

Fully Complex-Valued Radial Basis Function Networks for Orthogonal Least Squares Regression

S. Chen, X. Hong and C.J. Harris

Abstract—We consider a fully complex-valued radial basis function (RBF) network for regression application. The locally regularised orthogonal least squares (LROLS) algorithm with the D -optimality experimental design, originally derived for constructing parsimonious real-valued RBF network models, is extended to the fully complex-valued RBF network. Like its real-valued counterpart, the proposed algorithm aims to achieve maximised model robustness and sparsity by combining two effective and complementary approaches. The LROLS algorithm alone is capable of producing a very parsimonious model with excellent generalisation performance while the D -optimality design criterion further enhances the model efficiency and robustness. By specifying an appropriate weighting for the D -optimality cost in the combined model selecting criterion, the entire model construction procedure becomes automatic. An example of identifying a complex-valued nonlinear channel is used to illustrate the regression application of the proposed fully complex-valued RBF network.

I. INTRODUCTION

The complex-valued radial basis function (RBF) network of [1] has widely been used in nonlinear signal processing applications that involve complex-valued signals. In this complex-valued RBF network, each RBF node has a real-valued response that can be interpreted as a conditional probability density function. This interpretation makes such a complex-valued RBF network particularly useful in the equalisation application of communication channels with complex-valued signals [2], [3], [4], [5], [6]. Because the RBF node's response is real-valued, this complex-valued RBF network is essentially two separate real-valued RBF networks. Various learning methods, such as the orthogonal least squares (OLS) forward selection algorithm [7], [8], can readily be adopted to this complex-valued RBF network for regression application. In this study, we consider a different complex-valued RBF network, where each RBF node has a complex-valued response, and we develop an efficient construction algorithm for selecting sparse fully complex-valued RBF networks with excellent generalisation capability.

Among various learning algorithms for real-valued RBF networks, the local regularisation assisted OLS (LROLS) algorithm combined with the D -optimality experimental design criterion [9] is a powerful algorithm for constructing parsimonious real-valued RBF networks that generalise well, because it combines two effective and complementary approaches for modelling, namely, the local regularisation

assisted OLS regression [10], [11] and the D -optimality experimental design [12], [13]. By adopting multiple regularisers, the LROLS algorithm is capable of constructing very sparse real-valued RBF models with excellent generalisation capability from noisy data [10], [11]. Optimal experimental designs [12] have been used to construct smooth model response surfaces based on the setting of the experimental variables under well controlled experimental conditions. In optimal design, model adequacy is evaluated by design criteria that are statistical measures of goodness of experimental designs by virtue of design efficiency and experimental effort. For real-valued RBF models, quantitatively, model adequacy is measured as function of the eigenvalues of the design matrix, as it is known that the eigenvalues of the design matrix are linked to the covariance matrix of the least squares (LS) parameter estimate. There exist a variety of optimal design criteria based on different aspects of experimental design [12]. The D -optimality criterion is most effective in optimising the parameter efficiency and model robustness via the maximisation of the determinant of the design matrix. Combining the D -optimality criterion with OLS regression [13] leads to an enhanced construction algorithm, as the coupling effects of the two approaches in the combined algorithm further enhance each other. Moreover, the user only needs to specify a weighting for the D -optimality criterion and the model construction process is fully automatic. The value of this weighting does not influence the model selecting procedure critically and it can be chosen with ease from a wide range of values [9].

We extend this combined LROLS algorithm and D -optimality experimental design to the fully complex-valued RBF network. An example involving the identification of a complex-valued nonlinear channel is used to demonstrate the effectiveness of the proposed algorithm for constructing sparse fully complex-valued RBF network models for regression application. The paper is organised as follows. Section II briefly outlines the proposed fully complex-valued RBF network, while Section III details the LROLS algorithm with D -optimality design for constructing sparse fully complex-valued RBF networks from noisy data. In Section IV we present the case of identifying a complex-valued nonlinear channel, while in Section V we offer our conclusions.

II. FULLY COMPLEX-VALUED RBF NETWORK

Consider the modelling of the data set $D_N = \{\mathbf{x}(k), y(k)\}_{k=1}^N$, where N is the number of training data, $\mathbf{x}(k) \in \mathcal{C}^m$ and $y(k) \in \mathcal{C}$ are the k -th complex-valued training input vector and corresponding complex-valued desired

S. Chen and C.J. Harris are with School of Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, UK, E-mails: {sqc,cjh}@ecs.soton.ac.uk

X. Hong is with School of Systems Engineering, University of Reading, Reading RG6 6AY, U.K. E-mail: x.hong@reading.ac.uk

response, respectively, by the RBF network of the form

$$\hat{y}(k) = \sum_{i=1}^M \theta_i \phi_i(\mathbf{x}(k)), \quad (1)$$

where $\hat{y}(k)$ denotes the complex-valued model output, θ_i are the complex-valued model weights, M is the number of RBF nodes, and $\phi_i(\mathbf{x}(k))$ denote the complex-valued RBF nodes' response. In particular, the response of the RBF node is specified by

$$\phi_i(\mathbf{x}) = \varphi(\|\Re[\mathbf{x}] - \Re[\mathbf{c}_i]\|/\rho_i) + j\varphi(\|\Im[\mathbf{x}] - \Im[\mathbf{c}_i]\|/\rho_i), \quad (2)$$

where $\Re[\bullet]$ and $\Im[\bullet]$ denote the real and imaginary parts, respectively, $\mathbf{c}_i \in \mathcal{C}^m$ is the i -th complex-valued RBF centre vector, $\rho_i^2 > 0$ is the i -th RBF variance, and $\varphi(\bullet)$ is the usual real-valued basis function. Two typical basis functions are the thin-plate-spline function

$$\varphi(\chi/1) = \chi^2 \log(\chi) \quad (3)$$

and the Gaussian function

$$\varphi(\chi/\rho) = e^{-\chi^2/\rho^2}. \quad (4)$$

Define the modelling residual for $\mathbf{x}(k) \in D_N$ as $e(k) = y(k) - \hat{y}(k)$. Further consider every data points as candidate centres, namely, $M = K$ and $\mathbf{c}_i = \mathbf{x}(i)$ for $1 \leq i \leq M$. Moreover, set every RBF variance to a given value $\rho_i^2 = \rho^2$. Then we obtain the regression model over the data set D_N

$$\mathbf{y} = \Phi \boldsymbol{\theta} + \mathbf{e} \quad (5)$$

where $\mathbf{y} = [y(1) \ y(2) \ \dots \ y(N)]^T$, $\boldsymbol{\theta} = [\theta_1 \ \theta_2 \ \dots \ \theta_M]^T$, $\mathbf{e} = [e(1) \ e(2) \ \dots \ e(N)]^T$ and the complex-valued regression matrix

$$\Phi = [\phi_1 \ \phi_2 \ \dots \ \phi_M] \quad (6)$$

with columns $\phi_i = [\phi_i(\mathbf{x}(1)) \ \phi_i(\mathbf{x}(2)) \ \dots \ \phi_i(\mathbf{x}(N))]^T$. Let an orthogonal decomposition of Φ be $\Phi = \mathbf{W}\mathbf{A}$, where

$$\mathbf{A} = \begin{bmatrix} 1 & a_{1,2} & \dots & a_{1,M} \\ 0 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \alpha_{M-1,M} \\ 0 & \dots & 0 & 1 \end{bmatrix} \quad (7)$$

with complex-valued $\alpha_{i,l}$, $1 \leq i < l \leq M$, and the complex-valued orthogonal matrix

$$\mathbf{W} = [\mathbf{w}_1 \ \mathbf{w}_2 \ \dots \ \mathbf{w}_M] \quad (8)$$

with columns satisfying $\mathbf{w}_i^H \mathbf{w}_l = 0$, if $i \neq l$. The regression model (5) can alternatively be expressed as

$$\mathbf{y} = \mathbf{W}\mathbf{g} + \mathbf{e}, \quad (9)$$

where the weight vector $\mathbf{g} = [g_1 \ g_2 \ \dots \ g_M]^T$ defined in the orthogonal model space satisfies the following triangular system $\mathbf{A}\boldsymbol{\theta} = \mathbf{g}$.

III. LOCALLY REGULARISED OLS ALGORITHM WITH D-OPTIMALITY DESIGN

We first briefly describe the two components of the combined LROLS algorithm and D -optimality design.

A. Locally regularised OLS algorithm

Like the real-valued LROLS algorithm [10], [11], the complex-valued version also adopts a similar regularised error criterion defined as

$$J_R(\mathbf{g}, \boldsymbol{\lambda}) = \mathbf{e}^H \mathbf{e} + \sum_{i=1}^M \lambda_i |g_i|^2 = \mathbf{e}^H \mathbf{e} + \mathbf{g}^H \boldsymbol{\Lambda} \mathbf{g}, \quad (10)$$

where $\boldsymbol{\lambda} = [\lambda_1 \ \lambda_2 \ \dots \ \lambda_M]^T$ is the regularisation parameter vector and $\boldsymbol{\Lambda} = \text{diag}\{\lambda_1, \lambda_2, \dots, \lambda_M\}$. Similar to the real-valued case [11], with \mathbf{g} set to its optimal value, i.e. $\frac{\partial J_R}{\partial \mathbf{g}} = 0$, the criterion (10) can be expressed as (see Appendix A)

$$\mathbf{e}^H \mathbf{e} + \mathbf{g}^H \boldsymbol{\Lambda} \mathbf{g} = \mathbf{y}^H \mathbf{y} - \sum_{i=1}^M (\mathbf{w}_i^H \mathbf{w}_i + \lambda_i) |g_i|^2. \quad (11)$$

Normalising (11) by $\mathbf{y}^H \mathbf{y}$ yields

$$(\mathbf{e}^H \mathbf{e} + \mathbf{g}^H \boldsymbol{\Lambda} \mathbf{g}) / \mathbf{y}^H \mathbf{y} = 1 - \sum_{i=1}^M (\mathbf{w}_i^H \mathbf{w}_i + \lambda_i) |g_i|^2 / \mathbf{y}^H \mathbf{y}. \quad (12)$$

As in the case of the original OLS algorithm [7], the regularised error reduction ratio due to \mathbf{w}_i is defined by

$$[\text{rerr}]_i = (\mathbf{w}_i^H \mathbf{w}_i + \lambda_i) |g_i|^2 / \mathbf{y}^H \mathbf{y}. \quad (13)$$

Based on this ratio, significant regressors can be selected in a forward regression procedure, and the selection process is terminated at the n_s -th stage when

$$1 - \sum_{l=1}^{n_s} [\text{rerr}]_l < \xi \quad (14)$$

is satisfied, where ξ is a chosen tolerance. This produces a sparse model containing n_s ($\ll M$) significant regressors.

The regularisation parameters specify the prior distributions of \mathbf{g} . Since initially we do not know the optimal value of \mathbf{g} , λ_i should be initialised to the same small value, and this corresponds to choose a same flat distribution for each prior of g_i [11]. Similar to the real-valued regression model case [11], applying the evidence procedure [14] will lead to the updating formulas for the regularisation parameters

$$\lambda_i^{\text{new}} = \frac{\gamma_i^{\text{old}}}{N - \gamma^{\text{old}}} \frac{\mathbf{e}^H \mathbf{e}}{|g_i|^2}, \quad 1 \leq i \leq M, \quad (15)$$

where g_i denotes the current optimal weight solution, and

$$\gamma_i = \frac{\mathbf{w}_i^H \mathbf{w}_i}{\lambda_i + \mathbf{w}_i^H \mathbf{w}_i} \quad \text{and} \quad \gamma = \sum_{i=1}^M \gamma_i. \quad (16)$$

Usually a few iterations (typically 10 to 20) are sufficient to find an optimal $\boldsymbol{\lambda}$.

B. D-optimality experimental design

Adopting the usual concepts of experimental design, we refer to the matrix $\Phi^H \Phi$ as the design matrix. The LS estimate of $\boldsymbol{\theta}$ is given by $\hat{\boldsymbol{\theta}} = (\Phi^H \Phi)^{-1} \Phi^H \mathbf{y}$. Assume that (5) represents the true data generating process and $\Phi^H \Phi$ is nonsingular. Then, the LS estimate $\hat{\boldsymbol{\theta}}$ is unbiased and the

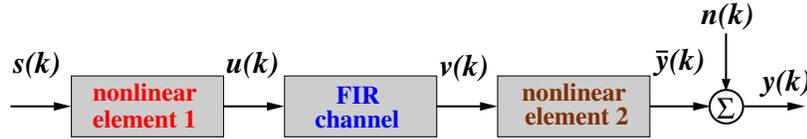


Fig. 1. Baseband discrete-time model of a nonlinear channel.

covariance matrix of the estimate is determined by the design matrix

$$\begin{cases} E[\hat{\theta}] = \theta, \\ Cov[\hat{\theta}] \propto (\Phi^H \Phi)^{-1}. \end{cases} \quad (17)$$

It is well known that the model based on the pure LS estimate tends to be unsatisfactory for an ill conditioned regression matrix (design matrix). The condition number of the design matrix is given by

$$C = \frac{\max\{\kappa_i, 1 \leq i \leq M\}}{\min\{\kappa_i, 1 \leq i \leq M\}}, \quad (18)$$

with κ_i , $1 \leq i \leq M$, being the eigenvalues of $\Phi^H \Phi$. Too large a condition number will result in unstable LS parameter estimate while a small condition number improves model robustness. The D -optimality design criterion maximises the determinant of the design matrix for the constructed model. Specifically, let Φ_{n_s} be a column subset of Φ representing a constructed n_s -term subset model. According to the D -optimality criterion, the selected subset model is the one that maximises $\det(\Phi_{n_s}^H \Phi_{n_s})$. This helps to prevent the selection of an oversized ill-posed model and the problem of high parameter estimate variances.

It is straightforward to verify that maximising $\det(\Phi_{n_s}^H \Phi_{n_s})$ is identical to maximising $\det(\mathbf{W}_{n_s}^H \mathbf{W}_{n_s})$ or, equivalently, minimising $-\log \det(\mathbf{W}_{n_s}^H \mathbf{W}_{n_s})$. In fact,

$$\begin{aligned} \det(\Phi^H \Phi) &= \det(\mathbf{A}^H) \det(\mathbf{W}^H \mathbf{W}) \det(\mathbf{A}) \\ &= \det(\mathbf{W}^H \mathbf{W}) = \prod_{i=1}^M \mathbf{w}_i^H \mathbf{w}_i \end{aligned} \quad (19)$$

and

$$-\log(\det(\mathbf{W}^H \mathbf{W})) = \sum_{i=1}^M -\log(\mathbf{w}_i^H \mathbf{w}_i). \quad (20)$$

C. Combined LROLS and D -optimality algorithm

The combined LROLS and D -optimality algorithm adopts the following combined criterion

$$J_{RD}(\mathbf{g}, \lambda, \beta) = J_R(\mathbf{g}, \lambda) + \beta \sum_{i=1}^M -\log(\mathbf{w}_i^H \mathbf{w}_i). \quad (21)$$

In this combined algorithm, the updating of the model weights and regularisation parameters is exactly as in the LROLS algorithm, but the selection is according to the combined regularised error reduction ratio defined as

$$[\text{crrr}]_i = ((\mathbf{w}_i^H \mathbf{w}_i + \lambda_i) |g_i|^2 + \beta \log(\mathbf{w}_i^H \mathbf{w}_i)) / \mathbf{y}^H \mathbf{y} \quad (22)$$

and the selection is terminated with an n_s -term model when

$$[\text{crrr}]_l \leq 0 \text{ for } n_s + 1 \leq l \leq M. \quad (23)$$

Note that there always exists a subset model size n_s such that (23) holds [13]. The iterative model selection procedure can now be summarised:

Initialisation. Set λ_i , $1 \leq i \leq M$, to the same small positive value (e.g. 10^{-6}), and choose a fixed β . Set iteration $I = 1$.

Step 1. Given the current λ , use the procedure described in Appendix B to select a subset model with n_I terms.

Step 2. Update λ using (15) with $M = n_I$. If λ remains sufficiently unchanged in two successive iterations or a pre-set maximum iteration number (e.g. 10) is reached, stop; otherwise set $I+ = 1$ and go to *Step 1*.

The introduction of the D -optimality cost into the algorithm further enhances the efficiency and robustness of the selected subset model and, as a consequence, the combined algorithm can often produce sparser models with equally good generalisation properties, compared with the LROLS algorithm alone. An additional advantage is that it simplifies the selection procedure. Note that it is no longer necessary to specify the tolerance ξ and the algorithm automatically terminates when condition (23) is met. The value of weighting β does not critically influence the performance of this combined LROLS and D -optimality algorithm. This is because the LROLS algorithm alone is capable of producing a very sparse model and the selected model terms are most likely to have large values of $\mathbf{w}_i^H \mathbf{w}_i$. Using the OLS algorithm without local regularisation, this is not necessarily the case, as model terms with small $\mathbf{w}_i^H \mathbf{w}_i$ can have very large $|g_i|^2$ (over-fitted) and consequently will be chosen. Note that with regularisation such over-fitting will not occur. The D -optimality design also favours the model terms with large $\mathbf{w}_i^H \mathbf{w}_i$ and therefore the two component criteria in

TABLE I
MODELLING PERFORMANCE FOR THE NONLINEAR CHANNEL.

basis function	RBF variance	number of RBFs	MSE for training	MSE for testing	mean state error
Gaussian	1.0	15	0.120016	0.129401	0.027739
thin-plate-spline	NA	15	0.120895	0.128526	0.027029

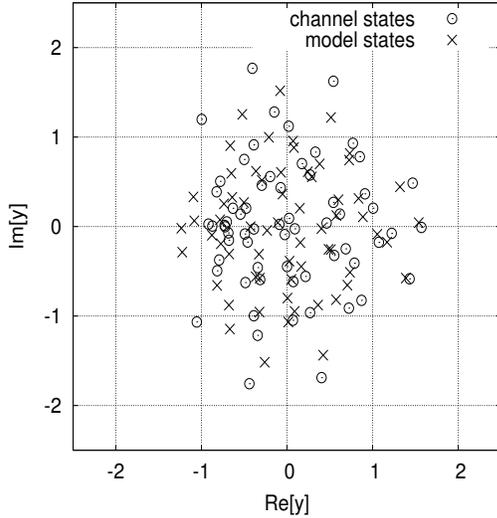


Fig. 2. State constellation, where circles indicate channel states while crosses indicate Gaussian RBF model states.

the combined criterion (22) are not in conflict. Thus, the two methods enhance each other. Consequently, the value of β is not critical in arriving a desired sparse model, and the suitable weighting β can be chosen with ease from a large range of values [9]. It should also be emphasised that the computational complexity of this algorithm is not significantly more than that of the OLS algorithm. This is simply because after the 1st iteration, which has a complexity of the OLS algorithm, the model set contains only n_1 ($\ll M$) terms, and the complexity of the subsequent iteration decreases dramatically. Typically, after a few iterations, the model set will converge to a constant size of very small n_s .

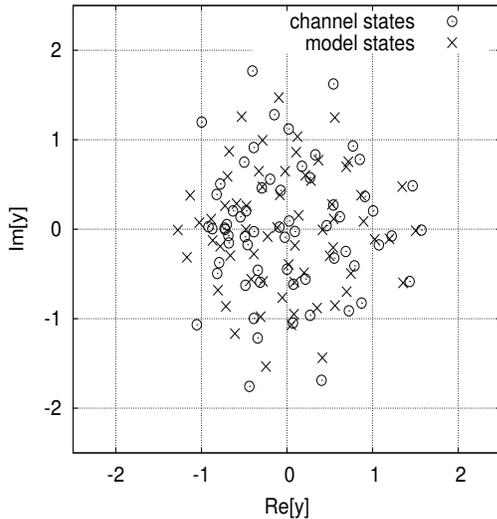


Fig. 3. State constellation, where circles indicate channel states while crosses indicate thin-plate-spline RBF model states.

IV. A MODELLING EXAMPLE

Modelling capabilities of the fully complex-valued RBF network and the efficiency of the combined LROLS and D -optimality algorithm was illustrated using an example of modelling a complex-valued nonlinear communication channel. Fig. 1 depicts the schematic of this nonlinear channel. The transmitted data symbols $s(k) = s_R(k) + js_I(k)$ took values from the 4-QAM constellation defined by

$$S = \{+1 + j, -1 + j, -1 - j, +1 - j\}. \quad (24)$$

The first nonlinear element, representing the nonlinear high power amplifier in the transmitter [15], was modelled by the static nonlinearity

$$u(k) = f_{\text{amp}}(s(k)) = \frac{2s(k)}{1 + |s(k)|^2} e^{j\frac{\pi}{3} \frac{|s(k)|^2}{1 + |s(k)|^2}}. \quad (25)$$

The time-dispersive transmission medium was modelled as a finite-duration impulse response (FIR) filter whose transfer function was defined by

$$A(z) = \frac{V(z)}{U(z)} = (0.3725 + j0.2172) (1 - (0.35 + j0.7)z^{-1}) \times (1 - (0.5 + j)z^{-1}). \quad (26)$$

The second static nonlinear element was a third-order complex-valued Volterra nonlinearity specified by

$$\bar{y}(k) = f_{\text{Vol}}(v(k)) = v(k) + 0.2v^2(k) - 0.1v^3(k). \quad (27)$$

The additive noise $n(k) = n_R(k) + jn_I(k)$, where both $n_R(k)$ and $n_I(k)$ were white Gaussian processes having a same variance σ_n^2 . This nonlinear channel thus was characterised by the complex-valued nonlinear model

$$y(k) = \bar{y}(k) + n(k) = f(\mathbf{x}(k)) + n(k), \quad (28)$$

where $\mathbf{x}(k) = [s(k) \ s(k-1) \ s(k-2)]^T$ and $f(\bullet)$ denoted the complex-valued mapping that specified this nonlinear channel.

For this example, the input vector $\mathbf{x}(k)$ only took values from the input state set defined by

$$\mathcal{X} = \{\bar{\mathbf{x}}_l, 1 \leq l \leq N_{\text{st}}\}, \quad (29)$$

where $N_{\text{st}} = 4^3 = 64$ was the number of input states. Therefore, the noise-free part of the channel output, $\bar{y}(k)$, only took values from the output state set specified by

$$\bar{\mathcal{Y}} = \{\bar{y}_l = f(\bar{\mathbf{x}}_l), 1 \leq l \leq N_{\text{st}}\}. \quad (30)$$

Similarly, the model output $\hat{y}(k) = \hat{f}(\mathbf{x}(k))$, where $\hat{f}(\bullet)$ denoted the RBF model mapping, over the input set \mathcal{X} was defined by

$$\hat{\mathcal{Y}} = \{\hat{y}_l = \hat{f}(\bar{\mathbf{x}}_l), 1 \leq l \leq N_{\text{st}}\}. \quad (31)$$

The mean state error of the model $\hat{y}(k) = \hat{f}(\mathbf{x}(k))$ was then defined by

$$\text{Mean State Error} = \frac{1}{2N_{\text{st}}} \sum_{l=1}^{N_{\text{st}}} |\bar{y}_l - \hat{y}_l|^2. \quad (32)$$

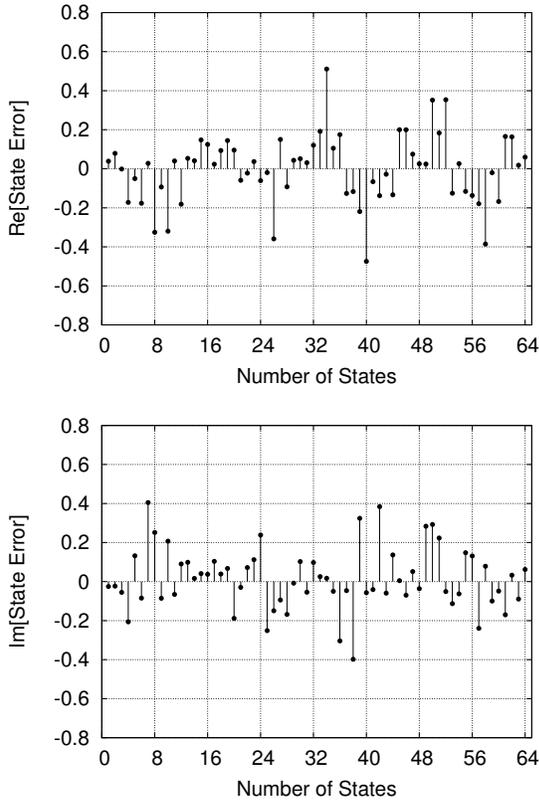


Fig. 4. State errors between the channel and Gaussian RBF model.

Given $\sigma_n^2 = 0.1$, two sets of data $\{\mathbf{x}(k), y(k)\}_{k=1}^N$, each having $N = 400$ points, were generated for the training and testing purposes, respectively. The fully complex-valued RBF networks with both the Gaussian and thin-plate-spline basis functions were applied to the training data set using the combined LROLS and D -optimality algorithm. For this example, it was found that the weighting β was not critical at all and any value in 10^2 to 10^{-6} gave the same excellent modelling performance. For the Gaussian RBF network, the RBF variance was set to $\rho^2 = 1$ via cross validation. The algorithm automatically selected 15 RBF nodes for both the Gaussian and thin-plate-spline RBF models. Table I summarises the modelling performance of the two selected RBF models, where the mean square error (MSE) was defined by

$$\text{MSE} = \frac{1}{2N} \sum_{k=1}^N |y(k) - \hat{y}(k)|^2 \quad (33)$$

with $\hat{y}(k)$ denoting the model output for the input $\mathbf{x}(k)$. It can be seen from Table I that the two RBF network models had similarly good generalisation performance. Fig. 2 plots the model output state set $\hat{\mathcal{Y}}$ for the Gaussian RBF model, while Fig. 3 displays $\hat{\mathcal{Y}}$ of the thin-plate-spline RBF model, in comparison with the true channel state set \mathcal{Y} . The state errors, defined by $\bar{y}_l - \hat{y}_l$, $1 \leq l \leq N_{st}$, are plotted in Figs. 4 and 5, respectively, for the two RBF network models.

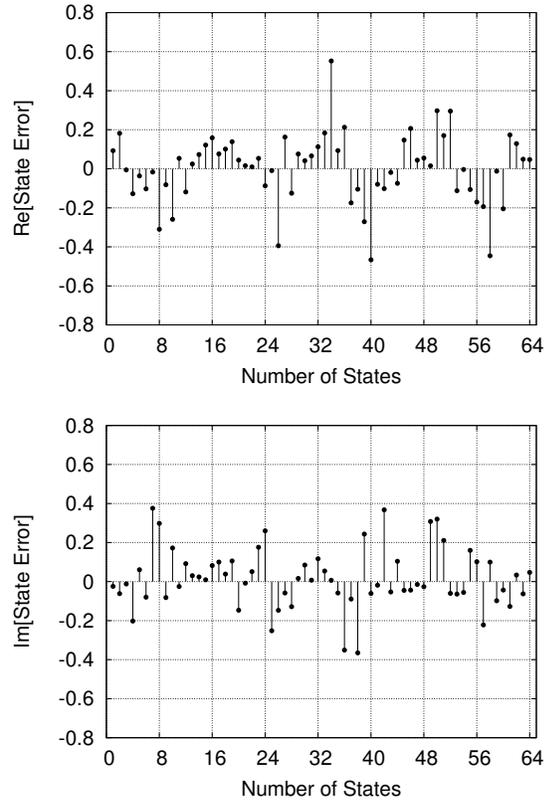


Fig. 5. State errors between the channel and thin-plate-spline RBF model.

V. CONCLUSIONS

A fully complex-valued RBF network has been proposed for regression application. The combined locally regularised OLS algorithm and the D -optimality design, originally derived for real-valued RBF networks, has been extended to select parsimonious fully complex-valued RBF networks with excellent generalisation capability. A modelling example involving the identification of a nonlinear channel has been used to illustrate the proposed approach. It has been demonstrated that combining the local regularisation with the D -optimality experimental design provides a state-of-the-art procedure for constructing very sparse models with excellent generalisation performance. The performance of the algorithm is insensitive to the D -optimality cost weighting, and the model construction process is fully automated.

APPENDIX A

The regularised least squares solution for \mathbf{g} is obtained by setting $\frac{\partial J_R}{\partial \mathbf{g}} = \mathbf{0}$, that is,

$$\mathbf{W}^H \mathbf{y} = (\mathbf{W}^H \mathbf{W} + \mathbf{\Lambda}) \mathbf{g}. \quad (34)$$

Now

$$\begin{aligned} \mathbf{y}^H \mathbf{y} - 2\mathbf{g}^H \mathbf{\Lambda} \mathbf{g} &= (\mathbf{W}\mathbf{g} + \mathbf{e})^H (\mathbf{W}\mathbf{g} + \mathbf{e}) - 2\mathbf{g}^H \mathbf{\Lambda} \mathbf{g} \\ &= \mathbf{g}^H \mathbf{W}^H \mathbf{W} \mathbf{g} + \mathbf{e}^H \mathbf{e} + \mathbf{g}^H \mathbf{W}^H \mathbf{e} + \mathbf{e}^H \mathbf{W} \mathbf{g} - 2\mathbf{g}^H \mathbf{\Lambda} \mathbf{g}. \end{aligned} \quad (35)$$

Noting (34),

$$\begin{aligned} \mathbf{g}^H \mathbf{W}^H \mathbf{e} - \mathbf{g}^H \Lambda \mathbf{g} &= \mathbf{g}^H \mathbf{W}^H (\mathbf{y} - \mathbf{W} \mathbf{g}) - \mathbf{g}^H \Lambda \mathbf{g} \\ &= \mathbf{g}^H (\mathbf{W}^H \mathbf{y} - \mathbf{W}^H \mathbf{W} \mathbf{g} - \Lambda \mathbf{g}) = \mathbf{0}. \end{aligned} \quad (36)$$

Similarly, $\mathbf{e}^H \mathbf{W} \mathbf{g} - \mathbf{g}^H \Lambda \mathbf{g} = \mathbf{0}$. Thus, $\mathbf{y}^H \mathbf{y} - 2\mathbf{g}^H \Lambda \mathbf{g} = \mathbf{g}^H \mathbf{W}^H \mathbf{W} \mathbf{g} + \mathbf{e}^H \mathbf{e}$, or

$$\mathbf{e}^H \mathbf{e} + \mathbf{g}^H \Lambda \mathbf{g} = \mathbf{y}^H \mathbf{y} - \mathbf{g}^H \mathbf{W}^H \mathbf{W} \mathbf{g} - \mathbf{g}^H \Lambda \mathbf{g}. \quad (37)$$

APPENDIX B

The complex-valued version of the modified Gram-Schmidt orthogonalisation procedure also calculates the \mathbf{A} matrix row by row and orthogonalises Φ as follows: at the l -th stage make the columns ϕ_i , $l+1 \leq i \leq M$, orthogonal to the l -th column and repeat the operation for $1 \leq l \leq M-1$. Specifically, denoting $\phi_i^{(0)} = \phi_i$, $1 \leq i \leq M$, then for $l = 1, 2, \dots, M-1$

$$\left. \begin{aligned} \mathbf{w}_l &= \phi_l^{(l-1)}, \\ a_{l,i} &= \mathbf{w}_l^H \phi_i^{(l-1)} / (\mathbf{w}_l^H \mathbf{w}_l), \quad l+1 \leq i \leq M, \\ \phi_i^{(l)} &= \phi_i^{(l-1)} - a_{l,i} \mathbf{w}_l, \quad l+1 \leq i \leq M. \end{aligned} \right\} \quad (38)$$

The last stage of the procedure is simply $\mathbf{w}_M = \phi_M^{(M-1)}$. The elements of \mathbf{g} are computed by transforming $\mathbf{y}^{(0)} = \mathbf{y}$ in a similar way:

$$\left. \begin{aligned} g_l &= \mathbf{w}_l^H \mathbf{y}^{(l-1)} / (\mathbf{w}_l^H \mathbf{w}_l + \lambda_l), \\ \mathbf{y}^{(l)} &= \mathbf{y}^{(l-1)} - g_l \mathbf{w}_l, \end{aligned} \right\} 1 \leq l \leq M. \quad (39)$$

This orthogonalisation scheme can be used to derive a simple and efficient algorithm for selecting subset models in a forward-regression manner, just as in the real-valued case. First define

$$\Phi^{(l-1)} = [\mathbf{w}_1 \dots \mathbf{w}_{l-1} \quad \phi_l^{(l-1)} \dots \phi_M^{(l-1)}]. \quad (40)$$

If some of the columns $\phi_l^{(l-1)}, \dots, \phi_M^{(l-1)}$ in $\Phi^{(l-1)}$ have been interchanged, this will still be referred to as $\Phi^{(l-1)}$ for notational convenience. The l -th stage of the selection procedure is given as follows.

Step 1. For $l \leq i \leq M$, compute

$$\left. \begin{aligned} g_l^{(i)} &= \left(\phi_i^{(l-1)} \right)^H \mathbf{y}^{(l-1)} / \left(\left(\phi_i^{(l-1)} \right)^H \phi_i^{(l-1)} + \lambda_i \right), \\ [\text{crerr}]_l^{(i)} &= \left(\left| g_l^{(i)} \right|^2 \left(\left(\phi_i^{(l-1)} \right)^H \phi_i^{(l-1)} + \lambda_i \right) \right. \\ &\quad \left. + \beta \log \left(\left(\phi_i^{(l-1)} \right)^H \phi_i^{(l-1)} \right) \right) / (\mathbf{y}^H \mathbf{y}). \end{aligned} \right\}$$

Step 2. Find

$$[\text{crerr}]_l = [\text{crerr}]_l^{(i_l)} = \max\{[\text{crerr}]_l^{(i)}, l \leq i \leq M\}.$$

Then the i_l -th column of $\Phi^{(l-1)}$ is interchanged with the l -th column of $\Phi^{(l-1)}$, the i_l -th column of \mathbf{A} is interchanged with the l -th column of \mathbf{A} up to the $(l-1)$ -th row, and the i_l -th element of λ is interchanged with the l -th element

of λ . This effectively selects the i_l -th candidate as the l -th regressor in the subset model.

Step 3. Perform the orthogonalisation as indicated in (38) to derive the l -th row of \mathbf{A} and to transform $\Phi^{(l-1)}$ into $\Phi^{(l)}$. Calculate g_l and update $\mathbf{y}^{(l-1)}$ into $\mathbf{y}^{(l)}$ in the way shown in (39).

The selection is terminated at the n_s stage when the criterion (23) is satisfied and this produces a subset model containing n_s significant regressors. The algorithm described here is in its standard form. A fast implementation can be adopted, just as shown in the real-valued case [16], to reduce complexity.

REFERENCES

- [1] S. Chen, S. McLaughlin and B. Mulgrew, "Complex-valued radial basis function network, Part I: Network architecture and learning algorithms," *Signal Processing*, vol.35, pp.19-31, 1994.
- [2] S. Chen, S. McLaughlin and B. Mulgrew, "Complex-valued radial basis function network, Part II: Application to digital communications channel equalisation," *Signal Processing*, vol.36, pp.175-188, 1994.
- [3] I. Cha and S.A. Kassam, "Channel equalization using adaptive complex radial basis function networks," *IEEE J. Selected Areas in Communications*, vol.13, no.1, pp.122-131, 1995.
- [4] Q. Gan, P. Saratchandran, N. Sundararajan and K.R. Subramanian, "A complex valued radial basis function network for equalization of fast time varying channels," *IEEE Trans. Neural Networks*, vol.10, no.4, pp.958-960, 1999.
- [5] J. Deng, N. Sundararajan and P. Saratchandran, "Communication channel equalization using complex-valued minimal radial basis function neural networks," *IEEE Trans. Neural Networks*, vol.13, no.3, pp.687-696, 2002.
- [6] C. Botoca and G. Budura, "Symbol decision equalizer using a radial basis functions neural network," in *Proc. 7th WSEAS Int. Conf. Neural Networks* (Cavta, Croatia), June 12-14, 2006, pp.79-84.
- [7] S. Chen, S.A. Billings and W. Luo, "Orthogonal least squares methods and their application to non-linear system identification," *Int. J. Control*, vol.50, no.5, pp.1873-1896, 1989.
- [8] S. Chen, C.F.N. Cowan and P.M. Grant, "Orthogonal least squares learning algorithm for radial basis function networks," *IEEE Trans. Neural Networks*, vol.2, no.2, pp.302-309, 1991.
- [9] S. Chen, X. Hong and C.J. Harris, "Sparse kernel regression modeling using combined locally regularized orthogonal least squares and D-optimality experimental design," *IEEE Trans. Automatic Control*, vol.48, no.6, pp.1029-1036, 2003.
- [10] S. Chen, "Locally regularised orthogonal least squares algorithm for the construction of sparse kernel regression models," in *Proc. 6th Int. Conf. Signal Processing* (Beijing, China), Aug.26-30, 2002, Vol.2, pp.1229-1232.
- [11] S. Chen, "Local regularization assisted orthogonal least squares regression," *Neurocomputing*, vol.69, no.4-6, pp.559-585, 2006.
- [12] A.C. Atkinson and A.N. Donev, *Optimum Experimental Designs*. Oxford: Clarendon Press, 1992.
- [13] X. Hong and C.J. Harris, "Nonlinear model structure design and construction using orthogonal least squares and D-optimality design," *IEEE Trans. Neural Networks*, Vol.13, No.5, pp.1245-1250, 2002.
- [14] D.J.C. MacKay, "Bayesian interpolation," *Neural Computation*, vol.4, no.3, pp.415-447, 1992.
- [15] S. Pupolin and L.J. Greenstein, "Performance analysis of digital radio links with nonlinear transmit amplifier," *IEEE J. Selected Areas Communications*, vol.SAC-5, no.3, pp.534-546, 1987.
- [16] S. Chen and J. Wigger, "Fast orthogonal least squares algorithm for efficient subset model selection," *IEEE Trans. Signal Processing*, vol.43, no.7, pp.1713-1715, 1995.