraints on the black-box model to yield the grey-box model class in which a Bayesian approach is adopted to associate the physical knowledge with a prior distribution. The authors of [36] have proposed an approach which can potentially incorporate the system knowledge naturally into the linear-in-the-parameter nonlinear black-box model. They argue that, instead of a black-box polynomial expansion, various nonlinear functions or bases can be adopted to form an extended model set and the choices of nonlinear bases may be determined from physical knowledge of the system to be modelled. The study [37] has emphasised that for practical nonlinear engineering systems, some of the underlying physical parameters are usually known *a priori* and, therefore, a grey-box nonlinear model should be adopted to explicitly utilise the *a priori* system knowledge. The works [38,39] have further refined the concept of the extended model set [36] and have proposed a novel eng-genes framework which chooses the activation functions of neural network nodes or nonlinear bases to reflect physical reality of the process to be modelled. It should be emphasised, however, that there does not exist a generic grey-box model which can represent any *a priori* system knowledge.

How to incorporating prior knowledge to form a grey-box model is highly problem dependent and is really an art. But there exist some desired objectives in using a grey-box model. Firstly, by

* Corresponding author.
E-mail addresses: sqc@ecs.soton.ac.uk (S. Chen), x.hong@reading.ac.uk (X. Hong), cjh@ecs.soton.ac.uk (C.J. Harris).

0925-2312/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.neucom.2011.01.023

incorporating *a priori* information regarding the underlying process to be modelled, better generalisation performance should be achieved. Secondly, a grey-box modelling should not result in an increased computational complexity. For example, when developing a grey-box RBF model, it is highly desirable that the existing learning algorithms for the black-box RBF model can readily be used, and one is not forced to derive new learning algorithms. In this contribution, we specifically consider two types of *a priori* information. In the first type of data modelling problems, the underlying data generating mechanism exhibits a known symmetric property and we introduce the symmetric RBF (SRBF) model that guarantees to possess the known symmetry. For the second type of applications, the underlying process obeys a set of the given boundary value constraints (BVCs) and we adopt the novel BVC-RBF structure which automatically meets the given BVCs. All the learning algorithms originally derived for the black-box RBF model can be applied to these two grey-box RBF models without the need for any modification. In particular, the class of OLS learning algorithms [18–22] provides efficient means of building parsimonious grey-box RBF models with improved generalisation performance.

The remainder of this contribution is structured as follows. Section 2 summarises the black-box RBF modelling based on the class of efficient OLS learning algorithms. The two grey-box RBF models are derived in Sections 3 and 4, respectively, by incorporating *a priori* knowledge of symmetric property and a given set of BVCs. Our conclusions are offered in Section 5.

2. Black-Box RBF modelling

Give the training data set $D_K=\{\mathbf{x}(k),y(k)\}_{k=1}^K$, where $\mathbf{x}(k)=[x_1(k)\cdots x_m(k)]^T\in\mathsf{R}^m$ is the input vector and $y(k)\in\mathsf{R}$ is the desired output for $\mathbf{x}(k)$. The data is generated by the unknown nonlinear data generating mechanism with the nonlinear mapping $f:\mathsf{R}^m\to\mathsf{R}$ as

$$y(k)=f(\mathbf{x}(k))+\varepsilon(k) \tag{1}$$

where $\varepsilon(k)$ is the observation noise. The RBF model of the form

$$\hat{y}^{(M)}(k)=\hat{f}^{(M)}(\mathbf{x}(k))=\sum_{i=1}^{M}\theta_i p_i(\mathbf{x}(k);\sigma) \tag{2}$$

is constructed from the training data D_K to realise the underlying data generating mechanism $f:\mathsf{R}^m\to\mathsf{R}$, where M is the number of RBF units, and each RBF basis

$$p_i(\mathbf{x};\sigma)=\varphi(\|\mathbf{x}-\mathbf{c}_i\|/\sigma) \tag{3}$$

is specified by its centre vector $\mathbf{c}_i\in\mathsf{R}^m$, RBF variance σ^2 and the chosen basis function $\varphi(\bullet)$. This is a black-box modelling approach, as no prior knowledge regarding f is required and everything is learnt from the data, which is inherently stochastic due to the observation noise. The class of efficient OLS learning algorithms [18–22] have been developed to construct the RBF model from the training data D_K.

Use every data $\mathbf{x}(k)$ as a candidate RBF centre and assume that a common RBF variance σ^2 is obtained separately via cross validation. Then the resulting K-term RBF model over the training data $(\mathbf{x}(k),y(k))\in D_K$ can be expressed as

$$y(k)=(\mathbf{p}^{(K)}(k))^T\boldsymbol{\theta}_K+\varepsilon^{(K)}(k) \tag{4}$$

where $\varepsilon^{(K)}(k)=y(k)-\hat{y}^{(K)}(k)$ is the K-term modelling error, $\boldsymbol{\theta}_K=[\theta_1\cdots\theta_K]^T$ is the RBF weight vector, and $\mathbf{p}^{(K)}(k)=[p_1(k)\cdots p_K(k)]^T$ with

$$p_i(k)=\varphi(\|\mathbf{x}(k)-\mathbf{x}(i)\|/\sigma),\quad 1\le i\le K \tag{5}$$

Furthermore, the model (4) over the training data set D_K can be expressed as

$$\mathbf{y}=\mathbf{P}_K\boldsymbol{\theta}_K+\boldsymbol{\varepsilon}^{(K)} \tag{6}$$

by introducing the notations $\mathbf{y}=[y(1)\cdots y(K)]^T$, $\boldsymbol{\varepsilon}^{(K)}=[\varepsilon^{(K)}(1)\cdots\varepsilon^{(K)}(K)]^T$ and

$$\mathbf{P}_K=[\mathbf{p}_1\cdots\mathbf{p}_K] \tag{7}$$

with $\mathbf{p}_i=[p_i(1)\cdots p_i(K)]^T$. Note that $\mathbf{p}_k$ is the kth column of $\mathbf{P}_K$, while $(\mathbf{p}^{(K)}(k))^T$ denotes the kth row of $\mathbf{P}_K$.

Let an orthogonal decomposition of the regression matrix $\mathbf{P}_K$ be $\mathbf{P}_K=\mathbf{W}_K\mathbf{A}_K$ with

$$\mathbf{A}_K=\begin{bmatrix}1 & \alpha_{1,2} & \cdots & \alpha_{1,K}\\ 0 & 1 & \ddots & \vdots\\ \vdots & \ddots & \ddots & \alpha_{K-1,K}\\ 0 & \cdots & 0 & 1\end{bmatrix} \tag{8}$$

and the orthogonal regression matrix

$$\mathbf{W}_K=[\mathbf{w}_1\cdots\mathbf{w}_K] \tag{9}$$

that satisfies $\mathbf{w}_i^T\mathbf{w}_j=0$ if $i\ne j$. Then the regression model (6) can be written equivalently as

$$\mathbf{y}=\mathbf{W}_K\mathbf{g}_K+\boldsymbol{\varepsilon}^{(K)} \tag{10}$$

where the weight vector $\mathbf{g}_K$ satisfies the relationship $\mathbf{A}_K\boldsymbol{\theta}_K=\mathbf{g}_K$. Similar to (4), $y(k)$ can be modelled by

$$y(k)=(\mathbf{w}^{(K)}(k))^T\mathbf{g}_K+\varepsilon^{(K)}(k) \tag{11}$$

where $\mathbf{w}^{(K)}(k)=[w_1(k)\cdots w_K(k)]^T$ is the kth row of $\mathbf{W}_K$.

The OLS forward selection procedure chooses model terms one by one from the full K-term candidate set. Specifically, after the $n-1$-th stage of the subset selection, the selected model contains $n-1$ model columns while the candidate pool contains the remaining $K-n+1$ candidate columns, as illustrated in the following:

$$\left[\overbrace{\mathbf{w}_1\mathbf{w}_2\cdots\mathbf{w}_{n-1}}^{\text{selected model terms}}\ \vdots\ \overbrace{\mathbf{p}_n\mathbf{p}_{n+1}\cdots\mathbf{p}_K}^{\text{candidate pool}}\right]$$

At the n-th stage of the subset selection, one model term is selected from the candidate pool as the nth selected model term to add to the selected subset model. This selected model term $\mathbf{w}_n$ should maximally improve the modelling performance of the n-term subset model according to some specified criterion.

2.1. D-optimality enhanced ROLS algorithm

In the D-optimality enhanced regularised OLS (ROLS) algorithm, the criterion for the subset model selection is the combined regularised training mean square error (MSE) and the D-optimality criterion [21] defined by

$$J_{\mathrm{CRD}}(\mathbf{g}_K,\boldsymbol{\lambda}_K,\beta)=(\boldsymbol{\varepsilon}^{(K)})^T\boldsymbol{\varepsilon}^{(K)}+\mathbf{g}_K^T\boldsymbol{\Lambda}_K\mathbf{g}_K)+\beta\sum_{n=1}^{K}-\log(\mathbf{w}_n^T\mathbf{w}_n) \tag{12}$$

where $\boldsymbol{\lambda}_K=[\lambda_1\cdots\lambda_K]^T$ is the regularisation parameter vector, $\boldsymbol{\Lambda}_K=\mathrm{diag}\{\lambda_1,\cdots,\lambda_K\}$, and β is the D-optimality weighting. The local regularisation term $\mathbf{g}_K^T\boldsymbol{\Lambda}_K\mathbf{g}_K$ in the criterion (12) enhances the generalisation and sparseness of the selected model [40], while the D-optimality criterion, the last term in $J_{\mathrm{CRD}}(\mathbf{g}_K,\boldsymbol{\lambda}_K,\beta)$, prevents the selection of an oversized ill-posed model and reduces the parameter estimate variances [21,41].

Denote the value of J_{CRD} for the selected $n-1$-term subset model as $J_{\mathrm{CRD}}^{(n-1)}$. Then at the n-th stage of selection, the selected model term is the one that minimises the combined criterion

$$J_{\mathrm{CRD}}^{(n)}=J_{\mathrm{CRD}}^{(n-1)}-g_n^2(\mathbf{w}_n^T\mathbf{w}_n+\lambda_n)-\beta\log(\mathbf{w}_n^T\mathbf{w}_n) \tag{13}$$

As shown in [21], with an appropriate chosen value for β, there exists an "optimal" subset model size $M \ll K$ such that: for $n \le M$, the criterion $J_{\rm CRD}^{(n)}$ decreases as n increases, while

$$J_{\rm CRD}^{(M)} < J_{\rm CRD}^{(M+1)} \tag{14}$$

Thus, the subset model selection is automatically terminated, yielding an M-term RBF model. The regularisation parameters can be updated using the evidence procedure [21,22,40]. The detailed algorithm can be found in [21], and it will not be repeated here. In particular, when no regularisation is employed, i.e. $\lambda_n = 0$ for all n, this algorithm reduces to the D-optimality assisted OLS algorithm presented in [42].

2.2. ROLS algorithm based on LOO statistics

It is highly desirable to select model terms by directly optimising the model generalisation performance, instead of the training performance. Model generalisation can be evaluated by the test performance on the data not used in training the model, and a commonly used cross validation method is the leave-one-out (LOO) cross validation [43,44]. The idea of LOO cross validation is as follow. Remove the kth data from the training set $D_K = \{\mathbf{x}(k), y(k)\}_{k=1}^K$, and use the remaining $K-1$ data $D_K \backslash (\mathbf{x}(k), y(k))$ to identify the n-term model, which is denoted by $\hat{y}^{(n,-k)}$. The test error on the single data point not used in training is

$$\varepsilon^{(n,-k)}(k) = y(k) - \hat{y}^{(n,-k)}(k) \tag{15}$$

Repeating the procedure for each k leads to the LOO test MSE for the n-term model

$$J_{\rm LOO}^{(n)} = \frac{1}{K}\sum_{k=1}^{K} (\varepsilon^{(n,-k)}(k))^2 \tag{16}$$

which is a generalisation measure for the model $\hat{y}^{(n)}$ identified using the whole D_K [43,44]. For the linear-in-the-weights models, which the model (6) is, the above steps of the LOO cross validation are *virtual*, as the LOO test errors can be generated, without actually sequentially splitting the training data set and repeatedly estimating the associated models, by applying the Sherman–Morrison–Woodbury theorem [43,44].

In particular, the use of the equivalent orthogonal model (10) leads to an efficient computation of the LOO test MSE [22,45]. This is because the LOO error can be expressed as [44]

$$\varepsilon^{(n,-k)}(k) = \frac{\varepsilon^{(n)}(k)}{\eta^{(n)}(k)} \tag{17}$$

where the n-term modelling error $\varepsilon^{(n)}(k)$ and the associated LOO error weighting $\eta^{(n)}(k)$ can be calculated recursively according to [22,45]

$$\varepsilon^{(n)}(k) = \varepsilon^{(n-1)}(k) - w_n(k) g_n \tag{18}$$

$$\eta^{(n)}(k) = \eta^{(n-1)}(k) - \frac{w_n^2(k)}{\mathbf{w}_n^T \mathbf{w}_n + \lambda_n} \tag{19}$$

As shown in [45], the LOO test MSE has the following desired property, namely, there exists an "optimal" subset size $M \ll K$ such that: for $n \le M$, the criterion $J_{\rm LOO}^{(n)}$ decreases as n increases, while

$$J_{\rm LOO}^{(M)} < J_{\rm LOO}^{(M+1)} \tag{20}$$

Thus, the subset model selection is automatically terminated, yielding an M-term RBF model. The detailed ROLS algorithm based on the LOO test MSE can be found in [22].

3. Symmetric RBF modelling

Consider again the training data set $D_K = \{\mathbf{x}(k), y(k)\}_{k=1}^K$ that is generated by the underlying system (1). The system mapping $f : \mathrm{R}^m \to \mathrm{R}$ is unknown. However, the system f is known to possess the odd symmetry

$$f(-\mathbf{x}) = -f(\mathbf{x}) \tag{21}$$

This *a priori* information may come from the known physics law governing the system. For example, from physics, the underlying optimal discriminant function or detector for the binary digital signals has this old symmetry [32]. Although we consider the old symmetry in this contribution, the even symmetry can be treated in a similar way. In fact, our approach can be extended to deal with more complex symmetric properties, such as those encountered in the complex-valued digital signal detection [33].

3.1. Symmetric RBF network

Our goal is to construct the RBF model (2) from the data D_K to discover the underlying data generating mechanism f. To the defence of the black-box RBF model with the standard RBF node (3), it has a good learning capability and should be able to approximate the underlying system f well. Thus, $\hat{f}^{(M)}$ learnt from the training data set D_K alone should approximately possess the odd symmetry. However, this is not guaranteed, particularly when the training data D_K is noisy. Since the underlying system is known to possess the old symmetry (21), we would like the model to possess the same old symmetry, namely,

$$\hat{f}^{(M)}(-\mathbf{x}) = -\hat{f}^{(M)}(\mathbf{x}) \tag{22}$$

Furthermore, we would like to exploit the prior knowledge (21) for improving the modelling efficiency as well.

To explicitly incorporate the prior knowledge (21), we adopt the following symmetric RBF (SRBF) node

$$p_i(\mathbf{x};\sigma) = \varphi(\|\mathbf{x}-\mathbf{c}_i\|/\sigma) - \varphi(\|\mathbf{x}+\mathbf{c}_i\|/\sigma) \tag{23}$$

With this symmetric node structure, the prior information is *naturally* incorporated into the model structure and the resulting SRBF model guarantees to have the same odd symmetry as the underlying system. Moreover, this grey-box RBF model with the symmetric node structure (23) has the same regression modelling form as the black-box RBF model discussed in Section 2. Therefore, we do not need to develop any new learning algorithm for this grey-box RBF model. Instead, the class of OLS learning algorithms [18–22] can readily be used to identify a parsimonious SRBF model based on D_K.

3.2. A symmetric modelling example

The system to be identified was given by

$$f(x_1,x_2) = 10\left(\frac{\sin(x_1-5)\sin(x_2-5)}{(x_1-5)(x_2-5)} - \frac{\sin(x_1+5)\sin(x_2+5)}{(x_1+5)(x_2+5)}\right) \tag{24}$$

This system has the odd symmetry and $f(x_1,x_2)$ is plotted in Fig. 1 (a) using a grid of 90 601 points. The training data set D_K contained 961 noisy data points as shown in Fig. 1(b), where the system noise $\varepsilon(k)$ was a white Gaussian noise with variance $\sigma_\varepsilon^2 = 0.16$. The basis function was chosen to be the Gaussian function. The ROLS algorithm based on the LOO test MSE, summarised in Section 2.2 (also see [22]), was used to *automatically* identify both the conventional RBF and SRBF models. The RBF variance $\sigma^2 = 8.0$ was determined separately using cross validation. A separate test data set of $K_{\rm test} = 961$ noisy data points was also generated to compute the test MSE according to

$$\mathrm{MSE} = E\left[(y(k)-\hat{y}^{(M)}(k))^2\right] = \frac{1}{K_{\rm test}}\sum_{k=1}^{K_{\rm test}} (y(k)-\hat{y}^{(M)}(k))^2 \tag{25}$$

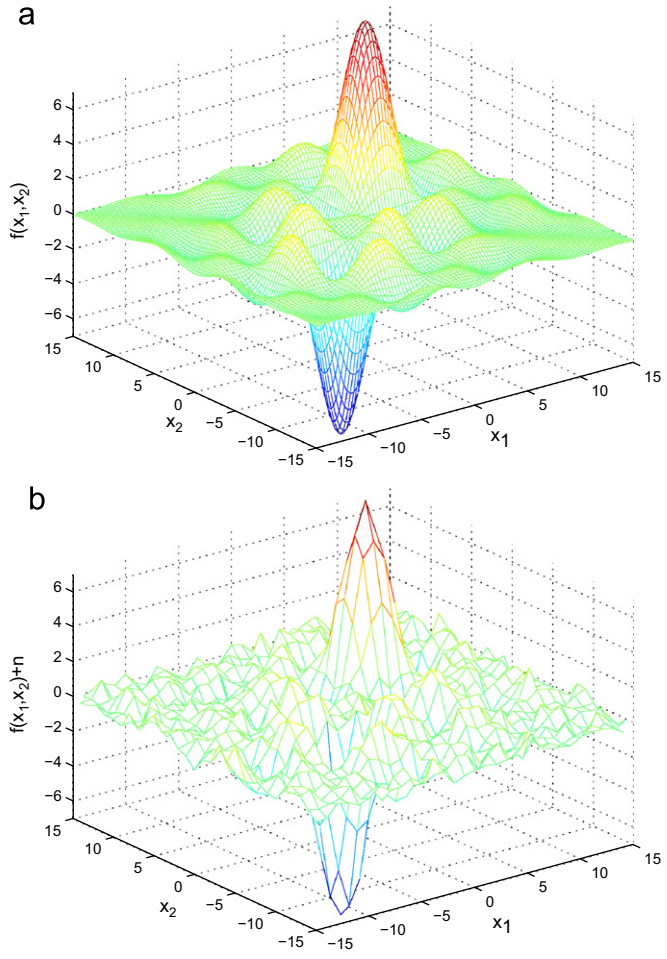

Fig. 1. (a) The underlying symmetric function $f(x_1,x_2)$ shown on the grid of 90 601 points, and (b) the 961 noisy training data points.

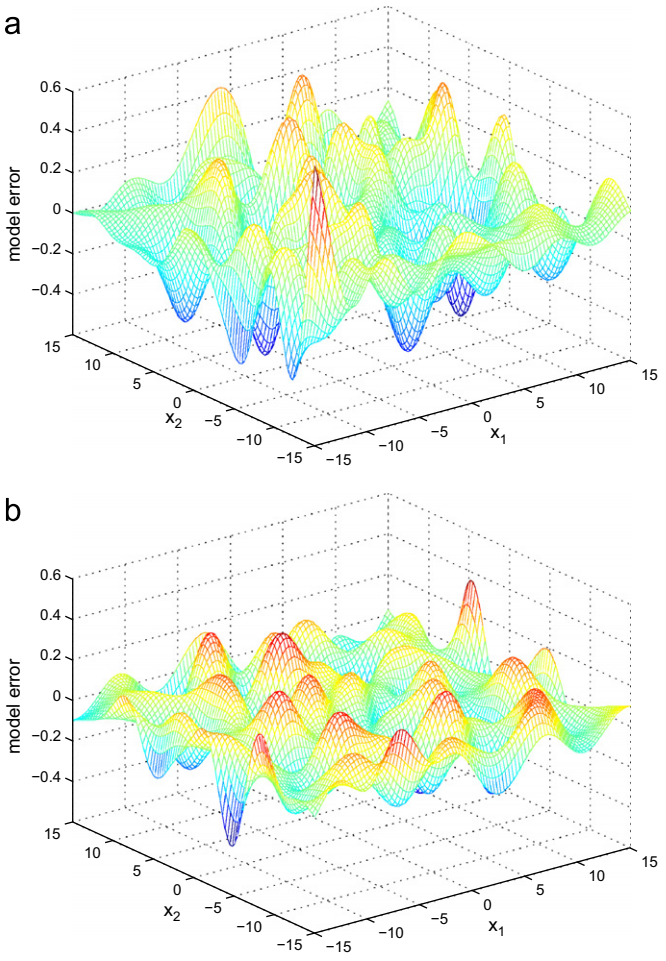

Fig. 2. (a) The modelling error $f(x_1,x_2)-\hat{f}(x_1,x_2)$ of the standard RBF model, and (b) the modelling error $f(x_1,x_2)-\hat{f}(x_1,x_2)$ of the SRBF model, for the symmetric system identification example.

Table 1
Performance comparison between the conventional RBF and SRBF models for the symmetric system identification example.

	Model size	Training MSE	Test MSE	Test MME
RBF	105	0.1543	0.2047	0.0294
SRBF	68	0.1566	0.1839	0.0093

The generalisation performance was also evaluated with the mean modelling error (MME)

$$\text{MME} = E[(f(x_1,x_2)-\hat{f}(x_1,x_2))^2] \tag{26}$$

by averaging over the grid of 90 601 points, where $\hat{f}(x_1,x_2)$ denotes the identified model mapping.

Table 1 compares the performance of the two RBF models obtained. Fig. 2(a) and (b) show the modelling error $f(x_1,x_2)-\hat{f}(x_1,x_2)$ on the grid of 90 601 points for the two obtained models, respectively. It can be seen that, by incorporating the prior information, the SRBF model offers a significantly better generalisation performance. Specifically, its test MME is three times smaller than that of the standard RBF model. It is also interesting to compare the efficiency of model construction for the two models. For the class of OLS learning algorithms [18–22], the complexity of selecting an M-term model from the K-term candidate set is well known to be

$$C = (M+1)\times\mathcal{O}(K^2) \tag{27}$$

where $\mathcal{O}(K^2)$ stands for the order of K^2. For the SRBF model, we obtained $M=68$, while $M=105$ was arrived for the black-box RBF model. Thus, for this example, the complexity of the SRBF model construction is only 65% of the complexity for the standard RBF model construction. By incorporating the prior information naturally, we also improve the efficiency of model construction procedure. Finally, the prediction complexity of the two models are approximately the same. This is because, although the SRBF unit (23) requires more computation than the standard RBF unit (3), the SRBF model has fewer RBF units and, therefore, the computational requirements for calculating a test data point are roughly equal for the two models.

4. BVC-RBF modelling

Again consider the identification of the unknown system f of (1) using the RBF model (2) based on the noisy training data set D_K. In addition, the unknown system mapping f is known to satisfy a set of the L BVCs given by

$$f(\mathbf{x}_j) = d_j, \quad 1\le j\le L \tag{28}$$

where $\mathbf{x}_j\in\mathrm{R}^m$ and $d_j\in\mathrm{R}$ are known. These BVCs may represent the fact that at some critical regions, there is a complete knowledge about the system. For example, at some boundary points $\mathbf{x}_j$, the behaviour of the process is completely determined by the known physics laws that govern the process. Note that the sensor observations on these points $\mathbf{x}_j$ are, however, stochastic because

of the observation noise. Thus, from the noisy D_K, the BVCs (28) may not be seen clearly.

4.1. BVC-RBF network

Since the BVCs of (28) are critical to the underlying system f to be identified, any identified model $\hat{f}^{(M)}$ is required to *strictly* meet these BVCs, that is,

$$\hat{f}^{(M)}(\mathbf{x}_j) = d_j, \quad 1 \le j \le L \tag{29}$$

It is obvious that the black-box RBF model with the node structure (3) cannot guarantee to satisfy the known set of BVCs. The conventional way of incorporating the BVCs (29) as a set of equality constraints in the learning will complicate the resulting optimisation problem and dramatically increases the learning complexity. The novel BVC-RBF network model proposed in [46] has the capacity of satisfying the given BVCs automatically without any added algorithmic complexity and computational cost.

The BVC-RBF model derived in [46] takes the form

$$\hat{y}^{(M)}(k) = \hat{f}^{(M)}(\mathbf{x}(k)) = \sum_{i=1}^{M} \theta_i p_i(\mathbf{x}(k);\sigma) + q(\mathbf{x}(k)) \tag{30}$$

with the novel RBF node structure

$$p_i(\mathbf{x};\sigma) = h(\mathbf{x})\varphi(\|\mathbf{x}-\mathbf{c}_i\|/\sigma) \tag{31}$$

where

$$h(\mathbf{x}) = \sqrt[L]{\prod_{j=1}^{L} \|\mathbf{x}-\mathbf{x}_j\|} \tag{32}$$

is the geometric mean of the data sample $\mathbf{x}$ to the set of boundary values $\mathbf{x}_j$, $1 \le j \le L$. The function $q(\mathbf{x})$ is known as the offset function which takes the form

$$q(\mathbf{x}) = \sum_{j=1}^{L} \gamma_j e^{-(\|\mathbf{x}-\mathbf{x}_j\|^2/\tau)} \tag{33}$$

where τ is a positive scalar, and $\boldsymbol{\gamma}_L = [\gamma_1 \cdots \gamma_L]^T$ is the set of parameters that are obtained by solving the set of linear equations $q(\mathbf{x}_j) = d_j$, $1 \le j \le L$, as follows:

$$\boldsymbol{\gamma}_L = \mathbf{Q}_L^{-1}\mathbf{d}_L \tag{34}$$

where $\mathbf{d}_L = [d_1 \cdots d_L]^T$ and

$$\mathbf{Q}_L = \begin{bmatrix} 1 & e^{-(\|\mathbf{x}_1-\mathbf{x}_2\|^2/\tau)} & \cdots & e^{-(\|\mathbf{x}_1-\mathbf{x}_L\|^2/\tau)} \\ e^{-(\|\mathbf{x}_2-\mathbf{x}_1\|^2/\tau)} & 1 & \ddots & e^{-(\|\mathbf{x}_2-\mathbf{x}_L\|^2/\tau)} \\ \vdots & \ddots & \ddots & \vdots \\ e^{-(\|\mathbf{x}_L-\mathbf{x}_1\|^2/\tau)} & e^{-(\|\mathbf{x}_L-\mathbf{x}_2\|^2/\tau)} & \cdots & 1 \end{bmatrix} \tag{35}$$

In the case that (35) is ill-conditioned, a regularisation technique can be applied to the above solution.

It is easy to verify that with this BVC-RBF model, the BVCs (29) are automatically satisfied. To elaborate further, we note the following features of the BVC-RBF structure.

1. The BVC-RBF nodes (31) have the property of *zero forcing* at the boundary points $\mathbf{x}_j$, $1 \le j \le L$, and the adjustable RBF weights θ_i have no effects on the summation term in (30) at any of the boundary points.
2. The term $q(\mathbf{x})$ passes all the predetermined boundary values $f(\mathbf{x}_j) = q(\mathbf{x}_j) = d_j$, $1 \le j \le L$, and it is completely determined by the BVCs (28) but does not contain any adjustable parameters dependent on D_K.
3. Over the input range, the set of smooth BVC-RBF nodes $p_i(\mathbf{x};\sigma)$ has diverse local responses, and has non-zero adjustable contribution towards modelling $f(\mathbf{x})$ via the adjustable parameters θ_i which are learnt based on the training set D_K.

The above properties 1. and 2. of the BVC-RBF nodes (31) and the offset function (33) are illustrated in Fig. 3 for a one-dimensional function $f(x)$ with the two BVCs of $f(0.1) = -2$ and $f(0.5) = 3$.

With this BVC-RBF model, no constrained optimisation is needed. In fact, define the desired output vector for training this grey-box RBF model as

$$\mathbf{y} = [y(1) - q(\mathbf{x}(1))\ y(2) - q(\mathbf{x}(2)) \cdots y(K) - q(\mathbf{x}(K))]^T \tag{36}$$

where $(\mathbf{x}(k), y(k)) \in D_K$, $1 \le k \le K$. Then the learning of this grey-box RBF model with the node structure (31) and the offset function (33) takes the same regression modelling form as the black-box RBF model discussed in Section 2. Thus, the class of OLS learning algorithms [18–22] can readily be applied to identify a parsimonious BVC-RBF model from the noisy training data D_K.

4.2. A BVC modelling example

A 31×31 meshed data set $f(x_1,x_2)$, as depicted in Fig. 4(a), was generated by using Matlab command *membrane.m* for the third eigenfunction of the L-shaped membrane, which was defined over a unit square input region $(x_1,x_2) \in [0,1]^2$. In Fig. 4(b), the required $L = 120$ BVCs, given by the coordinates of $\{(x_1,x_2), f(x_1,x_2)\}$, are marked by the cross points at the corresponding $\{(x_1,x_2)\}$.

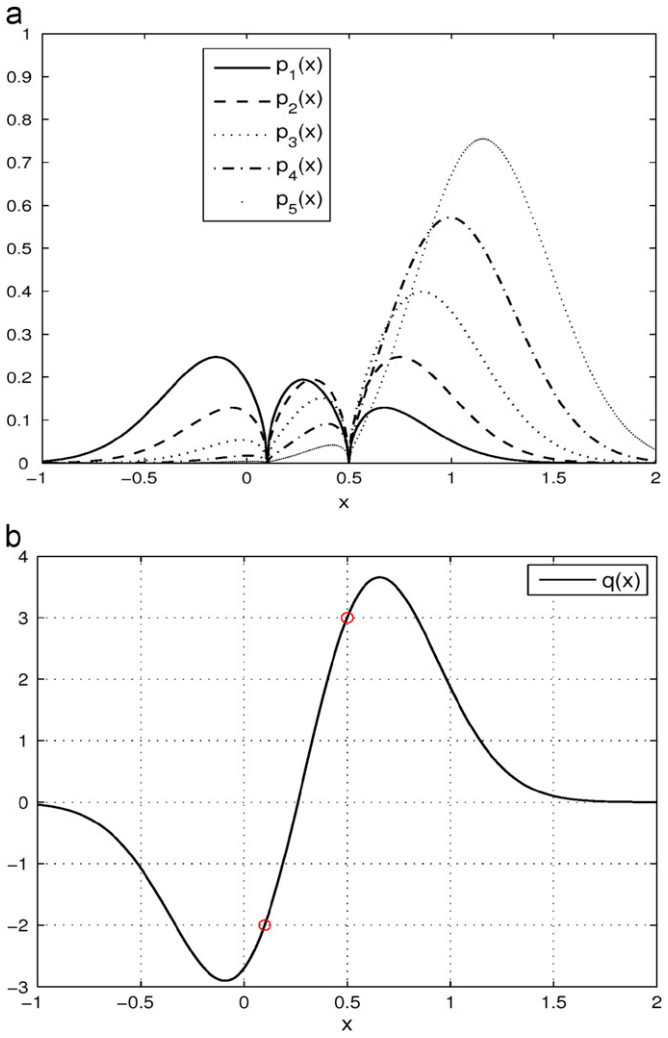

Fig. 3. (a) Five BVC-RBF nodes with zero forcing at the two boundary points, and (b) the offset passing function $q(x)$, for the one-dimensional function $f(x)$ with the two BVCs of $f(0.1) = -2$ and $f(0.5) = 3$.

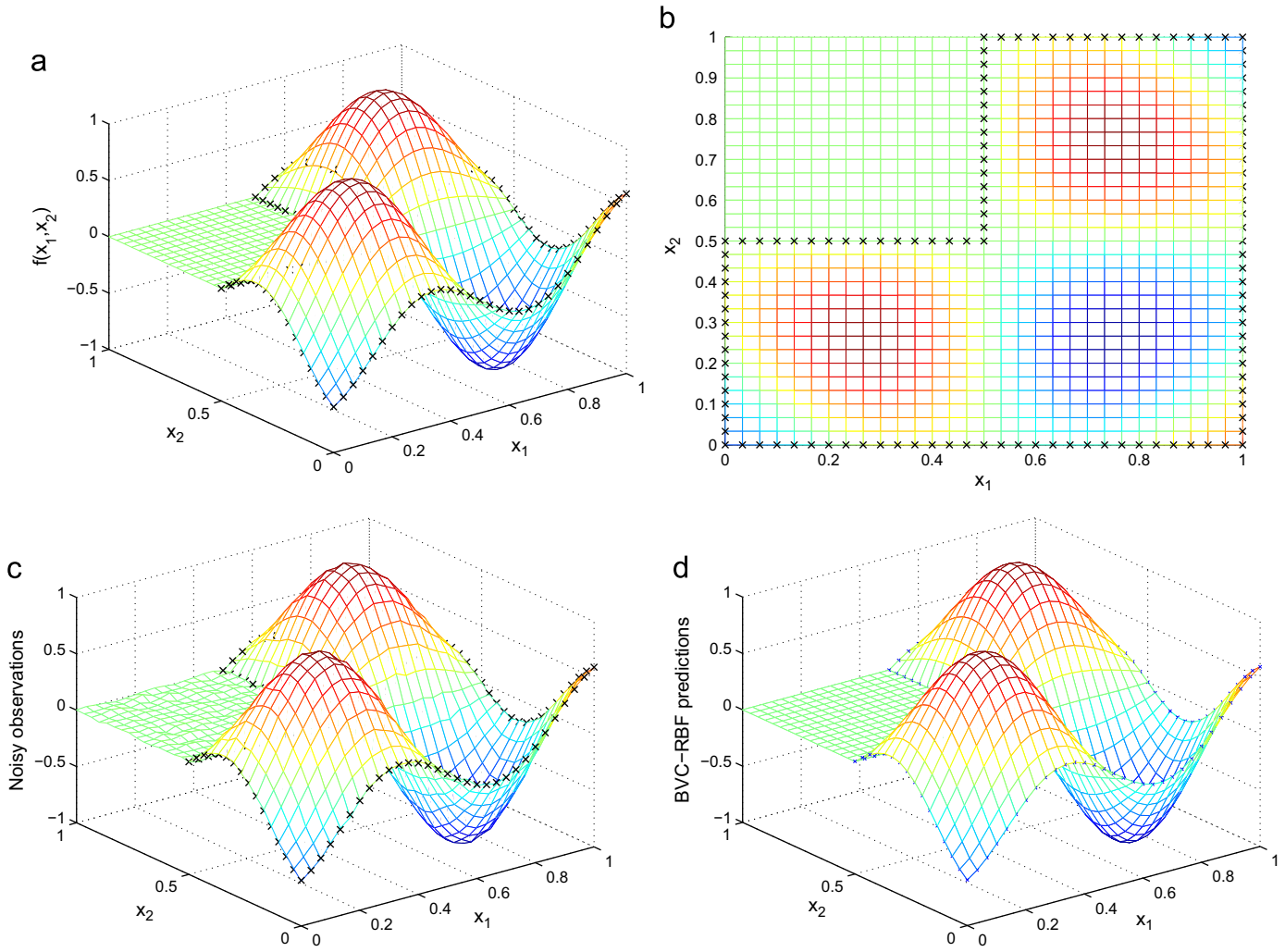

Fig. 4. (a) The underlying function $f(x_1,x_2)$ shown on the grid of 961 points, (b) the $L=120$ BVCs, $\mathbf{x}_j$ for $1 \le j \le L$, marked as cross points, (c) the 961 noisy training data points, and (d) the prediction $\hat{f}(x_1,x_2)$ of the resulting BVC-RBF model.

Table 2
Performance comparison between the conventional RBF and BVC-RBF models for the BVCs system identification example.

	Model size	Training MSE (inside D_K)	Test MME (inside boundary)	Test MME (on boundary)
RBF	42	1.2254×10^{-4}	4.6043×10^{-5}	8.5540×10^{-5}
BVC-RBF	34	9.8634×10^{-5}	1.8230×10^{-5}	5.1462×10^{-11}

The noisy training data set D_K was generated by adding a white Gaussian noise of variance $\sigma_\varepsilon^2 = 0.01^2$ to $f(x_1,x_2)$, and D_K is plotted in Fig. 4(c). We used all the data points of D_K that were inside the boundary as the training samples and applied the D-optimality aided OLS regression algorithm, discussed in Section 2.1 (also see [42]), to construct both the standard RBF and BVC-RBF models. The basic function $\varphi(\bullet)$ was chosen to be Gaussian and the RBF variance $\sigma^2 = 0.2$ was determined separately based on cross validation. For the offset function (33), $\tau = 0.2$ was found to be appropriate. The D-optimality weighting for the combined cost function (12) was chosen to be $\beta = 10^{-6}$.

Table 2 compares the performance of the conventional RBF model obtained with that of the novel BVC-RBF model constructed, where the sizes of the two models were *automatically* determined by the learning algorithm. Fig. 5(a) and (b) depict the modelling error $f(x_1,x_2)-\hat{f}(x_1,x_2)$ of the two obtained models, respectively, where $\hat{f}$ denotes the model mapping identified. The resulting BVC-RBF model is also shown in Fig. 4(d). From Table 2, it can be seen that the BVC-RBF model has a much better generalisation performance than the black-box RBF model. Specifically, the MME calculated inside the boundary marked by the cross points in Fig. 4(b) is more than two times smaller than that for the conventional RBF model. More significantly, the MME of the BVC-RBF model calculated on the boundary is effectively zero, confirming that all the $L=120$ BVCs are strictly met by the BVC-RBF model. By contrast, the black-box RBF model cannot satisfy these BVCs strictly. The results obtained also confirm that the model construction is more efficient for the BVC-RBF model, as a smaller model size was achieved for this grey-box RBF model. Similar to the case of the SRBF modelling, it can be argued that the prediction complexity of the conventional RBF and BVC-RBF models are approximately the same.

5. Conclusions

In this contribution, we have discussed the art of incorporating the prior knowledge to form the appropriate grey-box RBF model. Two types of *a priori* information have been considered. In the first case, the underlying data generating mechanism exhibits the known symmetry property, while in the second case, the underlying

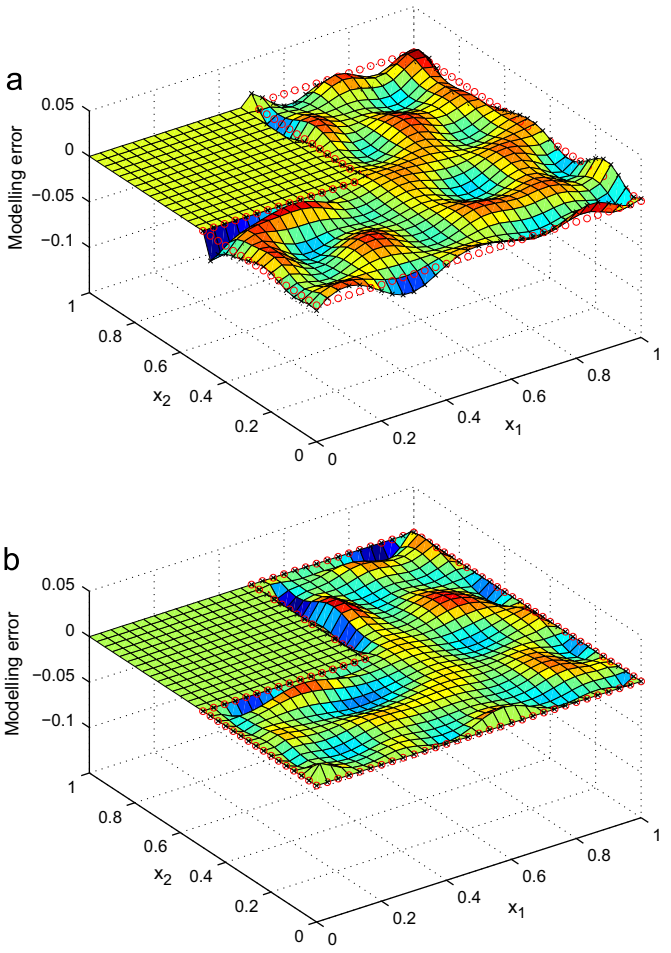

Fig. 5. (a) The modelling error $f(x_1,x_2)-\hat{f}(x_1,x_2)$ of the standard RBF model, and (b) the modelling error $f(x_1,x_2)-\hat{f}(x_1,x_2)$ of the BVC-RBF model, for the BVCs system identification example.

process obeys a set of boundary value constraints. The novel SRBF model and the BVC-RBF model have been proposed, respectively, to incorporate these two types of *a priori* information naturally. The existing state-of-the-arts RBF learning methods for the black-box RBF model can readily be applied to construct these two grey-box RBF models efficiently, without any modification or added algorithmic complexity and computational cost. This contribution has clearly demonstrated that incorporating appropriate prior knowledge naturally into the model structure leads to a better generalisation performance, a smaller model size and a reduced complexity in model construction.

References

[1] S. Chen, S.A. Billings, C.F.N. Cowan, P.M. Grant, Non-linear systems identification using radial basis functions, Int. J. Syst. Sci. 21 (12) (1990) 2513–2539.
[2] J.A. Leonard, M.A. Kramer, Radial basis function networks for classifying process faults, IEEE Control Systems Mag. 11 (3) (1991) 31–38.
[3] S. Chen, B. Mulgrew, P.M. Grant, A clustering technique for digital communications channel equalization using radial basis function networks, IEEE Trans. Neural Networks 4 (4) (1993) 570–579.
[4] A. Caiti, T. Parisini, Mapping ocean sediments by RBF networks, IEEE J. Oceanic Eng. 19 (4) (1994) 577–582.
[5] D. Gorinevsky, A. Kapitanovsky, A. Goldenberg, Radial basis function network architecture for nonholonomic motion planning and control of free-flying manipulators, IEEE Trans. Robotics and Autom. 12 (3) (1996) 491–496.
[6] I. Cha, S.A. Kassam, RBFN restoration of nonlinearly degraded images, IEEE Trans. Image Process. 5 (6) (1996) 964–975.
[7] M. Rosenblum, L.S. Davis, An improved radial basis function network for visual autonomous road following, IEEE Trans. Neural Networks 7 (5) (1996) 1111–1120.
[8] J.A. Refaee, M. Mohandes, H. Maghrabi, Radial basis function networks for contingency analysis of bulk power systems, IEEE Trans. Power Syst. 14 (2) (1999) 772–778.
[9] S. Muraki, T. Nakai, Y. Kita, K. Tsuda, An attempt for coloring multichannel MR imaging data, IEEE Trans. Visualization and Comput. Graphics 7 (3) (2001) 265–274.
[10] R. Mukai, V.A. Vilnrotter, P. Arabshahi, V. Jamnejad, Adaptive acquisition and tracking for deep space array feed antennas, IEEE Trans. Neural Networks 13 (5) (2002) 1149–1162.
[11] C.-T. Su, T. Yang, C.-M. Ke, A neural-network approach for semiconductor wafer post-sawing inspection, IEEE Trans. Semicond. Manuf. 15 (2) (2002) 260–266.
[12] Y. Li, N. Sundararajan, P. Saratchandran, Z. Wang, Robust neuro-H_∞ controller design for aircraft auto-landing, IEEE Trans. Aerosp. Electron. Syst. 40 (1) (2004) 158–167.
[13] M.-J. Lee, Y.-K. Choi, An adaptive neurocontroller using RBFN for robot manipulators, IEEE Trans. Ind. Electron. 51 (3) (2004) 711–717.
[14] S.X. Ng, M.-S. Yee, L. Hanzo, Coded modulation assisted radial basis function aided turbo equalization for dispersive Rayleigh-fading channels, IEEE Trans. Wireless Commun. 3 (6) (2004) 2198–2206.
[15] Y.-J. Oyang, S.-C. Hwang, Y.-Y. Ou, C.-Y. Chen, Z.W. Chen, Data classification with radial basis function networks based on a novel kernel density estimation algorithm, IEEE Trans. Neural Networks 16 (1) (2005) 225–236.
[16] N. Acir, I. Oztura, M. Kuntalp, B. Baklan, C. Guzelis, Automatic detection of epileptiform events in EEG by a three-stage procedure based on artificial neural networks, IEEE Trans. Biomed. Eng. 52 (1) (2005) 30–40.
[17] K.K. Tan, S. Zhao, S. Huang, Iterative reference adjustment for high-precision and repetitive motion control applications, IEEE Trans. Control Syst. Technol. 13 (1) (2005) 85–97.
[18] S. Chen, S.A. Billings, W. Luo, Orthogonal least squares methods and their application to non-linear system identification, Int. J. Control 50 (5) (1989) 1873–1896.
[19] S. Chen, C.F.N. Cowan, P.M. Grant, Orthogonal least squares learning algorithm for radial basis function networks, IEEE Trans. Neural Networks 2 (2) (1991) 302–309.
[20] S. Chen, Y. Wu, B.L. Luk, Combined genetic algorithm optimisation and regularised orthogonal least squares learning for radial basis function networks, IEEE Trans. Neural Networks 10 (5) (1999) 1239–1243.
[21] S. Chen, X. Hong, C.J. Harris, Sparse kernel regression modelling using combined locally regularized orthogonal least squares and D-optimality experimental design, IEEE Trans. Autom. Control 48 (6) (2003) 1029–1036.
[22] S. Chen, X. Hong, C.J. Harris, P.M. Sharkey, Sparse modelling using orthogonal forward regression with PRESS statistic and regularization, IEEE Trans. Syst., Man Cybernet. Part B 34 (2) (2004) 898–911.
[23] S. Chen, L. Hanzo, A. Wolfgang, Kernel-based nonlinear beamforming construction using orthogonal forward selection with Fisher ratio class separability measure, IEEE Signal Process. Lett. 11 (5) (2004) 478–481.
[24] X. Hong, S. Chen, C.J. Harris, Fast kernel classifier construction using orthogonal forward selection to minimise leave-one-out misclassification rate, in: Proceedings of the 2006 International Conference on Intelligent Computing, Kunming, China, August 16–19, 2006, pp. 106–114.
[25] X. Hong, S. Chen, C.J. Harris, A fast linear-in-the-parameters classifier construction algorithm using orthogonal forward selection to minimize leave-one-out misclassification rate, Int. J. Syst. Sci. 39 (2) (2008) 119–125.
[26] S. Chen, X. Hong, C.J. Harris, Sparse kernel density construction using orthogonal forward regression with leave-one-out test score and local regularization, IEEE Trans. Syst., Man Cybernet. Part B 34 (4) (2004) 1708–1717.
[27] S. Chen, X. Hong, C.J. Harris, Sparse kernel density estimator using orthogonal regression based on *D*-optimality experimental design, in: Proceedings of the IJCNN 2008, Hong Kong, China, June 1–6, 2008, pp. 1–6.
[28] S. Chen, X. Hong, C.J. Harris, An orthogonal forward regression technique for sparse kernel density estimation, Neurocomputing 71 (4–6) (2008) 931–943.
[29] Y.S. Abu-Mostafa, An algorithm for learning from hints, in: Proceedings of the IJCNN'93, vol. 2, October 25–29, 1993, pp. 1653–1656.
[30] L.A. Aguirre, R.A.M. Lopes, G.F.V. Amaral, C. Letellier, Constraining the topology of neural networks to ensure dynamics with symmetry properties, Phys. Rev. E 69 (2) (2004) 026701-1–0267011-1.
[31] M. Espinoza, J.A.K. Suykens, B. De Moor, Imposing symmetry in least squares support vector machines regression, in: Proceedings of the Joint 44th IEEE CDC and European Control Conference, 2005, Seville, Spain, December 12–15, 2005, pp. 5716–5721.
[32] S. Chen, A. Wolfgang, C.J. Harris, L. Hanzo, Symmetric RBF classifier for nonlinear detection in multiple-antenna aided systems, IEEE Trans. Neural Networks 19 (5) (2008) 737–745.
[33] S. Chen, L. Hanzo, S. Tan, Symmetric complex-valued RBF receiver for multiple-antenna aided wireless systems, IEEE Trans. Neural Networks 19 (9) (2008) 1657–1663.
[34] H.J.A.F. Tulleken, Grey-box modelling and identification using physical knowledge and Bayesian techniques, Automatica 29 (2) (1993) 285–308.
[35] T. Bohlin (Ed.), Special issue on grey box modelling, Int. J. Adaptive Control Signal Process. 9 (6) (1995).
[36] S.A. Billings, S. Chen, Extended model set, global data and threshold model identification of severely non-linear systems, Int. J. Control 50 (5) (1989) 1897–1923.

[37] K.C. Tan, Y. Li, Grey-box model identification via evolutionary computing, Control Eng. Pract. 10 (7) (2002) 673–684.
[38] K. Li, Eng-genes: a new genetic modelling approach for nonlinear dynamic systems, in: Proceedings of the 16th IFAC World Congress, Prague, Czech Republic, July 3–8, 2005, p. 6.
[39] K. Li, J. Peng, System oriented neural networks—problem formulation, methodology and application, Int. J. Recogn. Artif. Intell. 20 (2) (2006) 143–158.
[40] S. Chen, Local regularization assisted orthogonal least squares regression, Neurocomputing 69 (4–6) (2006) 559–585.
[41] A.C. Atkinson, A.N. Donev, Optimum Experimental Designs, Clarendon Press, Oxford, UK, 1992.
[42] X. Hong, C.J. Harris, Nonlinear model structure detection using optimum experimental design and orthogonal least squares, IEEE Trans. Neural Networks 12 (2) (2001) 435–439.
[43] M. Stone, Cross validation choice and assessment of statistical predictions, J. R. Stat. Soc. Ser. B 36 (1) (1974) 111–147.
[44] R.H. Myers, Classical and Modern Regression with Applications, second ed. PWS-KEN, Boston, 1990.
[45] X. Hong, P.M. Sharkey, K. Warwick, Automatic nonlinear predictive model construction algorithm using forward regression and the PRESS statistic, IEE Proc. Control Theory Appl. 150 (3) (2003) 245–254.
[46] X. Hong, S. Chen, A new RBF neural network with boundary value constraints, IEEE Trans. Syst., Man, Cybernet., Part B 39 (1) (2009) 298–303.

Sheng Chen received his Ph.D. degree in control engineering from the City University, London, UK, in September 1986. He was awarded the Doctor of Sciences (D.Sc.) degree by the University of Southampton, Southampton, UK, in 2005.

From October 1986 to August 1999, he held research and academic appointments at the University of Sheffield, the University of Edinburgh and the University of Portsmouth, all in UK. Since September 1999, he has been with the School of Electronics and Computer Science, University of Southampton, UK. Professor Chen's research interests include wireless communications, adaptive signal Processing for communications, machine learning, and evolutionary computation methods. He has published over 400 research papers.

Dr Chen is a Fellow of IET and a Fellow of IEEE. 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 (2004) in the engineering category.

Xia Hong received her university education at National University of Defence Technology, China (B.Sc. 1984, M.Sc., 1987), and her Ph.D. degree from University of Sheffield, Sheffield, UK, in 1998, all in automatic control.

She worked as a research assistant in Beijing Institute of Systems Engineering, Beijing, China from 1987 to 1993. She worked as a research fellow in the School of Electronics and Computer Science at the University of Southampton, UK, from 1997 to 2001. Since 2001, She has been with the School of Systems Engineering, the University of Reading, UK, where she currently holds a Readership post. Dr. Hong is actively engaged in research into nonlinear systems identification, data modelling, estimation and intelligent control, neural networks, pattern recognition, learning theory and their applications. She has published over 200 research papers, and coauthored a research book.

Dr. Hong was awarded a Donald Julius Groen Prize by IMechE in 1999.

Chris J. Harris received his Ph.D. degree from the University of Southampton, Southampton, UK, in 1972. He was awarded the Doctor of Sciences (D.Sc.) degree by the University of Southampton in 2001.

He previously held appointments at the University of Hull, the UMIST, the University of Oxford, and the University of Cranfield, all in UK, as well as being employed by the UK 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. His research interests lie in 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. He has authored and coauthored 12 research books and over 400 research papers, and he is the associate editor of numerous international journals.

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 IEE Faraday medal in 2001 for his work in intelligent control and neurofuzzy systems.