Monograph:
Chen, S., Billings, S.A., Cowan, C.F.N. et al. (1989) Non-Linear Systems Identification Using Radial Basis Functions. Research Report. Acse Report 378 . Dept of Automatic Control and System Engineering. University of Sheffield

NON-LINEAR SYSTEMS IDENTIFICATION USING RADIAL BASIS FUNCTIONS

S. Chen†, S.A. Billings‡, C.F.N. Cowan† and P.M. Grant†

†Department of Electrical Engineering
University of Edinburgh
Mayfield Road
Edinburgh EH9 3JL
Scotland

‡Department of Control Engineering
University of Sheffield
Mappin street
Sheffield S1 3JD
England

December 1989

Research Report No 378

Abstract

This paper investigates the identification of discrete-time non-linear systems using radial basis functions. A forward regression algorithm based on an orthogonal decomposition of the regression matrix is employed to select a suitable set of radial-basis-function centres from a large number of possible candidates, and this provides, for the first time, a fully automatic selection procedure for identifying parsimonious radial-basis-function models of structure-unknown non-linear systems. The relationship between neural networks and radial basis functions is discussed and the application of the algorithms to real data is included to demonstrate the effectiveness of this approach.

1. Introduction

Feedforward layered neural networks have been widely used in many areas of signal processing and the advantages of realizing complicated non-linear mappings using neural networks are now well-known (Cybenko 1989, Funahashi 1989). Chen et al (1989b) investigated the identification of discrete-time non-linear systems using a two-layered neural network model. The results obtained in this previous work suggested that the neural network approach is very flexible and it offers a viable alternative to conventional methods, such as a polynomial expansion, for modelling complex non-linear systems. Generally neural network models are highly non-linear-in-the-parameters and parameter estimation must be based on non-linear optimisation techniques, such as the prediction error estimation method (Chen et al 1989b) which require intensive computation. Another consequence of highly non-linear-in-the-parameters models is that the parameter estimate may become trapped at a local minimum of the chosen optimisation criterion during the estimation process using a gradient based algorithm.

Using a feedforward neural network to model complex data can be considered as performing a curve fitting operation in a multi-dimensional space. Broomhead and Lowe (1988) adopted this viewpoint and revealed explicitly the connections between feedforward layered neural networks and radial basis functions. The method of radial basis functions is a traditional technique for strict interpolation in multi-dimensional space (Powell 1985, Micchelli 1986). If the strict interpolating restrictions are relexed, a radial-basis-function model can be regarded as a special two-layered network which is linear-in-the-parameters (Broomhead and Lowe 1988), assuming that there is a separate mechanism for selecting a suitable set of radial-basis-function centres and non-linearities. Once the centres have been

chosen, the hidden layer performs a fixed non-linear transformation and which maps the input space onto a new space. The output layer implements a linear combiner on this new space, and the adjustable weights of hidden-to-output layers can therefore be determined using the linear least squares method, which is an obvious advantage of this approach.

The non-linearities within a radial-basis-function model can be chosen from a few typical non-linear functions. Usually the centres are either chosen to be a subset of the data or distributed uniformly within the region of the input space for which there is data. The set of possible candidate centres in each case can be very large and an arbitrary selection by a human operator may not produce a satisfactory procedure. A possible answer to this problem is to regard the centres as adjustable parameters and to determine their values by non-linear optimisation (Lowe 1989). This however will destroy the linear-in-the-parameters structure and is equivalent to employing a two-layered neural network.

The present study adopts a different approach to the problem of selecting centres. Because a centre can be considered as corresponding to a regressor in a linear regression model, radial basis functions can be investigated within the framework of the extended model set introduced by Billings and Chen (1989). This means that the orthogonal forward regression routine (Chen et al 1989a, Billings and Chen 1989) can be readily applied to select a suitable set of centres (regressors) from a large set of candidates. Such an approach provides, for the first time, a practical and efficient identification procedure for fitting radial-basis-function models to non-linear systems. The application of this identification procedure to an automotive diesel engine, sunspot numbers and a heat exchanger is included and the estimates are compared with results obtained from neural network models.

2. Non-linear system representation

In this paper the class of non-linear systems which can be described in terms of some non-linear functional expansion of lagged inputs and outputs is considered. Mathematically such systems can be represented by

$$y(t)=f_s(y(t-1),\ldots,y(t-n_y),u(t-1),\ldots,u(t-n_u))+e(t) \tag{1}$$

where

$$y(t)=\begin{bmatrix} y_1(t) \\ \cdot \\ \cdot \\ \cdot \\ y_m(t) \end{bmatrix},\quad u(t)=\begin{bmatrix} u_1(t) \\ \cdot \\ \cdot \\ \cdot \\ u_r(t) \end{bmatrix},\quad e(t)=\begin{bmatrix} e_1(t) \\ \cdot \\ \cdot \\ \cdot \\ e_m(t) \end{bmatrix} \tag{2}$$

are the system output, input and additive white noise vectors respectively; n_y and n_u are the maximum lags in the output and input respectively; and $f_s(.)$ is some vector-valued non-linear function.

The functional form $f_s(.)$ for a practical system is generally very complicated and is rarely available. A model has to be constructed based on some known simpler function and the present study uses a radial-basis-function expansion to model the input-output relationship (1).

Equation (1) is a simplified version of the more general system description known as the NARMAX (Non-linear AutoRegressive Moving Average with eXogenous inputs) model (Leontaritis and Billings 1985, Chen and Billings 1989a), which can be written as follows

$$y(t)=f_s(y(t-1),\ldots,y(t-n_y),u(t-1),\ldots,u(t-n_u),e(t-1),\ldots,e(t-n_e))+e(t) \quad (3)$$

The identification procedure developed in the present study can be extended to this general model.

3. Modelling by radial basis functions

A radial-basis-function model with n inputs and m outputs implements a mapping $f_r:\mathbf{R}^n\rightarrow\mathbf{R}^m$ according to

$$f_r(\mathbf{x})=\Lambda_0+\sum_{j=1}^{M_c}\Lambda_j\phi(||\mathbf{x}-\mathbf{c}_j||) \quad (4)$$

where $\mathbf{x}\in\mathbf{R}^n$, $\phi(.)$ is a function from $\mathbf{R}^+$ to $\mathbf{R}$, $||.||$ denotes the Euclidean norm, $\Lambda_j\in\mathbf{R}^m$ $j=1,\ldots,M_c$ are the weight vectors, $\Lambda_0\in\mathbf{R}^m$ is a constant (bias) vector and $\mathbf{c}_j\in\mathbf{R}^n$ $j=1,\ldots,M_c$ are the radial-basis-function centres. Let

$$\Lambda_j=[\lambda_{1j}\cdots\lambda_{mj}]^T \quad j=0,1,\ldots,M_c \quad (5)$$

Then eqn.(4) can be written in a decomposed form

$$f_{r_i}(\mathbf{x})=\lambda_{i0}+\sum_{j=1}^{M_c}\lambda_{ij}\phi(||\mathbf{x}-\mathbf{c}_j||) \quad i=1,\ldots,m \quad (6)$$

The functional form $\phi(.)$ and the centres $\mathbf{c}_j$ are assumed to have been fixed. If a set of the inputs $\mathbf{x}(t)$ and the corresponding desired outputs $\mathbf{d}(t)$ for $t=1$ to N are provided, the values of the weights λ_{ij} for $i=1,\ldots,m$ and $j=0,1,\ldots,M_c$ can be determined using the linear least squares method.

The radial-basis-function expansion is a traditional technique for interpolating in multi-dimensional space (Powell 1985, Micchelli 1986). The generalization to the form of

eqn.(4) or (6) was introduced by Broomhead and Lowe (1988). The relationship between the radial-basis-function expansion and a two-layered neural network can be clearly seen by realizing eqn.(4) or (6) in the two-layered network structure depicted in Fig.1. The diagram of Fig.1 can also be used to represent a general two-layered neural network. The difference lies in the structure of the hidden layer. For a radial-basis-function model, each node in the hidden layer performs a fixed non-linear transformation on the inputs and it contains no adjustable weights. Each hidden node in a two-layered neural network however would contain an adjustable weight vector $\tilde{\Lambda}_j$ and an adjustable threshold parameter μ_j, and the output of the node would be given as $a(\tilde{\Lambda}_j^T \mathbf{x} + \mu_j)$ where $a(.)$ is a non-linear function from $\mathbf{R}$ to $\mathbf{R}$. A radial-basis-function model can therefore be viewed as a special two-layered network which is linear-in-the-parameters.

The objective of the present study is to use radial basis functions to model non-linear systems described by eqn.(1). Define

$$n = mn_y + rn_u \tag{7}$$

and

$$\mathbf{x}(t) = [y^T(t-1)\cdots y^T(t-n_y)u^T(t-1)\cdots u^T(t-n_u)]^T \tag{8}$$

The radial-basis-function expansion $f_r(\mathbf{x}(t))$ can then be used as the one-step-ahead prediction for $y(t)$. The identification thus involves determining values of λ_{ij} based on the input-output observations $\{u(t), y(t)\}_{t=1}^{N}$.

The linear-in-the-parameters structure of the radial-basis-function expansion is of course obtained under the assumption that the function $\phi(.)$ and the centres have been fixed. Typical choices for $\phi(.)$ are the thin-plate-spline function

$$\phi(v) = v^2 \log(v) \tag{9}$$

the gaussian function

$$\phi(v) = exp(-v^2/\beta^2) \tag{10}$$

and the multi-quadric function

$$\phi(v) = \sqrt{v^2 + \beta^2} \tag{11}$$

where β is a real constant. The centres $\mathbf{c}_j$ $j = 1,\ldots,M_c$ must suitably sample the domain of the model inputs and they are usually chosen either to be a subset of the data or distributed uniformly in the input domain. In the former case the possible candidates for $\mathbf{c}_j$ are the data points $\mathbf{x}(t)$, the number of which is often very large. In the latter case the choices are

in effect infinitely many. Assume that the number of possible candidate centres is M, one thousand say. An adequate radial-basis-function expansion may require only M_c, say thirty centres if they are wisely chosen. These centres may be selected by an operator who inspects the candidate centres and picks those he or she thinks are appropriate. It is, however, desirable that this selection should be automated and preferably optimal in some sense to ensure that an adequate radial-basis-function model is obtained. By combining the radial-basis-function expansion with the extended model set representation introduced in Billings and Chen (1989), a new algorithm can be formulated to achieve this aim.

4. Extended model set representation

Linear-in-the-parameter models can be represented in the following general form

$$z(t)=\sum_{j=1}^{M} p_j(t)\theta_j+\xi(t) \tag{12}$$

This is a linear regression model where the $p_j(t)$ are the regressors, $z(t)$ is known as the dependent variable, $\xi(t)$ is some modelling error which is assumed to be uncorrelated with the regressors, and the θ_j are unknown parameters to be estimated. A constant term is included in eqn.(12) by setting the corresponding $p_j(t)=1$. In the context of system identification, $z(t)$ is $y_i(t)$, the i-th element of $y(t)$; $\xi(t)$ is $\epsilon_i(t)$, the i-th element of the one-step-ahead prediction error or residual; and the $p_j(t)$ are some fixed non-linear functions of lagged outputs and inputs, that is

$$p_j(t)=p_j(\mathbf{x}(t)) \tag{13}$$

where $\mathbf{x}(t)$ is defined in eqn.(8).

The primary choices for the regressors are monomials of $\mathbf{x}(t)$. Other non-linear functions that commonly exist in non-linear systems can be included as the choices for $p_j(t)$ and some examples are shown in (Billings and Chen 1989). The set that contains a variety of choices of regressors is referred to as the extended model set (Billings and Chen 1989) since it provides a much richer description of non-linear systems than the polynomial model set alone. Effective modelling can be achieved by selecting a parsimonious subset model from this extended model set.

It is obvious that a given centre $\mathbf{c}_j$ with a fixed non-linearity $\phi(.)$ corresponds to a regressor $p_j(t)$ in eqn.(12). The radial-basis-function expansion is therefore a particular choice of the extended model set representation. The problem of how to select a subset of centres from a large number of candidate centres can thus be regarded as an example of

how to select a subset of significant regressors from a given extended model set. An efficient solution to this general problem has been developed (Chen et al 1989a, Billings and Chen 1989), and is summarized in the following section.

5. The orthogonal forward regression procedure

Assume that the data length is N. Collecting eqn.(12) from $t=1$ to N together yields the matrix equation

$$\mathbf{z}=\mathbf{P}\Theta+\Xi \tag{14}$$

where

$$\mathbf{z}=\begin{bmatrix} z(1) \\ \cdot \\ \cdot \\ \cdot \\ z(N) \end{bmatrix},\quad \mathbf{P}=[\mathbf{p}_1\cdots\mathbf{p}_M],\quad \Theta=\begin{bmatrix} \theta_1 \\ \cdot \\ \cdot \\ \cdot \\ \theta_M \end{bmatrix},\quad \Xi=\begin{bmatrix} \xi(1) \\ \cdot \\ \cdot \\ \cdot \\ \xi(N) \end{bmatrix} \tag{15}$$

and

$$\mathbf{p}_j=[p_j(1)\cdots p_j(N)]^T \quad j=1,\ldots,M \tag{16}$$

The matrix $\mathbf{P}$ is known as the regression matrix. An orthogonal decomposition of $\mathbf{P}$ is given as

$$\mathbf{P}=\mathbf{WA} \tag{17}$$

Here

$$\mathbf{A}=\begin{bmatrix} 1 & \alpha_{12} & \alpha_{13} & \cdots & \alpha_{1M} \\ & 1 & \alpha_{23} & \cdots & \alpha_{2M} \\ & & \cdot & \cdot & \cdot \\ & & & \cdot \cdot & \cdot \\ & & & \cdot \cdot & \cdot \\ & & & 1 & \alpha_{M-1M} \\ & & & & 1 \end{bmatrix} \tag{18}$$

is an $M\times M$ unit upper triangular matrix and

$$\mathbf{W}=[\mathbf{w}_1\cdots\mathbf{w}_M] \tag{19}$$

is an $N\times M$ matrix with orthogonal columns that satisfy

$$\mathbf{W}^T\mathbf{W}=\mathbf{D} \tag{20}$$

and $\mathbf{D}$ is a positive diagonal matrix

$$\mathbf{D}=diag\{d_1\cdots d_M\} \tag{21}$$

with

$$d_j = <\mathbf{w}_j, \mathbf{w}_j> \quad j = 1,\ldots,M \tag{22}$$

where $<.,.>$ denotes the inner product, that is

$$<\mathbf{w}_i, \mathbf{w}_j> = \mathbf{w}_i^T \mathbf{w}_j = \sum_{t=1}^{N} w_i(t) w_j(t) \tag{23}$$

Equation (14) can be rearranged as

$$\mathbf{z} = \left(\mathbf{P}\mathbf{A}^{-1}\right)\left(\mathbf{A}\,\Theta\right) + \Xi = \mathbf{W}\mathbf{g} + \Xi \tag{24}$$

where

$$\mathbf{A}\,\Theta = \mathbf{g} \tag{25}$$

Because $\xi(t)$ is uncorrelated with the regressors, it is easy to show that

$$\mathbf{g} = \mathbf{D}^{-1}\mathbf{W}^T\mathbf{z} \tag{26}$$

or

$$g_j = \frac{<\mathbf{w}_j, \mathbf{z}>}{<\mathbf{w}_j, \mathbf{w}_j>} \quad j = 1,\ldots,M \tag{27}$$

Alternatively by normalizing the columns of $\mathbf{W}$ and augmenting the resulting matrix with $N-M$ further orthonormal columns to make up a full set of N orthonormal vectors for the N-dimensional Euclidean space, an equivalent decomposition to (17) is obtained

$$\mathbf{P} = \overline{\mathbf{W}} \begin{bmatrix} \mathbf{Q} \\ 0 \end{bmatrix} \tag{28}$$

where $\mathbf{Q}$ is an $M \times M$ upper triangular matrix with positive diagonal elements and

$$\overline{\mathbf{W}} = [\overline{\mathbf{w}}_1 \cdots \overline{\mathbf{w}}_N] \tag{29}$$

is an $N \times N$ orthogonal matrix that satisfies

$$<\overline{\mathbf{w}}_i, \overline{\mathbf{w}}_j> = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases} \tag{30}$$

Multiplying $\overline{\mathbf{W}}^T$ by $\mathbf{z}$ yields

$$\overline{\mathbf{W}}^T\mathbf{z} = [\overline{z}_1 \cdots \overline{z}_M\, \overline{z}_{M+1} \cdots \overline{z}_N]^T \tag{31}$$

It is readily seen that the parameter estimate $\hat{\Theta}$ can be obtained from

$$\mathbf{Q}\,\Theta = [\overline{z}_1 \cdots \overline{z}_M]^T \tag{32}$$

Several orthogonal least squares methods can be used to obtain the triangular system (25) or (32) and, consequently, to solve for the parameter estimate $\hat{\Theta}$. These include the classical Gram-Schmidt and modified Gram-Schmidt methods (Björck 1967), and the

Householder transformation method (Golub 1965).

As mentioned previously M, the number of all the candidate regressors, can be very large even though adequate modelling may only require $M_s (<<M)$ significant regressors. By modifying and augmenting the orthogonal least squares estimators, an efficient forward regression procedure has been derived (Chen et al 1989a) to identify these significant regressors. From (24), the sum of squares of the dependent variable is

$$<\mathbf{z},\mathbf{z}>=\sum_{j=1}^{M} g_j^2<\mathbf{w}_j,\mathbf{w}_j>+<\Xi,\Xi> \tag{33}$$

The error reduction ratio due to $\mathbf{w}_j$ is thus expressed as the proportion of the dependent variable variance explained by $\mathbf{w}_j$

$$[err]_j=\frac{g_j^2<\mathbf{w}_j,\mathbf{w}_j>}{<\mathbf{z},\mathbf{z}>} \quad 1\le j\le M \tag{34}$$

If the decomposition (28) is employed, a similar definition for error reduction ratio is given as

$$[err]_j=\frac{(\bar{z}_j)^2}{<\mathbf{z},\mathbf{z}>} \quad 1\le j\le M \tag{35}$$

The error reduction ratio offers a simple and effective means of selecting a subset of significant regressors from a large number of candidates in a forward-regression manner. At the i-th step, a regressor is selected if it produces the largest value of $[err]_i$ among the rest of candidates. The selection procedure can be terminated when

$$1-\sum_{i=1}^{M_s}[err]_i<\rho \tag{36}$$

where $0<\rho<1$ is a desired tolerance, and this leads to a subset of M_s regressors. The parameter estimate $\hat{\Theta}_s$ for this subset model is then computed from

$$\mathbf{A}_s\Theta_s=\mathbf{g}_s \quad or \quad \mathbf{Q}_s\Theta_s=[\bar{z}_1\cdots\bar{z}_{M_s}]^T \tag{37}$$

where $\mathbf{A}_s$ and $\mathbf{Q}_s$ are the $M_s\times M_s$ unit or positive upper triangular matrices respectively. The detailed description of the selection procedure can be found in (Chen et al 1989c, Billings and Chen 1989).

It is obvious that $1-\sum[err]_i$ is the proportion of the unexplained dependent variable variance. The tolerance ρ is an important instrument which affects both the prediction accuracy and complexity of the final model. Notice that an appropriate value for ρ can be learned during the forward regression procedure so that the regressor selection becomes an automatic procedure. This aspect is fully explained in (Billings and Chen 1989).

The criterion (36) emphasizes only the performance of the model (variance of residuals). Because a more accurate performance is often achieved at the expense of using a more complex model a trade-off between the performance and complexity of the model is often appropriate. Akaike's information criterion which provides a compromise between the performance and the number of parameters takes the form

$$AIC(\eta)=N\log(\sigma_\xi^2)+M_s\eta \tag{38}$$

where σ_ξ^2 is the variance of the residual $\xi(t)$ and η is the critical value of the chi-square distribution with one degree of freedom for a given level of significance. $\eta=4.0$ corresponds to the significance level of 0.046 and is often a suitable choice (Leontaritis and Billings 1987). If the criterion (38) or (36) is coupled with model validity tests (Bohlin 1978, Billings and Voon 1986, Leontaritis and Billings 1987, Billings et al 1989b), an adequate subset model can often be found very quickly.

In the above discussion it is implicitly assumed that $\mathbf{P}^T\mathbf{P}$ is positive definite so that nonsingular $\mathbf{D}$ or $\mathbf{Q}$ exist. This assumption however was introduced only for the benefit of the derivation and is not really required. Notice that $<\mathbf{w}_k,\mathbf{w}_k>=0$ means that $\mathbf{p}_k$ is a linear combination of $\mathbf{p}_1$ to $\mathbf{p}_{k-1}$. Therefore if $<\mathbf{w}_k,\mathbf{w}_k>$ is less than a small predetermined threshold, the regressor $\mathbf{p}_k$ will not be selected and numerical ill-conditioning can thus be avoided. A more detailed discussion of this aspect is given in (Billings et al 1989b). The inclusion of this mechanism in the selection procedure has an important implication for fitting radial-basis-function models. If the centres are improperly chosen (for example, some centres are too close), numerical ill-conditioning will occur. To combat ill-conditioning, Broomhead and Lowe (1988) suggested the use of more robust techniques such as singular valued decomposition (e.g. Golub and Reinsch 1970) for parameter estimation. The orthogonal selection procedure discussed above should avoid numerical ill-conditioning problems and this is obviously an advantage.

6. Real data identification

This section applies the orthogonal forward regression procedure to fit radial-basis-function models to some real industrial data. All the data sets concerned were generated from single-input single-output ($r=1$ and $m=1$) non-linear systems. For multi-input multi-output systems, the identification procedure is similar because, as shown in eqn.(6), each element of the system output vector requires one radial-basis-function expansion and each model can be identified separately. The non-linearity $\phi(.)$ was chosen to be the thin-

plate-spline function (9). Some definitions are first introduced. Given a radial-basis-function expansion $f_r(.)$, the one-step-ahead prediction is defined as

$$\hat{y}(t)=f_r(\mathbf{x}(t))=f_r(y(t-1),\ldots,y(t-n_y),u(t-1),\ldots,u(t-n_u)) \tag{39}$$

The prediction error or residual is accordingly given by

$$\epsilon(t)=y(t)-\hat{y}(t) \tag{40}$$

The model deterministic output is defined by

$$\hat{y}_d(t)=f_r(\mathbf{x}_d(t))=f_r(\hat{y}_d(t-1),\ \ldots,\hat{y}_d(t-n_y),u(t-1),\ldots,u(t-n_u)) \tag{41}$$

and the deterministic error is then given as

$$\epsilon_d(t)=y(t)-\hat{y}_d(t) \tag{42}$$

Model validity tests employed are the following simple correlation tests (Billings and Voon 1986, Billings and Chen 1989)

$$\left.\begin{array}{l}\Phi_{\epsilon\epsilon}(k)\\ \Phi_{u\epsilon}(k)\\ \Phi_{\epsilon(\epsilon u)}(k)\\ \Phi_{u^2\epsilon}(k)\\ \Phi_{u^2\epsilon^2}(k)\end{array}\right\} \tag{43}$$

Example 1. The data was collected from a turbocharged automotive diesel engine in a low engine speed test. A detailed description of the experiment design can be found in work by Billings et al (1989a). The data set, named as mob6shv2.dta in (Billings et al 1989a), contains 410 points and has been shown to be inherently non-linear.

The dimension of the radial-basis-function input was set to $n=n_y+n_u=1+2$. All the centres were to be chosen from the data, thus giving rise to a total of more than 400 candidate centres. The orthogonal selection procedure was used to identify a subset of significant centres and the selection was terminated when AIC(4.0) achieved a minimum value. The final model chosen contained a constant term and $M_c=34$ centres. The coefficients associated with the estimated model are illustrated in Table 1. The model thus takes the form

$$\hat{y}(t)=3.8036+9.3393\phi(||\mathbf{x}(t)-\mathbf{c}_1||)-\cdots-8.6896\phi(||\mathbf{x}(t)-\mathbf{c}_{34}||)$$

where $\phi(.)$ is defined in eqn.(9) and

$$||\mathbf{x}(t)-\mathbf{c}_i||=\sqrt{(y(t-1)-c_{1i})^2+(u(t-1)-c_{2i})^2+(u(t-2)-c_{3i})^2}\quad i=1\ to\ 34$$

c_{1i}, c_{2i} and c_{3i} for $i=1$ to 34 are also listed in Table 1. The model validity tests are displayed in Fig.2, and the data set and the model response are shown in Fig.3.

A two-layer neural network model was also fitted to the data as a comparison. The difficulty encountered in fitting the neural network model to this data set highlights a possible pitfall associated with models which have a highly non-linear-in-the-parameter structure. The input order of the neural network model was chosen to be $n=n_y+n_u=1+2$ and the number of hidden nodes was set to $n_h=7$. This gave a total of 35 parameters. The activation function $a(.)$ for each hidden node was chosen to be the sigmoid function

$$a(v)=\frac{1}{1+exp(-v)} \tag{44}$$

The activation function for the output node was chosen to be linear and no threshold parameter was introduced in the output node. The prediction error estimation algorithm (Chen et al 1989b) was used to produce parameter estimates. The initial parameter values were set randomly between -0.3 to 0.3. It was found that the algorithm quickly stopped, possibly trapped in a local minimum. Random perturbation had to be applied in order to continue the identification process. This situation occurred again and again. Finally after several perturbations and more than two hundred iterations, we gave up and accepted the results. The estimated model is given in Table 2, and the model validity tests and the model response are illustrated in Figs. 4 and 5 respectively. Most of the model validity tests, Fig.4, are well outside the 95% confidence bands indicating that the model is deficient.

We emphasize that it should be possible with this neural network model to provide a much better fit to the data than the results shown in Figs. 4 and 5. The problem is how to obtain an adequate parameter estimate. A gradient-based algorithm can fail in highly non-linear-in-the-parameter cases as shown here. Other algorithms (e.g. Hoptroff and Hall 1989) capable of finding a global minimum may have to be used. Unfortunately these algorithms are very complicated and the associated huge increase in computation often deter their application.

Example 2. The time series of annual sunspot numbers for the years 1700 to 1955 are considered. These 256 observations are listed in Appendix A1 of Tong (1983). The first 221 observations were used to fit a model and the last 35 observations were used as the test set.

The dimension of the centres was set to $n=n_y=9$ and all the data points $\mathbf{x}(t_i)$ $t_i=10$ to 221 were used as candidate centres. The error reduction ratio criterion (36) was

employed to terminate the orthogonal regression procedure. An appropriate tolerance $\rho=0.04$ was found during the regression procedure and the final selected model contained 25 centres (the constant term was not selected). The estimated radial-basis-function model thus takes the form

$$\hat{y}(t)=\sum_{i=1}^{25}\hat{\lambda}_i\,\phi(||\mathbf{x}(t)-\mathbf{c}_i||)$$

where

$$||\mathbf{x}(t)-\mathbf{c}_i||=\sqrt{\sum_{k=1}^{9}(y(t-k)-c_{ki})^2}\quad i=1\ to\ 25$$

The selected centres $\mathbf{c}_i$ and the values of the estimates $\hat{\lambda}_i$ are listed in Tables 3 and 4 respectively. The response of this radial-basis-function model is shown in Fig.6. It is interesting to see that the unforced response of this model, which is a sustained osillation with an approximate 11-year period, is similar to those produced by a two-layer neural network model (Chen et al 1989b) and a polynomial subset model (Chen and Billings 1989b). The neural network model and the subset polynomial model are given in Tables 5 and 6 as references.

Example 3. The data was generated from a heat exchanger and contains 996 points. A description of this process and the experiment design is given in (Billings and Fadzil 1985). The first 500 points of the data, depicted in Fig.7, were used as the identification set and the rest of the data as the test set.

The dimension of the centres was chosen to be $n=n_y+n_u=5+5$. 300 candidate centres were generated by randomly selecting points from the region

$$[y_{\min},\ y_{\max}]^{n_y}\times[u_{\min},\ u_{\max}]^{n_u}\subset\mathbf{R}^{n_y}\times\mathbf{R}^{n_u}$$

where $y_{\min}$, $u_{\min}$, $y_{\max}$ and $u_{\max}$ are the minimum and maximum values of the system output and input respectively. The error reduction ratio criterion (36) was used to stop the selection procedure and an appropriate value for ρ was found to be 0.0007. The orthogonal regression procedure selected a constant term and 27 centres. The estimated model takes the form

$$\hat{y}(t)=\hat{\lambda}_0+\sum_{i=1}^{27}\hat{\lambda}_i\,\phi(||\mathbf{x}(t)-\mathbf{c}_i||)$$

where

$$||\mathbf{x}(t)-\mathbf{c}_i||=\sqrt{\sum_{k=1}^{5}(y(t-k)-c_{ki})^2+\sum_{k=6}^{10}(u(t+5-k)-c_{ki})^2}\quad i=1\ to\ 27$$

The chosen centres and the associated coefficients are shown in Tables 7 and 8 respectively. The model validity tests for both the identification and test sets are shown in Figs. 8 and 9 respectively, and the test set and model response for the test set are illustrated in Fig.10. The results shown in Figs. 8 to 10 are extremely similar to those produced by a two-layer neural network model (Example 3 in Chen et al 1989b). As a comparison this neural network model is given in Table 9.

7. Discussion for further research

If modelling complex data is regarded as performing a curve fitting in a high dimensional space, the two basic concepts in neural networks, namely learning and generalization, have particularly simple meanings. From this viewpoint, learning corresponds to producing a surface in multi-dimensional space that fits to the set of data in some best sense, and generalization is equivalent to interpolating the test data set on this fitting surface. When fitting radial-basis-function models, it is possible that the learning error will be continuously reduced if we keep adding more centres. At some point, however, the generalization error may stop decreasing and may even start to increase as more centres are introduced into the model (see Lowe 1989). This is because at some point the model starts to fit to the noise rather than the underlying process and, although the fit seems to be improved, the generalization error correctly reveals that this is a misjudgement. The performance of generalization is a more "genuine" indicator for the "goodness" of the model. A new criterion for terminating the orthogonal regression procedure can be introduced based on the above discussion. The data set is partitioned as the fitting set and the test set. The former is used in the model selection procedure and the latter is used to compute the generalization error, that is the variance of the residuals over the test set computed based on the selected model. The selection procedure is terminated when this error is minimized. Such an identificaion procedure is of course not restricted to radial-basis-function models.

8. Conclusions

An identification procedure has been developed for discrete-time non-linear systems using a radial-basis-function expansion. By viewing radial basis functions as examples of the extended model set representation, it has been shown that the orthogonal forward regression procedure provides a powerful method for fitting parsimonious radial-basis-function models to practical systems.

Acknowledgment

This work was supported by the U.K. Science and Engineering Research Council.

References

[1] Billings, S.A., and S. Chen (1989). Extended model set, global data and threshold model identification of severely non-linear systems. *Int. J. Control*, (to appear).

[2] Billings, S.A., S. Chen, and R.J. Backhouse (1989a). 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.

[3] Billings, S.A., S. Chen, and M.J. Korenberg (1989b). Identification of MIMO non-linear systems using a forward-regression orthogonal estimator. *Int. J. Control*, Vol.49, No.6, pp.2157-2189.

[4] Billings, S.A., and M.B. Fadzil (1985). The practical identification of systems with nonlinearities. *Proc. of 7th IFAC Symp. on Identification and System Parameter Estimation*, York, U.K., pp.155-160.

[5] Billings, S.A., and W.S.F. Voon (1986). Correlation based model validity tests for non-linear models. *Int. J. Control*, Vol.44, No.1, pp.235-244.

[6] Björck, A. (1967). Solving linear least squares problems by Gram-Schmidt orthogonalization. *Nordisk Tidskr. Informations-Behandling*, Vol.7, pp.1-21.

[7] Bohlin, T. (1978). Maximum-power validation of models without higher-order fitting. *Automatica*, Vol.4, pp.137-146.

[8] Broomhead, D.S., and D. Lowe (1988). Multivariable functional interpolation and adaptive networks. *Complex Systems*, Vol.2, pp.321-355.

[9] Chen, S., and S.A. Billings (1989a). Representation of non-linear systems: the NARMAX model. *Int. J. Control*, Vol.49, No.3, pp.1013-1032.

[10] Chen, S., and S.A. Billings (1989b). Modelling and analysis of non-linear time series. *Int. J. Control*, (to appear).

[11] Chen, S., S.A. Billings, and W. Luo (1989a). Orthogonal least squares methods and their application to non-linear system identification. *Int. J. Control*, (to appear).

[12] Chen, S., S.A. Billings, and P.M. Grant (1989b). Non-linear systems identification using neural networks. *Int. J. Control*, (to appear).

[13] Cybenko, G. (1989). Approximations by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, (to appear).

[14] Funahashi, K. (1989). On the approximate realization of continuous mappings by neural networks. *Neural Networks*, Vol.2, pp.183-192.

[15] Golub, G. (1965). Numerical methods for solving linear least squares problems. *Numerische Mathematik*, Vol.7, pp.206-216.

[16] Golub, G.H., and C. Reinsch (1970). Singular value decomposition and least squares solutions. *Numerische Mathematik*, Vol.14, pp.403-420.

[17] Hoptroff, R.G., and T.J. Hall (1989). Diffusion learning for the multilayer perceptron. *Proc. of 1st IEE Int. Conf. on Artificial Neural Networks*, London, U.K., October 16-18, 1989, pp.390-394.

[18] Leontaritis, I.J., and S.A. Billings (1985). Input-output parametric models for non-linear systems. Part I: deterministic non-linear systems; Part II: stochastic non-linear systems. *Int. J. Control*, Vol.41, No.2, pp.303-344.

[19] Leontaritis, I.J., and S.A. Billings (1987). Model selection and validation methods for non-linear systems. *Int. J. Control*, Vol.45, No.1, pp.311-341.

[20] Lowe, D. (1989). Adaptive radial basis function non-linearities, and the problem of generalisation. *Proc. of 1st IEE Int. Conf. on Artificial Neural Networks*, London, U.K., October 16-18, 1989, pp.171-175.

[21] Micchelli, C.A. (1986). Interpolation of scattered data: distance matrices and conditionally positive definite functions. *Constructive Approximation*, Vol.2, pp.11-22.

[22] Powell, M.J.D. (1985). Radial basis functions for multivariable interpolation: a review. *IMA conf. on Algorithms for the Approximation of Functions and Data*, RMCS Shrivenham.

[23] Tong, H. (1983). *Threshold Models in Non-linear Time Series Analysis*. Lecture Notes in Statistics, Springer-Verlag, New York.

	selected centre			weight estimate	error reduction ratio
0	constant			0.38036E+1	0.97467E+0
1	0.28236E+1	0.37439E+1	0.37439E+1	0.93393E+1	0.11922E-1
2	0.28694E+1	0.37439E+1	0.37439E+1	-0.17323E+2	0.12066E-1
3	0.33887E+1	0.37633E+1	0.37633E+1	0.58415E+1	0.67927E-3
4	0.39310E+1	0.37439E+1	0.45200E+1	-0.84067E-1	0.67316E-4
5	0.40073E+1	0.45006E+1	0.45006E+1	0.39437E+1	0.13198E-3
6	0.47329E+1	0.50051E+1	0.58006E+1	0.16579E+1	0.20961E-3
7	0.37630E+1	0.37439E+1	0.37439E+1	0.18548E+0	0.90146E-4
8	0.40073E+1	0.45200E+1	0.45200E+1	-0.38228E+1	0.30101E-4
9	0.35491E+1	0.37633E+1	0.45394E+1	0.69678E+0	0.14630E-4
10	0.33887E+1	0.37439E+1	0.37439E+1	-0.36682E+1	0.22750E-4
11	0.46183E+1	0.58006E+1	0.58006E+1	-0.53896E-2	0.11112E-4
12	0.32131E+1	0.45006E+1	0.37439E+1	-0.39181E-1	0.24663E-4
13	0.42746E+1	0.58200E+1	0.50439E+1	-0.53673E-1	0.10515E-4
14	0.46183E+1	0.50051E+1	0.58006E+1	0.56102E+1	0.75975E-5
15	0.28923E+1	0.37439E+1	0.37439E+1	0.69630E+1	0.48470E-5
16	0.32284E+1	0.37633E+1	0.37633E+1	-0.19294E+1	0.79217E-6
17	0.46030E+1	0.50051E+1	0.50051E+1	0.10838E+0	0.75915E-6
18	0.30756E+1	0.37439E+1	0.37439E+1	0.19209E+1	0.64292E-6
19	0.46412E+1	0.58200E+1	0.50439E+1	-0.24009E+1	0.42742E-6
20	0.30909E+1	0.45200E+1	0.45200E+1	0.34243E+1	0.62074E-6
21	0.34651E+1	0.45394E+1	0.45394E+1	-0.61526E+0	0.13478E-5
22	0.45572E+1	0.58006E+1	0.50051E+1	-0.10082E+1	0.12400E-5
23	0.33505E+1	0.37633E+1	0.45394E+1	-0.47078E+1	0.74268E-6
24	0.46565E+1	0.50245E+1	0.58006E+1	-0.11648E+2	0.46292E-6
25	0.44656E+1	0.45006E+1	0.45006E+1	-0.43427E+1	0.68426E-6
26	0.44579E+1	0.45200E+1	0.45200E+1	0.42991E+1	0.12320E-5
27	0.43281E+1	0.50245E+1	0.58006E+1	-0.36752E+0	0.36844E-6
28	0.46565E+1	0.50439E+1	0.58200E+1	0.45455E+1	0.99245E-6
29	0.31291E+1	0.45006E+1	0.45006E+1	-0.40826E+1	0.53569E-6
30	0.33047E+1	0.45006E+1	0.45006E+1	0.13578E+1	0.48056E-6
31	0.45954E+1	0.58006E+1	0.50245E+1	0.35441E+1	0.46165E-6
32	0.33887E+1	0.37439E+1	0.45394E+1	0.12561E+2	0.26585E-6
33	0.34804E+1	0.37439E+1	0.37439E+1	-0.11516E+1	0.25359E-6
34	0.34193E+1	0.37439E+1	0.45394E+1	-0.86896E+1	0.23850E-6
variance of residuals σ_ϵ^2				0.37194E-3	

Table 1. RBF model of Example 1

position		parameter estimate
1	threshold of 1st hidden node	0.75616E+0
2	$y(t-1)$ to 1st hidden node	-0.17303E+1
3	$u(t-1)$ to 1st hidden node	-0.14154E+1
4	$u(t-2)$ to 1st hidden node	-0.11941E+1
5	threshold of 2nd hidden node	0.14623E+1
6	$y(t-1)$ to 2nd hidden node	0.16123E+1
7	$u(t-1)$ to 2nd hidden node	0.19052E+1
8	$u(t-2)$ to 2nd hidden node	0.51908E+0
9	threshold of 3rd hidden node	0.83829E+0
10	$y(t-1)$ to 3rd hidden node	-0.56831E+0
11	$u(t-1)$ to 3rd hidden node	0.41929E-1
12	$u(t-2)$ to 3rd hidden node	0.36044E+0
13	threshold of 4th hidden node	0.20483E+0
14	$y(t-1)$ to 4th hidden node	-0.12419E+1
15	$u(t-1)$ to 4th hidden node	-0.14822E+1
16	$u(t-2)$ to 4th hidden node	-0.14106E+1
17	threshold of 5th hidden node	-0.99735E+0
18	$y(t-1)$ to 5th hidden node	-0.10030E+1
19	$u(t-1)$ to 5th hidden node	-0.17959E+1
20	$u(t-2)$ to 5th hidden node	-0.11205E+1
21	threshold of 6th hidden node	-0.22396E+0
22	$y(t-1)$ to 6th hidden node	-0.93772E+0
23	$u(t-1)$ to 6th hidden node	-0.17224E+1
24	$u(t-2)$ to 6th hidden node	-0.22313E+1
25	threshold of 7th hidden node	-0.13540E+1
26	$y(t-1)$ to 7th hidden node	-0.34612E+0
27	$u(t-1)$ to 7th hidden node	0.12376E+0
28	$u(t-2)$ to 7th hidden node	0.57185E+0
29	1st hidden node to output node	0.19615E+1
30	2nd hidden node to output node	0.52446E+1
31	3rd hidden node to output node	-0.10368E+2
32	4th hidden node to output node	0.14119E+1
33	5th hidden node to output node	0.40043E+1
34	6th hidden node to output node	0.28167E+1
35	7th hidden node to output node	0.80448E+1
variance of residuals σ_ϵ^2		0.10167E-2

Table 2. Neural network model of Example 1

i	c_{1i}	c_{2i}	c_{3i}	c_{4i}	c_{5i}
	c_{6i}	c_{7i}	c_{8i}	c_{9i}	
1	0.8500E+1	0.2750E+2	0.4780E+2	0.7090E+2	0.6700E+2
	0.6420E+2	0.4960E+2	0.3630E+2	0.1660E+2	
2	0.2200E+2	0.2600E+2	0.2800E+2	0.3900E+2	0.6000E+2
	0.6300E+2	0.4700E+2	0.2700E+2	0.1100E+2	
3	0.3000E+1	0.8000E+1	0.1000E+2	0.2000E+2	0.2900E+2
	0.5800E+2	0.3600e+2	0.2300E+2	0.1600E+2	
4	0.1800E+1	0.4000E+1	0.6600E+1	0.1560E+2	0.2390E+2
	0.3010E+2	0.4110E+2	0.4580E+2	0.3540E+2	
5	0.0000E+0	0.3000E+1	0.8000E+1	0.1000E+2	0.2000E+2
	0.2900E+2	0.5800E+2	0.3600E+2	0.2300E+2	
6	0.3560E+2	0.7100E+1	0.6300E+1	0.6800E+1	0.1310E+2
	0.2540E+2	0.5220E+2	0.6350E+2	0.6370E+2	
7	0.8100E+1	0.1010E+2	0.2810E+2	0.4220E+2	0.4750E+2
	0.4310E+2	0.4500E+2	0.3400E+2	0.1450E+2	
8	0.2670E+2	0.2620E+2	0.4180E+2	0.6400E+2	0.7800E+2
	0.8510E+2	0.7300E+2	0.3560E+2	0.7100E+1	
9	0.1100E+2	0.3500E+2	0.4700E+2	0.7300E+2	0.1030E+3
	0.1220E+3	0.7800E+2	0.4000E+2	0.2100E+2	
10	0.7000E+2	0.3400E+2	0.1600E+2	0.5000E+1	0.1100E+2
	0.3500E+2	0.4700E+2	0.7300E+2	0.1030E+3	
11	0.5000E+1	0.2700E+1	0.9500E+1	0.1210E+2	0.2670E+2
	0.2620E+2	0.4180E+2	0.6400E+2	0.7800E+2	
12	0.1061E+3	0.6980E+2	0.3780E+2	0.1140E+2	0.2090E+2
	0.3640E+2	0.4510E+2	0.6120E+2	0.8590E+2	
13	0.8480E+2	0.1259E+3	0.1544E+3	0.9250E+2	0.1980E+2
	0.7000E+1	0.3060E+2	0.3480E+2	0.6650E+2	
14	0.5480E+2	0.2270E+2	0.4300E+1	0.6700E+1	0.2060E+2
	0.3900E+2	0.5410E+2	0.6450E+2	0.6660E+2	
15	0.2700E+1	0.9500E+1	0.1210E+2	0.2670E+2	0.2620E+2
	0.4180E+2	0.6400E+2	0.7800E+2	0.8510E+2	
16	0.9850E+2	0.6150E+2	0.4010E+2	0.1500E+2	0.1070E+2
	0.2420E+2	0.3670E+2	0.6460E+2	0.8570E+2	
17	0.1220E+3	0.7800E+2	0.4000E+2	0.2100E+2	0.1100E+2
	0.2200E+2	0.2600E+2	0.2800E+2	0.3900E+2	
18	0.9250E+2	0.1980E+2	0.7000E+1	0.3060E+2	0.3480E+2
	0.6650E+2	0.8160E+2	0.1008E+3	0.1061E+3	
19	0.2130E+2	0.4100E+2	0.4690E+2	0.6000E+2	0.6660E+2
	0.8990E+2	0.1181E+3	0.1309E+3	0.1320E+3	
20	0.9580E+2	0.9380E+2	0.5480E+2	0.2270E+2	0.4300E+1
	0.6700E+1	0.2060E+2	0.3900E+2	0.5410E+2	
21	0.2280E+2	0.3850E+2	0.6810E+2	0.8480E+2	0.1259E+3
	0.1544E+3	0.9250E+2	0.1980E+2	0.7000E+1	
22	0.1320E+2	0.8500E+1	0.2750E+2	0.4780E+2	0.7090E+2
	0.6700E+2	0.6420E+2	0.4960E+2	0.3630E+2	
23	0.1390E+2	0.1220E+2	0.5000E+1	0.1400E+1	0.0000E+0
	0.2500E+1	0.8100E+1	0.1010E+2	0.2810E+2	
24	0.2410E+2	0.1020E+2	0.2280E+2	0.3850E+2	0.6810E+2
	0.8480E+2	0.1259E+3	0.1544E+3	0.9250E+2	
25	0.1240E+2	0.1130E+2	0.1700E+2	0.4470E+2	0.6620E+2
	0.1016E+3	0.1112E+3	0.1390E+3	0.7400E+2	

Table 3. RBF centres of Example 2

centre number	weight estimate	error reduction ratio
1	0.74432E-2	0.76100E+0
2	-0.38507E-4	0.22344E-1
3	0.50951E-2	0.67346E-1
4	-0.86575E-2	0.45552E-1
5	0.47966E-2	0.10875E-1
6	-0.76933E-2	0.16170E-2
7	-0.91306E-2	0.63600E-2
8	-0.63918E-2	0.11488E-1
9	-0.12441E-2	0.27480E-2
10	-0.81618E-2	0.51441E-2
11	0.15415E-2	0.18138E-2
12	-0.69548E-2	0.16772E-2
13	0.23785E-3	0.45099E-2
14	0.15499E-1	0.16742E-2
15	-0.83772E-3	0.32627E-2
16	0.79692E-2	0.12948E-2
17	-0.78409E-2	0.17721E-2
18	0.27459E-2	0.27495E-2
19	-0.21946E-2	0.15405E-2
20	0.62307E-2	0.18582E-2
21	0.18980E-2	0.94094E-3
22	0.39493E-2	0.75510E-3
23	0.13891E-2	0.75840E-3
24	0.47437E-2	0.45238E-3
25	-0.44277E-2	0.90901E-3
σ_ϵ^2 for fitting set		0.12101E+3
σ_ϵ^2 for test set		0.17614E+3

Table 4. RBF model of Example 2

position		estimate	position		estimate
1	1st hidden node threshold	-0.28658E+1	31	4th hidden node threshold	-0.53484E+0
2	$y(t-1)$/1st hidden node	0.80735E-1	32	$y(t-1)$/4th hidden node	-0.43177E+0
3	$y(t-2)$/1st hidden node	-0.30500E-1	33	$y(t-2)$/4th hidden node	0.15600E+1
4	$y(t-3)$/1st hidden node	-0.77910E-2	34	$y(t-3)$/4th hidden node	-0.70270E+0
5	$y(t-4)$/1st hidden node	-0.71225E-3	35	$y(t-4)$/4th hidden node	0.20755E+1
6	$y(t-5)$/1st hidden node	0.10319E-1	36	$y(t-5)$/4th hidden node	-0.12969E+0
7	$y(t-6)$/1st hidden node	-0.10192E-1	37	$y(t-6)$/4th hidden node	0.14612E+1
8	$y(t-7)$/1st hidden node	0.41369E-2	38	$y(t-7)$/4th hidden node	0.87202E+0
9	$y(t-8)$/1st hidden node	0.97482E-2	39	$y(t-8)$/4th hidden node	-0.31423E+0
10	$y(t-9)$/1st hidden node	-0.50582E-2	40	$y(t-9)$/4th hidden node	-0.90798E+0
11	2nd hidden node threshold	0.24668E+0	41	5th hidden node threshold	-0.57649E+0
12	$y(t-1)$/2nd hidden node	-0.67449E+0	42	$y(t-1)$/5th hidden node	0.91878E+0
13	$y(t-2)$/2nd hidden node	-0.66665E+0	43	$y(t-2)$/5th hidden node	-0.76083E-1
14	$y(t-3)$/2nd hidden node	0.35548E-1	44	$y(t-3)$/5th hidden node	0.12793E+1
15	$y(t-4)$/2nd hidden node	-0.47036E+0	45	$y(t-4)$/5th hidden node	0.20041E+0
16	$y(t-5)$/2nd hidden node	0.50561E+0	46	$y(t-5)$/5th hidden node	-0.96107E-1
17	$y(t-6)$/2nd hidden node	-0.37458E+0	47	$y(t-6)$/5th hidden node	-0.56305E+0
18	$y(t-7)$/2nd hidden node	-0.54623E+0	48	$y(t-7)$/5th hidden node	0.62144E+0
19	$y(t-8)$/2nd hidden node	-0.56262E+0	49	$y(t-8)$/5th hidden node	0.94449E+0
20	$y(t-9)$/2nd hidden node	-0.44019E+0	50	$y(t-9)$/5th hidden node	0.71099E+0
21	3rd hidden node threshold	-0.10178E+2	51	1st hidden node/output node	0.88947E+2
22	$y(t-1)$/3rd hidden node	-0.31298E-1	52	2nd hidden node/output node	-0.95587E+0
23	$y(t-2)$/3rd hidden node	-0.73172E+0	53	3rd hidden node/output node	0.14143E+2
24	$y(t-3)$/3rd hidden node	0.64585E+0	54	4th hidden node/output node	0.66357E+3
25	$y(t-4)$/3rd hidden node	-0.30200E+0	55	5th hidden node/output node	-0.22212E+3
26	$y(t-5)$/3rd hidden node	-0.33024E+0			
27	$y(t-6)$/3rd hidden node	-0.70718E+0		σ_ϵ^2 for fitting set	0.18375E+3
28	$y(t-7)$/3rd hidden node	0.12325E+1		σ_ϵ^2 for test set	0.28777E+3
29	$y(t-8)$/3rd hidden node	-0.37358E+1			
30	$y(t-9)$/3rd hidden node	0.52889E+1			

Table 5. Neural network model of Example 2

selected term		parameter estimate
1	$y(t-1)$	0.11171E+1
2	$y(t-2)$	-0.91963E-1
3	$y(t-9)$	0.38299E+0
4	$y^3(t-1)$	-0.28084E-4
5	$y(t-1)y(t-8)$	0.44321E-2
6	$y(t-2)y(t-5)y(t-8)$	-0.16203E-3
7	$y^2(t-9)$	-0.25043E-2
8	$y(t-1)y(t-5)y(t-8)$	0.95842E-4
9	$y(t-5)y^2(t-7)$	0.11992E-4
10	$y(t-3)$	-0.21815E+0
11	$y(t-2)y(t-3)y(t-4)$	0.95906E-5
σ_ϵ^2 for fitting set		0.12311E+3
σ_ϵ^2 for test set		0.17769E+3

Table 6. Subset polynomial model of Example 2

i	c_{1i}	c_{2i}	c_{3i}	c_{4i}	c_{5i}
	c_{6i}	c_{7i}	c_{8i}	c_{9i}	c_{10i}
1	0.10730E+2	0.48105E+1	0.44015E+1	0.10281E+2	0.49365E+1
	0.50978E+1	0.59454E+1	0.36107E+1	0.46295E+1	-0.16897E+1
2	0.43947E+1	0.58497E+1	0.65888E+1	0.55126E+1	0.63810E+1
	-0.63816E+0	0.48672E+1	0.86241E+0	0.29730E+1	-0.37378E+0
3	0.61937E+1	0.44838E+1	0.75266E+1	0.42324E+1	0.78787E+1
	0.51713E+0	0.47697E+1	0.21983E+1	0.17473E+1	0.27374E+1
4	0.10142E+2	0.49572E+1	0.43386E+1	0.10005E+2	0.97342E+1
	-0.56596E+1	0.25487E+1	0.30751E+1	-0.52544E+0	0.21745E+1
5	0.47521E+1	0.90093E+1	0.49024E+1	0.62287E+1	0.73210E+1
	-0.14426E+0	0.53636E+1	0.23469E+1	0.26821E+1	-0.21996E+1
6	0.84442E+1	0.97073E+1	0.83585E+1	0.60480E+1	0.83650E+1
	-0.58780E+1	0.46791E+1	-0.57376E+1	0.15661E+1	0.32116E+1
7	0.91619E+1	0.10681E+2	0.10248E+2	0.79788E+1	0.59730E+1
	-0.70590E+0	0.55186E+1	-0.29881E+1	0.30784E+1	0.10231E+1
8	0.44671E+1	0.67735E+1	0.77331E+1	0.10118E+2	0.59616E+1
	-0.51030E+1	0.22469E+1	0.58893E+0	0.16169E+1	0.84655E+0
9	0.63035E+1	0.53274E+1	0.10758E+2	0.64871E+1	0.41489E+1
	-0.33397E+1	0.55870E+1	-0.29784E+1	0.49710E+1	-0.55263E+1
10	0.10785E+2	0.10564E+2	0.79311E+1	0.10212E+2	0.92241E+1
	-0.44918E+1	0.23565E+1	0.14982E+1	0.14907E+1	0.52527E+0
11	0.72888E+1	0.10881E+2	0.63223E+1	0.73839E+1	0.88146E+1
	0.12981E+1	-0.28209E+0	0.29325E+1	0.57439E+1	-0.70413E+0
12	0.48986E+1	0.57077E+1	0.10821E+2	0.91361E+1	0.49397E+1
	-0.52644E+1	-0.55796E+0	-0.33763E+1	0.46836E+1	0.38739E+1
13	0.10183E+2	0.84113E+1	0.92295E+1	0.96563E+1	0.71646E+1
	-0.22482E+1	0.34594E+1	0.46223E+1	0.30885E+0	0.33795E+1
14	0.81493E+1	0.82471E+1	0.90886E+1	0.64985E+1	0.85780E+1
	-0.42699E+1	0.50132E+1	0.55254E+1	0.15923E+0	-0.33745E+1
15	0.85782E+1	0.84725E+1	0.47643E+1	0.41189E+1	0.10579E+2
	0.20861E+1	0.56745E+1	-0.51807E+1	0.25977E+1	0.46805E+1

continue

Table 7. RBF centres of Example 3

16	0.57993E+1	0.10341E+2	0.91590E+1	0.96197E+1	0.10304E+2
	0.10156E+1	0.53332E+1	-0.52810E+1	0.73098E+0	-0.12890E+1
17	0.58484E+1	0.65139E+1	0.96002E+1	0.61347E+1	0.81367E+1
	-0.42773E+1	0.26217E+1	0.11905E+1	0.59480E+0	0.70246E+0
18	0.86802E+1	0.79164E+1	0.93173E+1	0.91086E+1	0.83180E+1
	-0.16962E+1	-0.23639E+1	0.27592E+1	-0.28571E+1	-0.45013E+1
19	0.48705E+1	0.64785E+1	0.46762E+1	0.61347E+1	0.44335E+1
	0.20232E+1	0.13841E+1	0.60182E-1	0.68656E+0	-0.29954E+1
20	0.10907E+2	0.10629E+2	0.73837E+1	0.69014E+1	0.77045E+1
	-0.42299E+1	-0.15428E+1	-0.71127E+0	-0.36490E+1	0.21935E+1
21	0.74207E+1	0.99884E+1	0.47531E+1	0.10821E+2	0.92408E+1
	-0.15020E+1	-0.32949E+1	0.47079E+1	-0.28941E+1	-0.17538E+1
22	0.94784E+1	0.41383E+1	0.10814E+2	0.43518E+1	0.43961E+1
	-0.94079E-1	0.48546E+1	-0.35511E+1	0.39188E+1	0.28930E+0
23	0.87107E+1	0.10291E+2	0.65657E+1	0.75886E+1	0.56123E+1
	0.12829E+1	0.52081E+1	-0.53658E+1	0.49687E+0	-0.37771E+1
24	0.68011E+1	0.77665E+1	0.10283E+2	0.54849E+1	0.99005E+1
	0.46377E+1	0.16682E-1	-0.10170E+1	0.35971E+1	0.34109E+1
25	0.79544E+1	0.10342E+2	0.96978E+1	0.46820E+1	0.40850E+1
	-0.52976E+1	-0.50760E+1	-0.13009E+1	-0.40418E+1	-0.33909E+1
26	0.65234E+1	0.10805E+2	0.10404E+2	0.92026E+1	0.47703E+1
	-0.97348E+0	-0.34912E-1	0.30863E+1	0.45714E+0	0.28937E+1
27	0.68422E+1	0.56610E+1	0.51511E+1	0.94633E+1	0.77392E+1
	-0.44312E+1	0.19245E+0	0.57948E+1	-0.44061E+1	0.14927E+1

Table 7. RBF centres of Example 3

centre number	weight estimate	error reduction ratio
constant	0.15643E+2	0.98539E+0
1	-0.19479E-1	0.10458E-1
2	0.28817E+0	0.19889E-2
3	-0.94923E-2	0.10727E-3
4	-0.47691E-1	0.49406E-4
5	-0.12648E+0	0.34403E-4
6	-0.18401E-1	0.59714E-4
7	0.45193E-4	0.14976E-4
8	-0.85851E-2	0.12031E-3
9	0.22686E-1	0.25194E-3
10	0.50058E-1	0.75817E-4
11	0.50162E-2	0.80147E-4
12	0.21515E-1	0.25299E-4
13	0.10867E-1	0.84137E-4
14	0.21019E-1	0.10289E-3
15	-0.10753E-1	0.13684E-4
16	-0.30959E-1	0.36280E-4
17	-0.13874E+0	0.15841E-4
18	0.33549E-1	0.65978E-4
19	-0.57112E-1	0.98660E-4
20	0.71061E-1	0.35300E-4
21	-0.40188E-1	0.63064E-4
22	-0.85792E-1	0.61189E-4
23	0.55479E-1	0.14886E-4
24	0.35190E-1	0.24787E-4
25	-0.24433E-1	0.11069E-4
26	-0.41399E-1	0.97732E-5
27	0.46347E-1	0.18176E-4
σ_ϵ^2 for fitting set		0.52555E-1
σ_ϵ^2 for test set		0.59605E-1

Table 8. RBF model of Example 3

position		parameter estimate
1	threshold of 1st hidden node	0.41542E+1
2	$y(t-1)$ to 1st hidden node	-0.86652E+0
3	$y(t-2)$ to 1st hidden node	0.65748E+0
4	$y(t-3)$ to 1st hidden node	-0.92485E+0
5	$y(t-4)$ to 1st hidden node	-0.14368E+1
6	$y(t-5)$ to 1st hidden node	-0.13881E+1
7	$u(t-1)$ to 1st hidden node	0.95717E-1
8	$u(t-2)$ to 1st hidden node	-0.24441E+1
9	$u(t-3)$ to 1st hidden node	-0.32247E+1
10	$u(t-4)$ to 1st hidden node	0.10609E+1
11	$u(t-5)$ to 1st hidden node	0.28496E+0
12	threshold of 2nd hidden node	-0.19307E+1
13	$y(t-1)$ to 2nd hidden node	0.28604E+0
14	$y(t-2)$ to 2nd hidden node	-0.21059E-2
15	$y(t-3)$ to 2nd hidden node	0.10688E+0
16	$y(t-4)$ to 2nd hidden node	-0.50079E-1
17	$y(t-5)$ to 2nd hidden node	-0.50801E-3
18	$u(t-1)$ to 2nd hidden node	0.14850E+0
19	$u(t-2)$ to 2nd hidden node	0.16681E-1
20	$u(t-3)$ to 2nd hidden node	0.20725E-2
21	$u(t-4)$ to 2nd hidden node	-0.49558E-2
22	$u(t-5)$ to 2nd hidden node	-0.26427E-1
23	threshold of 3rd hidden node	-0.15231E+1
24	$y(t-1)$ to 3rd hidden node	-0.30419E+0
25	$y(t-2)$ to 3rd hidden node	0.98374E+0
26	$y(t-3)$ to 3rd hidden node	-0.88712E+0
27	$y(t-4)$ to 3rd hidden node	0.12615E+1
28	$y(t-5)$ to 3rd hidden node	0.10536E+1
29	$u(t-1)$ to 3rd hidden node	-0.30323E+0
30	$u(t-2)$ to 3rd hidden node	0.63262E+0
31	$u(t-3)$ to 3rd hidden node	-0.47314E+0
32	$u(t-4)$ to 3rd hidden node	-0.16157E-1
33	$u(t-5)$ to 3rd hidden node	-0.45640E-1
34	1st hidden node to output node	0.89170E+1
35	2nd hidden node to output node	0.10030E+2
36	3rd hidden node to output node	0.14756E+1
	σ_ϵ^2 for fitting set	0.54283E-1
	σ_ϵ^2 for test set	0.52497E-1

Table 9. Neural network model of Example 3

Fig.1. Schematic of RBF model

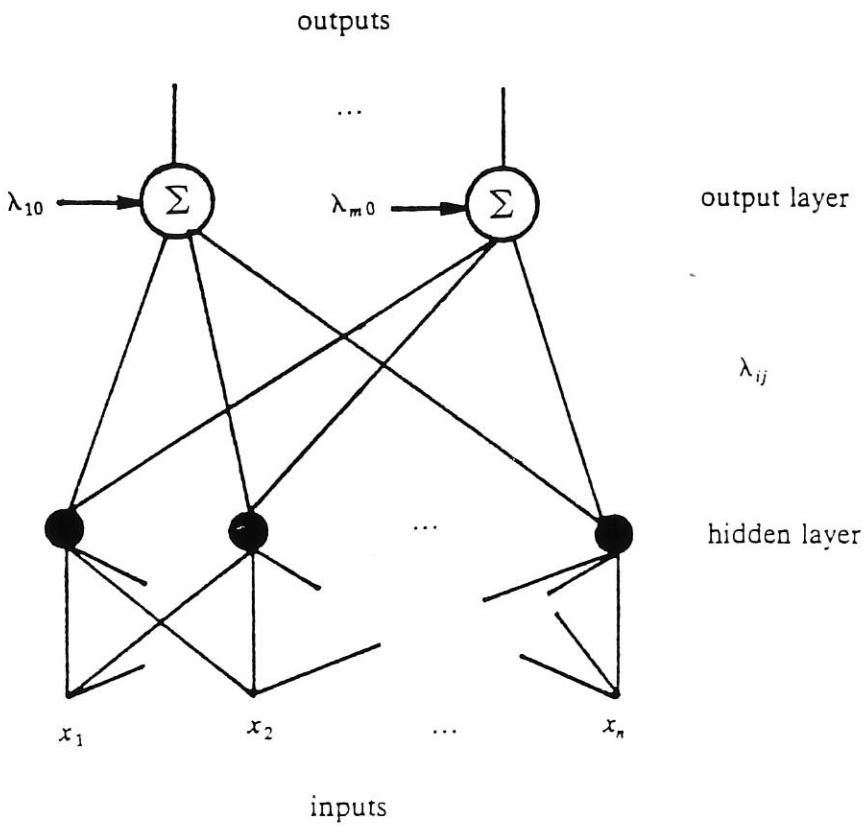

Fig.2. Model validity tests for RBF model (Example 1)

(a) $\Phi_{\varepsilon\varepsilon}(k)$; (b) $\Phi_{\varepsilon(\varepsilon u)}(k)$; (c) $\Phi_{uu}(k)$; (d) $\Phi_{u^2\varepsilon}(k)$; (e) $\Phi_{u^2\varepsilon^2}(k)$. Dashed line: 95% confidence interval.

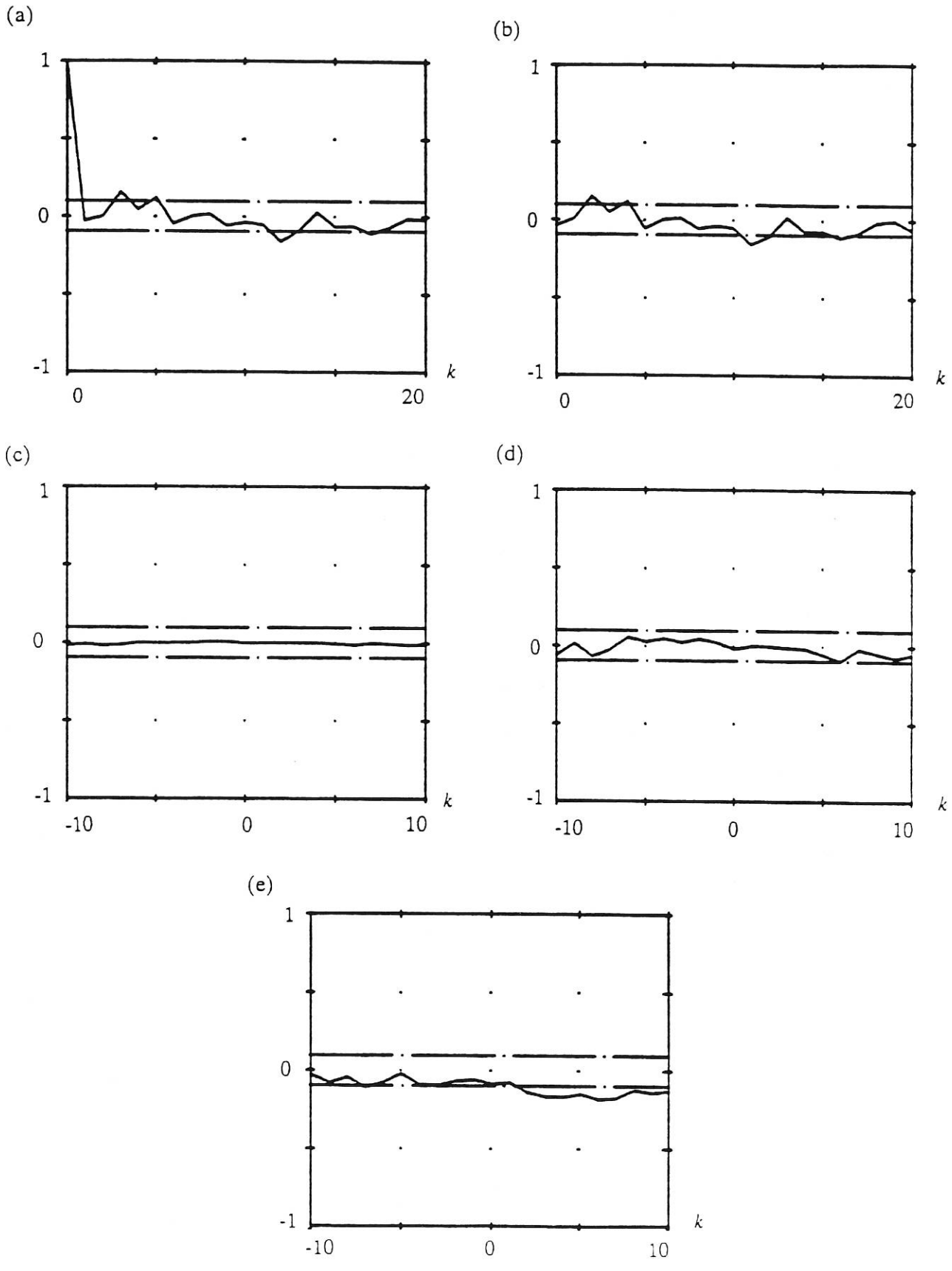

Fig.3. Data set and RBF model response (Example 1)

(a) $u(t)$; (b) $y(t)$; (c) $\hat{y}(t)$;

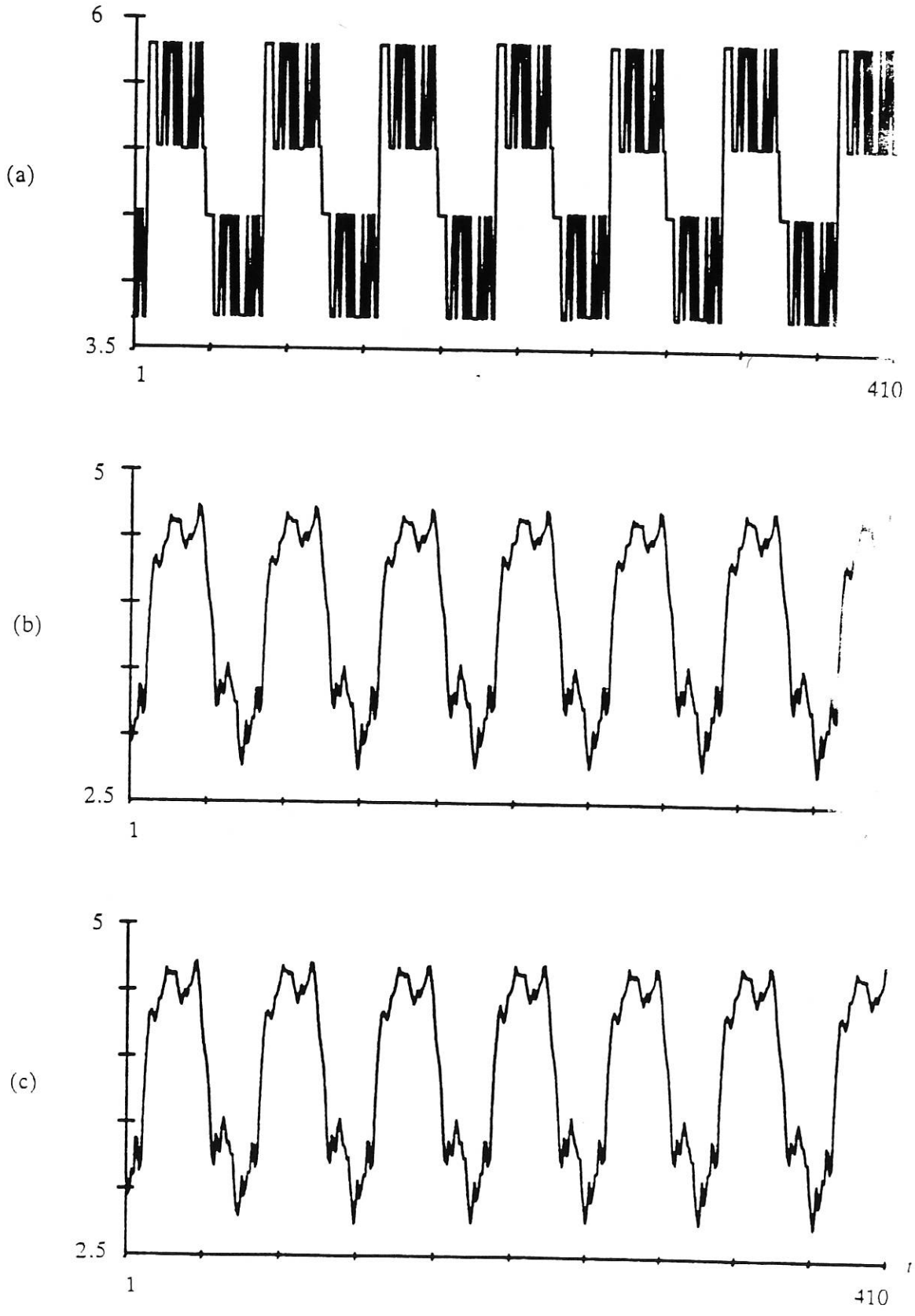

Fig.3. Data set and RBF model response (Example 1)

(d) $\epsilon(t)$; (e) $\epsilon_d(t)$; (f) $\hat{y}_d(t)$.

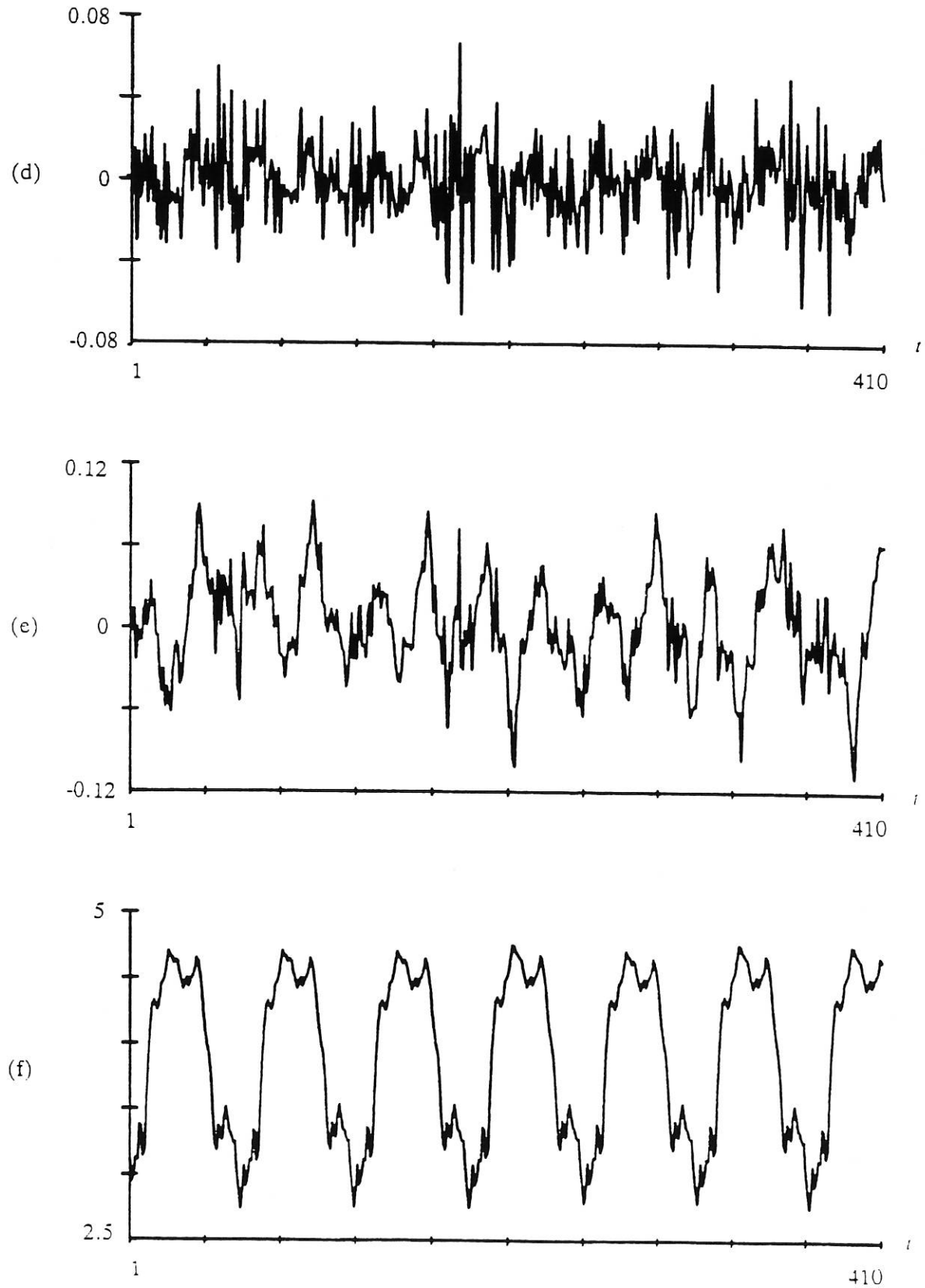

Fig.4. Model validity tests for neural network model (Example 1)

(a) $\Phi_{\varepsilon\varepsilon}(k)$; (b) $\Phi_{\varepsilon(\varepsilon u)}(k)$; (c) $\Phi_{ue}(k)$; (d) $\Phi_{u^2\varepsilon}(k)$; (e) $\Phi_{u^2\varepsilon^2}(k)$. Dashed line: 95% confidence interval.

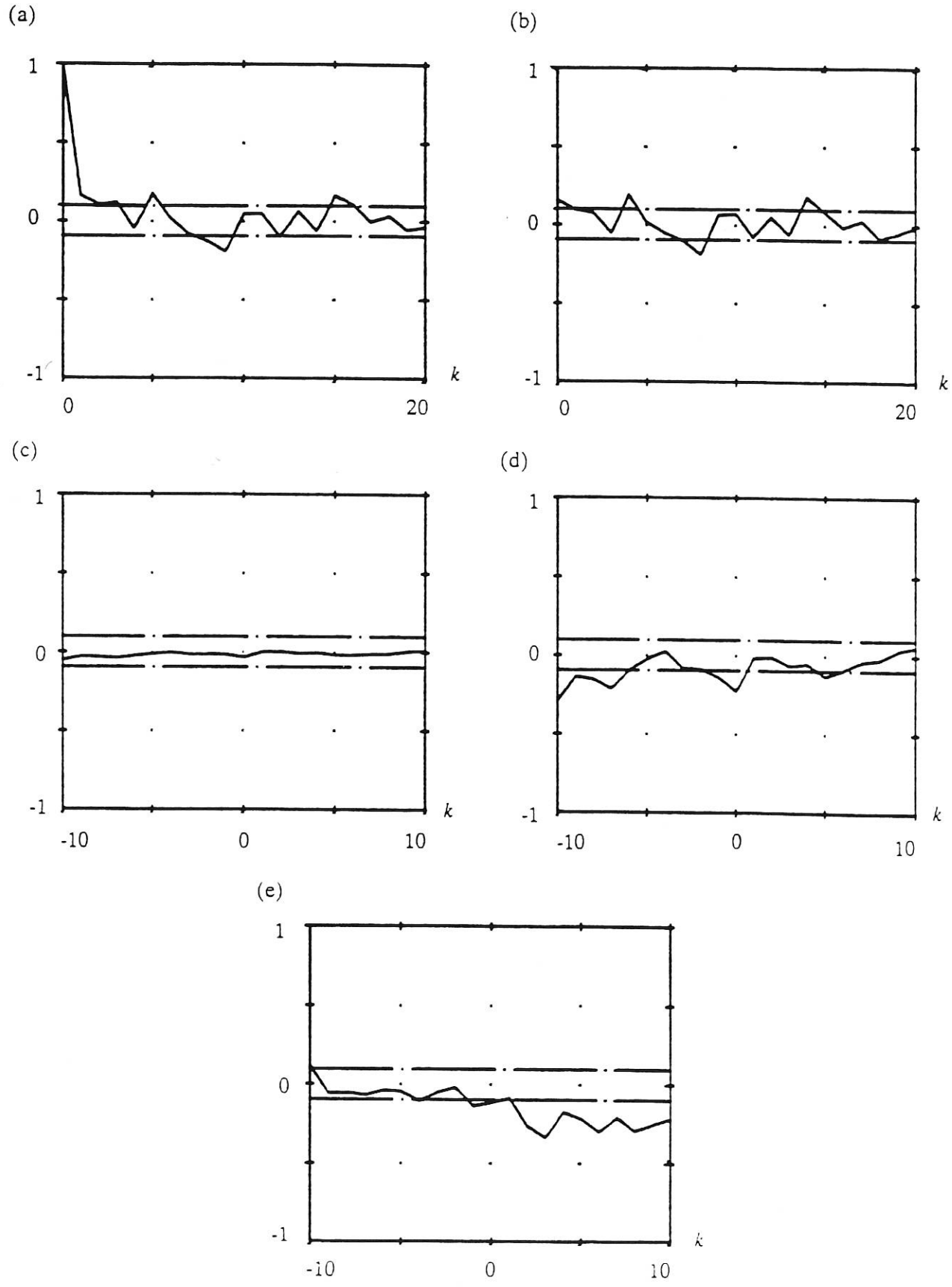

Fig.5. Neural network model response (Example 1)

(a) $\hat{y}(t)$; (b) $\epsilon(t)$;

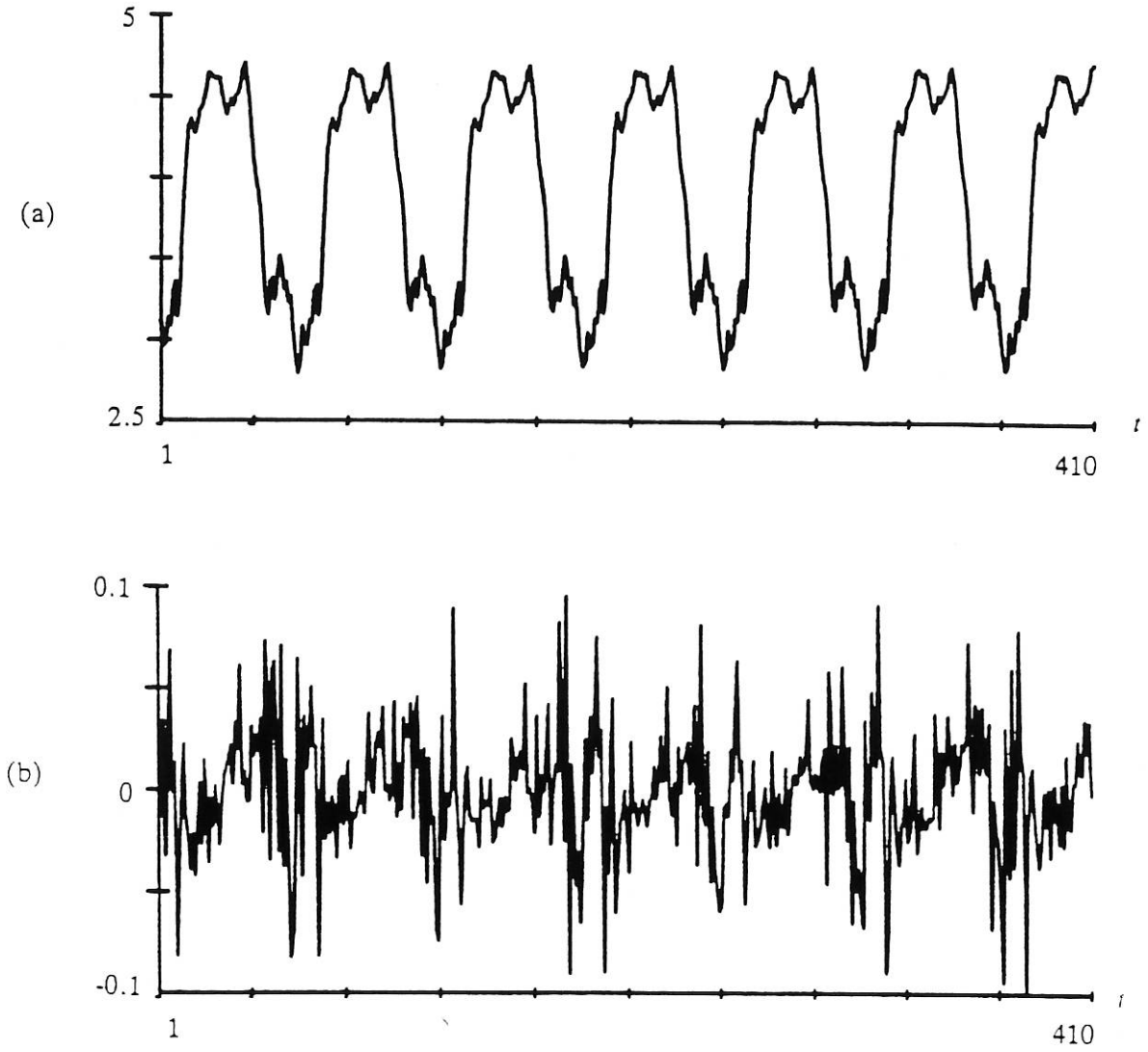

Fig.5. Neural network model response (Example 1)

(c) $\epsilon_d(t)$; (d) $\hat{y}_d(t)$.

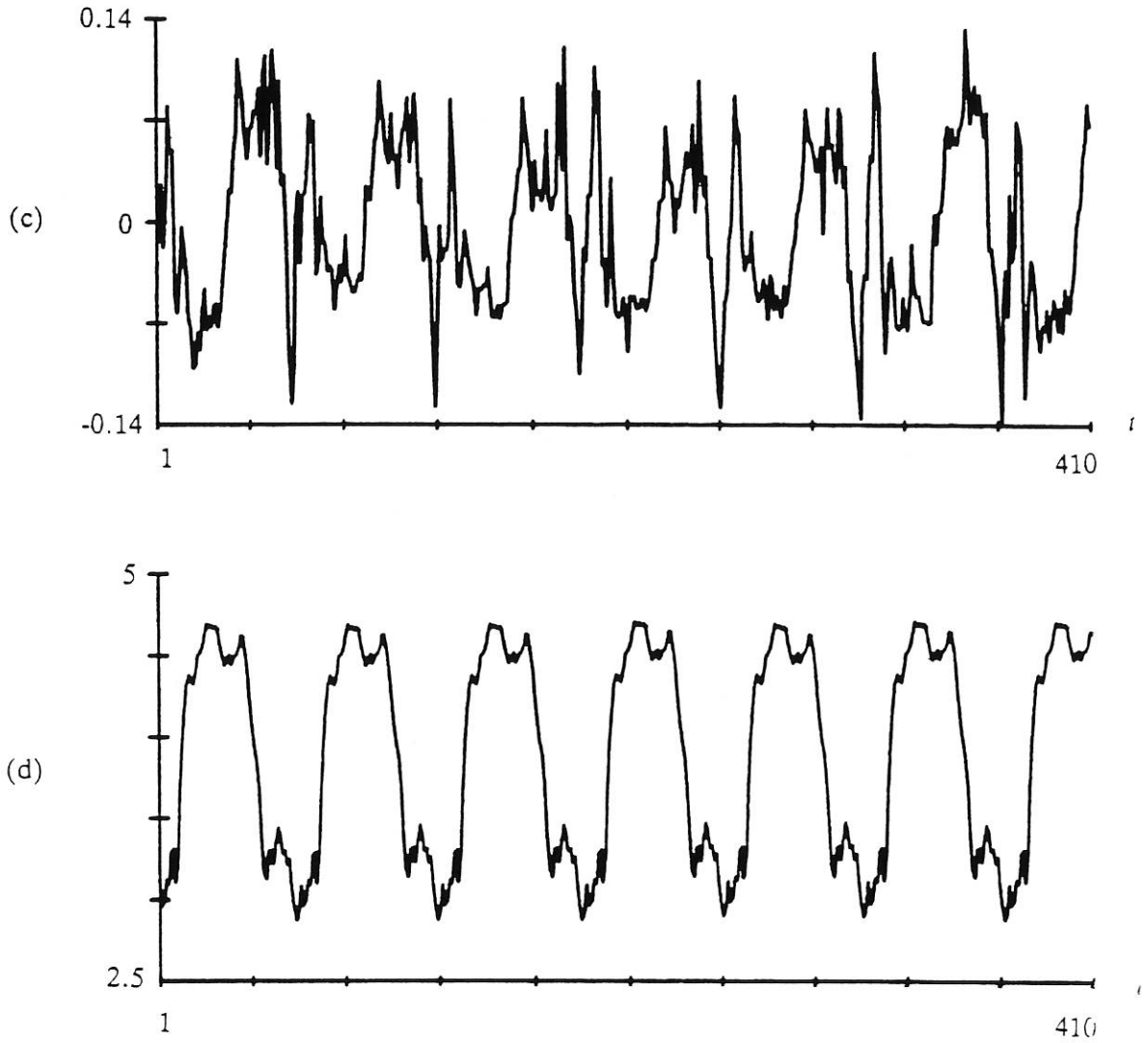

Fig.6. Observations and RBF model response (Example 2)

(a) observations; (b) one-step-ahead predictions; (c) unforced response, first 9 observations used as initial condition.

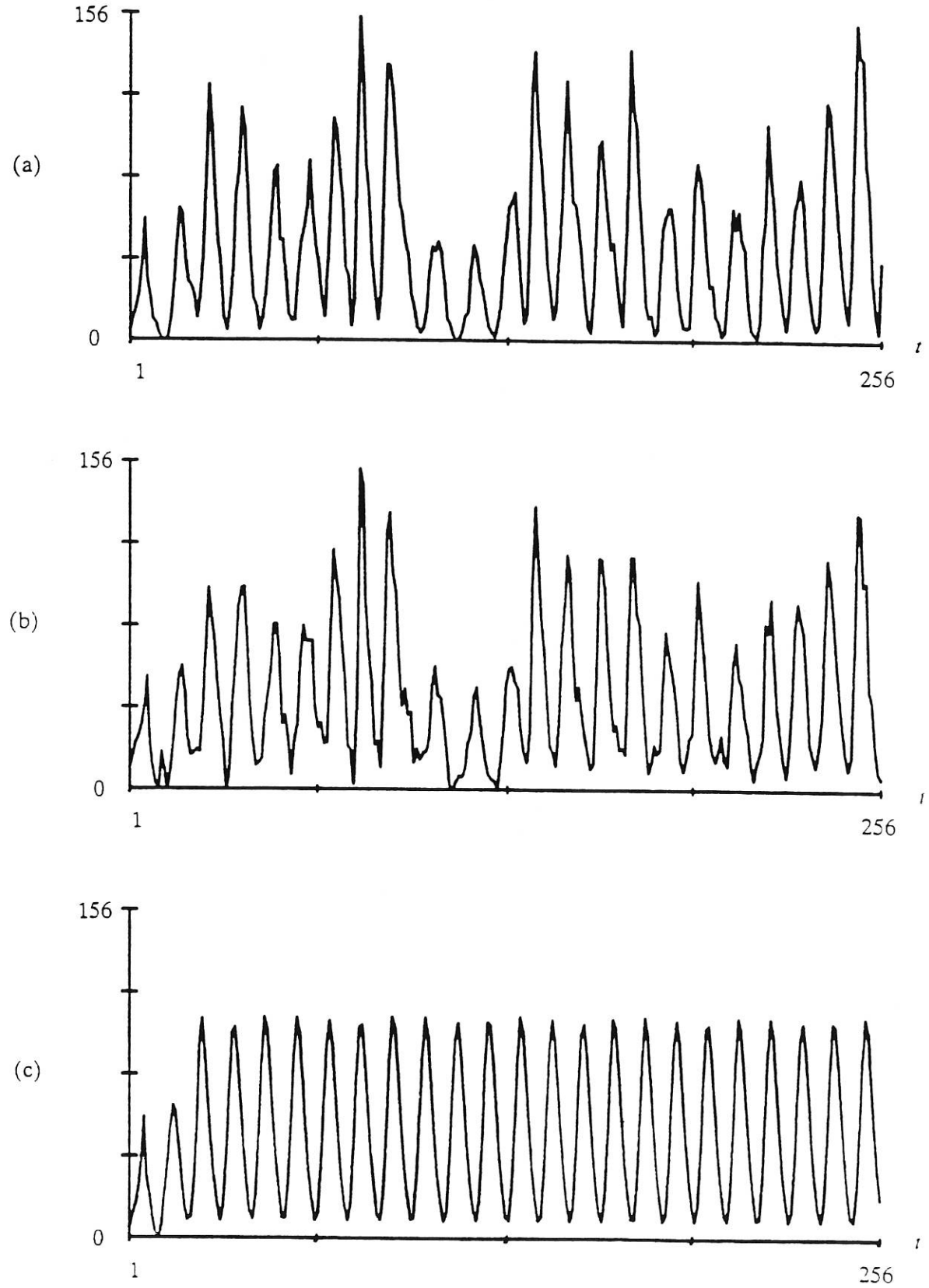

Fig.7. Identification data set (Example 3)

(a) $u(t)$; (b) $y(t)$.

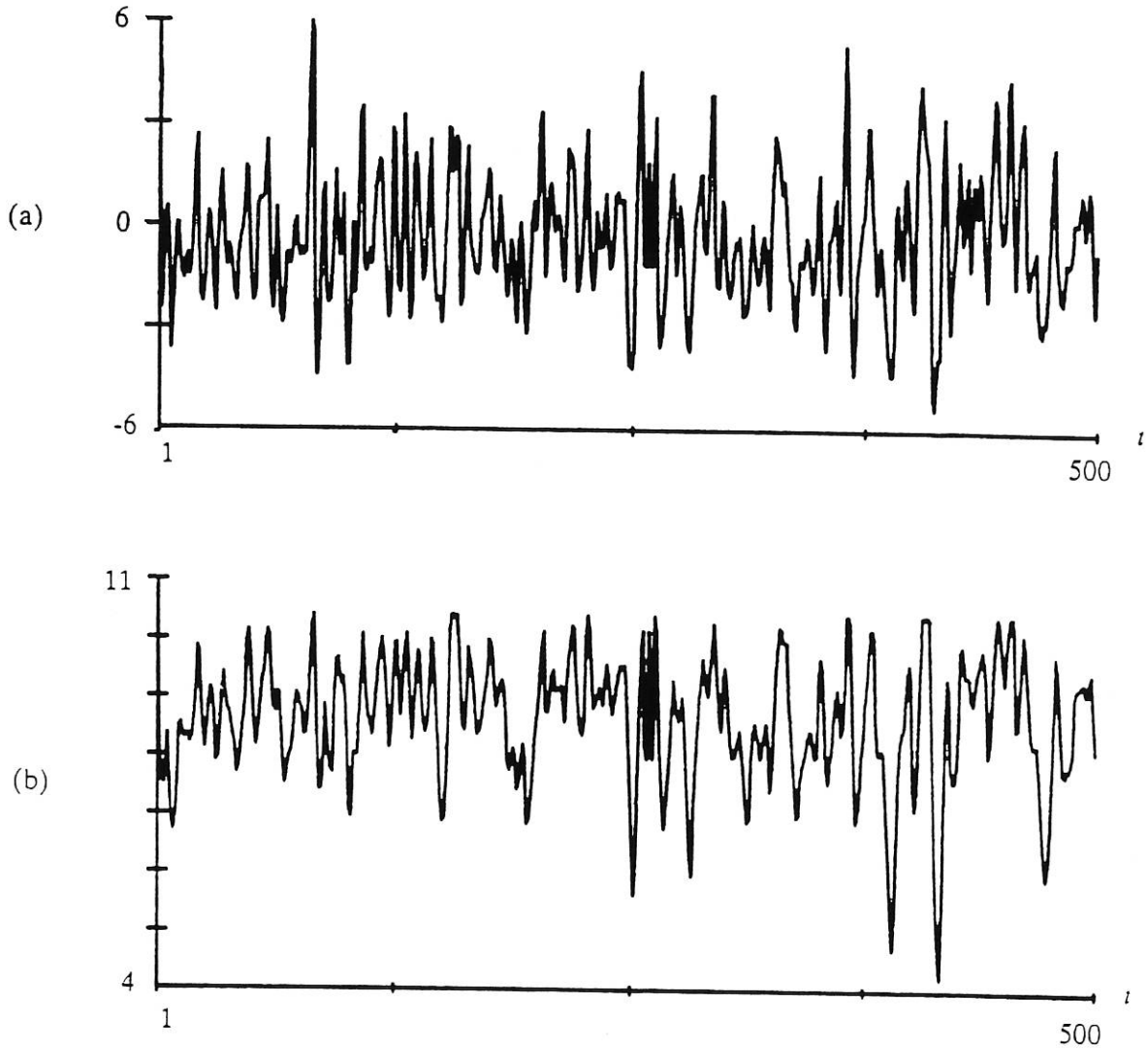

Fig.8. Model validity tests using identification set (Example 3)

(a) $\Phi_{\varepsilon\varepsilon}(k)$; (b) $\Phi_{\varepsilon(\varepsilon u)}(k)$; (c) $\Phi_{u\varepsilon}(k)$; (d) $\Phi_{u^2\varepsilon}(k)$; (e) $\Phi_{u^2\varepsilon^2}(k)$. Dashed line: 95% confidence interval.

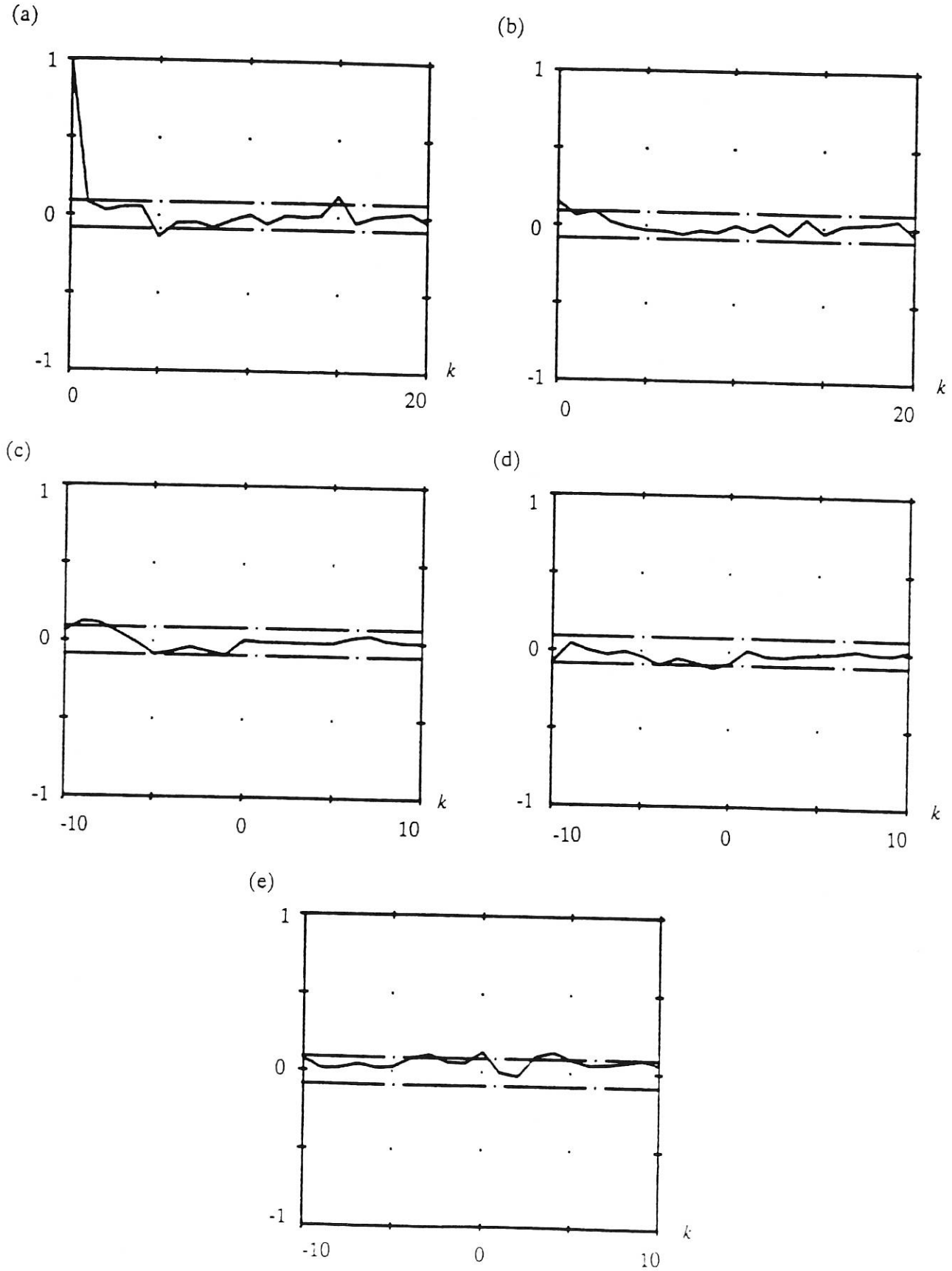

Fig.9. Model validity tests using test set (Example 3)

(a) $\Phi_{\varepsilon\varepsilon}(k)$; (b) $\Phi_{\varepsilon(\varepsilon u)}(k)$; (c) $\Phi_{u\varepsilon}(k)$; (d) $\Phi_{u^2\varepsilon}(k)$; (e) $\Phi_{u^2\varepsilon^2}(k)$. Dashed line: 95% confidence interval.

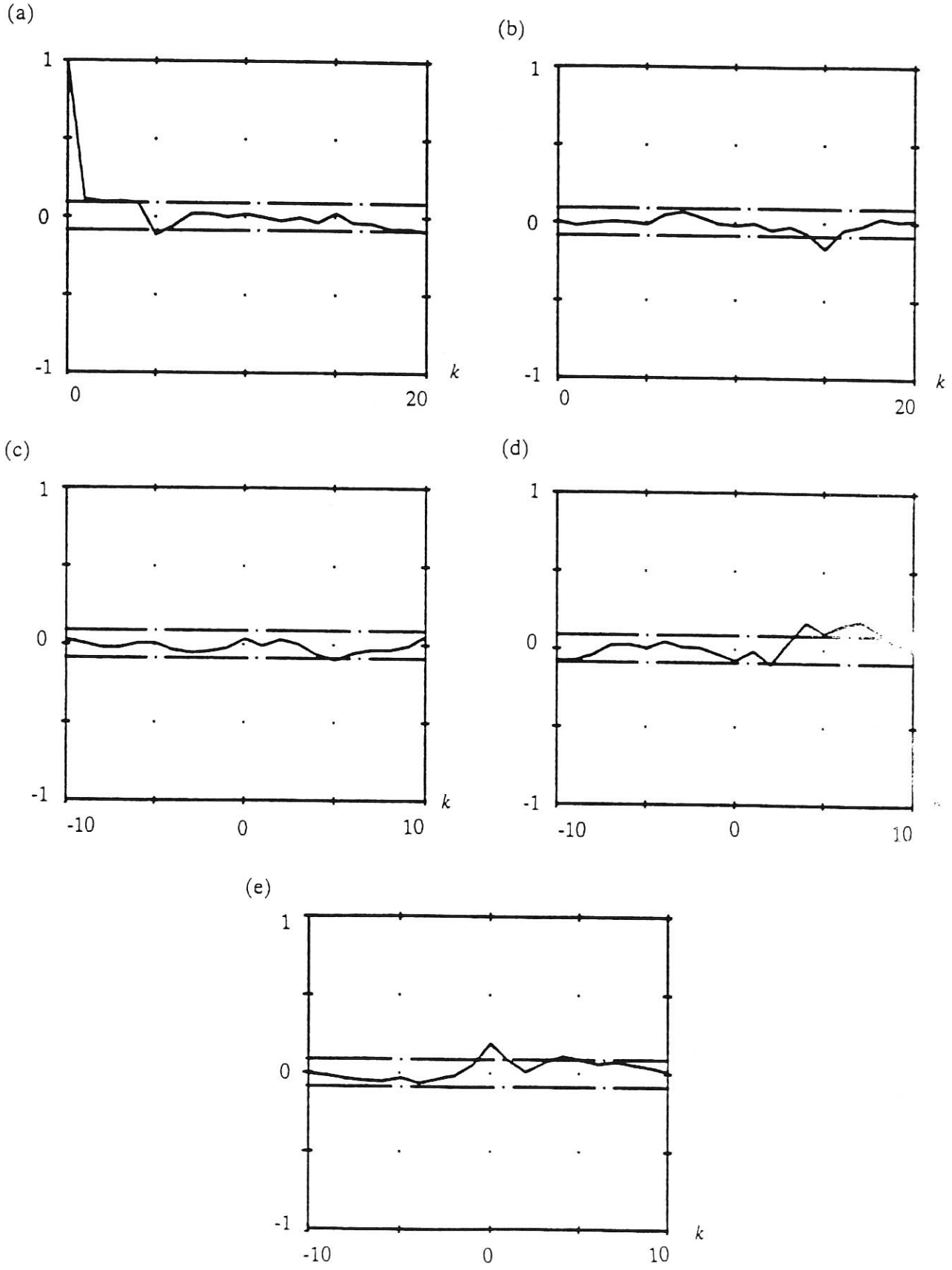

Fig.10. Test set and model response (Example 3)

(a) $u(t)$; (b) $y(t)$; (c) $\hat{y}(t)$;

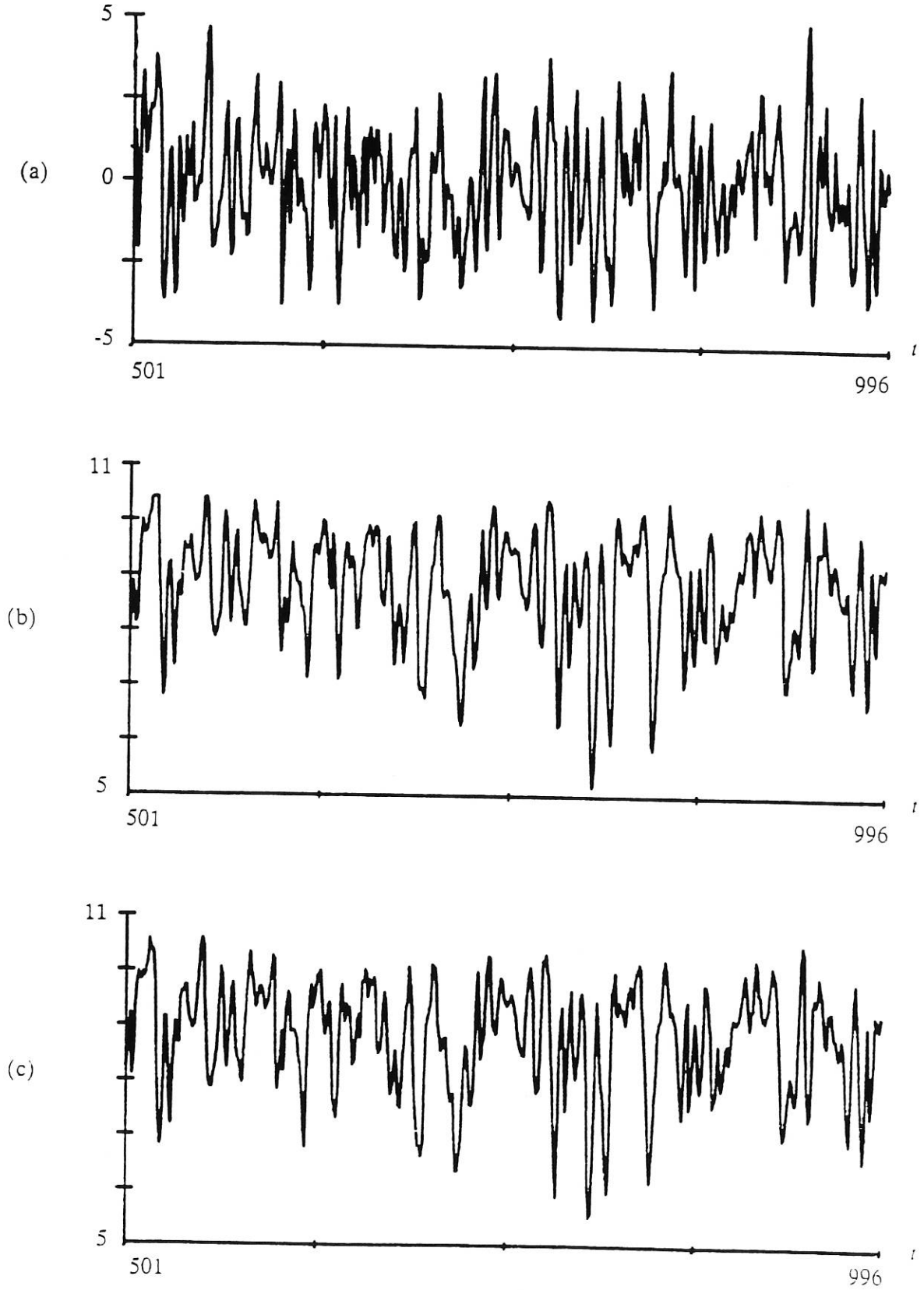

Fig.10. Test set and model response (Example 3)

(d) $\epsilon(t)$; (e) $\epsilon_d(t)$; (f) $\hat{y}_d(t)$.

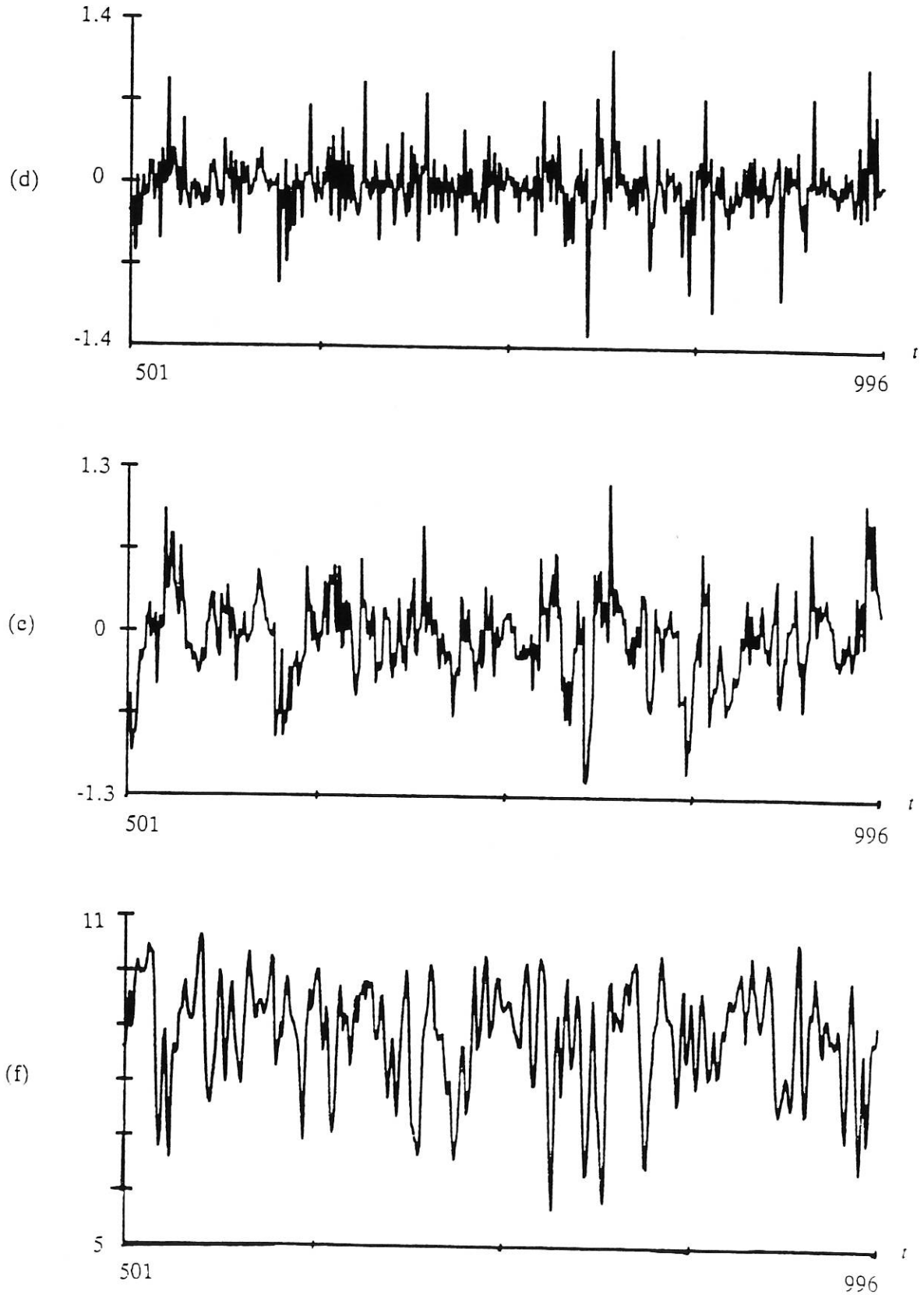

Fig.1. Schematic of RBF model

Fig.2. Model validity tests for RBF model (Example 1)

(a) $\Phi_{\epsilon\epsilon}(k)$; (b) $\Phi_{\epsilon(\epsilon u)}(k)$; (c) $\Phi_{u\epsilon}(k)$; (d) $\Phi_{u^2\epsilon}(k)$; (e) $\Phi_{u^2\epsilon^2}(k)$. Dashed line: 95% confidence interval.

Fig.3. Data set and RBF model response (Example 1)

(a) $u(t)$; (b) $y(t)$; (c) $\hat{y}(t)$;

(d) $\epsilon(t)$; (e) $\epsilon_d(t)$; (f) $\hat{y}_d(t)$.

Fig.4. Model validity tests for neural network model (Example 1)

(a) $\Phi_{\epsilon\epsilon}(k)$; (b) $\Phi_{\epsilon(\epsilon u)}(k)$; (c) $\Phi_{u\epsilon}(k)$; (d) $\Phi_{u^2\epsilon}(k)$; (e) $\Phi_{u^2\epsilon^2}(k)$. Dashed line: 95% confidence interval.

Fig.5. Neural network model response (Example 1)

(a) $\hat{y}(t)$; (b) $\epsilon(t)$;

(c) $\epsilon_d(t)$; (d) $\hat{y}_d(t)$.

Fig.6. Observations and RBF model response (Example 2)

(a) observations; (b) one-step-ahead predictions; (c) unforced response, first 9 observations used as initial condition.

Fig.7. Identification data set (Example 3)

(a) $u(t)$; (b) $y(t)$.

Fig.8. Model validity tests using identification set (Example 3)

(a) $\Phi_{\epsilon\epsilon}(k)$; (b) $\Phi_{\epsilon(\epsilon u)}(k)$; (c) $\Phi_{u\epsilon}(k)$; (d) $\Phi_{u^2\epsilon}(k)$; (e) $\Phi_{u^2\epsilon^2}(k)$. Dashed line: 95% confidence interval.

Fig.9. Model validity tests using test set (Example 3)

(a) $\Phi_{\epsilon\epsilon}(k)$; (b) $\Phi_{\epsilon(\epsilon u)}(k)$; (c) $\Phi_{u\epsilon}(k)$; (d) $\Phi_{u^2\epsilon}(k)$; (e) $\Phi_{u^2\epsilon^2}(k)$. Dashed line: 95% confidence interval.

Fig.10. Test set and model response (Example 3)

(a) $u(t)$; (b) $y(t)$; (c) $\hat{y}(t)$;

(d) $\epsilon(t)$; (e) $\epsilon_d(t)$; (f) $\hat{y}_d(t)$.