Review of rational (total) nonlinear dynamic system modelling, identification and control

Quanmin Zhu*
Department of Engineering Design and Mathematics, University of the West of England, Frenchay Campus, Coldharbour Lane, Bristol, BS16 1QY, UK
quan.zhu@uwe.ac.uk
*Corresponding Author

Yongji Wang
School of Automation, Huazhong University of Science and Technology Wuhan, China, 430074
wangyjch@mail.hust.edu.cn

Dongya Zhao
College of Chemical Engineering, China University of Petroleum, Qingdao, China, 266580
dyzhao@upc.edu.cn

Shaoyuan Li
Department of Automation, Shanghai Jiao Tong University, Shanghai, China, 200240
syli@sjtu.edu.cn

Stephen A Billings
Department of Automatic Control and Systems Engineering, University of Sheffield
Mappin Street, Sheffield, S1 3JD
s.billings@sheffield.ac.uk

Abstract
This paper is a summary of the research development in the rational (total) nonlinear dynamic modelling over the last two decades. Total nonlinear dynamic systems are defined as those where the model parameters and input (controller outputs) are subject to nonlinear to the output. Previously this class of models has been known as rational models, which is a model that can be considered to belong to the Non-linear AutoRegressive Moving Average with eXogenous input (NARMAX) model subset and is an extension of the well-known polynomial NARMAX model. The justification for using the rational model is that it provides a very concise and parsimonious representation for highly complex non-linear dynamic systems and has excellent interpolatory and extrapolatory properties. However model identification and controller design are much more challenging compared to the polynomial models. This has been a new and fascinating research trend in the area of mathematical modelling, control and applications, but still within a limited research community. This paper brings several representative algorithms together, developed by the authors and their colleagues, to form an easily referenced archive for promotion of the awareness, tutorial, applications, and even further research expansion.

Key words: rational (total) nonlinear dynamic systems, nonlinear rational model set, identification, validation, U-model and U-control

1 Introduction

The Non-linear AutoRegressive Moving Average with eXogenous input (NARMAX) model set has been extensively studied in theory (Sontag 1978, Chen and Billings 1989, Haber and Unbehauen 1990) and gradually adopted in applications (Proll and Karim 1994, Wang 1994). There are two main streams of sub-model sets, polynomial and rational models. A polynomial NARMAX model is defined as linear in the parameters and non-linear in the regression terms, and can be used to represent a wide range of linear and non-linear systems. The advantage for identification of polynomial models comes from the fact that the model is linear in the parameters. The rational model (Sontag 1979, Billings and Chen 1989, Billings and Zhu 1991) represents an extension of the polynomial model set, and is defined as a ratio of two polynomial expressions. Therefore rational model is non-linear in both the parameters and the regression terms. This is induced by the denominator polynomial. Accordingly the identification and control of rational models are more challenging based on the total nonlinear structures.

Rational models have been gradually adopted in various applications of non-linear system modelling and control (Ford, Titterington, and Kitsos 1989, Ponton 1993, Kambhampati, Mason, and Warwick 2000), particularly the importance of modelling of chemical kinetics has increased sharply as a consequence of the applicability of modelling of catalytic reactions (Dimitrov and Kamenski 1991, Kamenski and Dimitrov1993). Rational models are not only alternative expressions in approximating a wide range of data sets in chemical engineering, but also are a class of mechanistic models, which most previous experience or theoretical considerations had not put forward (Dimitrov and Kamenski 1991).

A number of methodologies and algorithms have been developed for rational model structure detection, parameter estimation, and correlation based validity tests. These include a prediction error estimator (Billings and Chen 1989), generalised least squares estimators (Billings and Zhu 1991, Zhu and Billings 1991, 1993), recursive least squares parameter estimator (Zhu and Billings 1993), orthogonal model structure detection and parameter estimation algorithm (Billings and Zhu 1994a), genetic model structure detection algorithm (Billings and Mao 1998), back propagation parameter estimator (2003), implicit least squares parameter estimator (Zhu 2005), correlation based model validity tests (Billings and Zhu 1994b, 1995, Zhang, Ashley, and Zhu 2007, Zhu, Zhang, and Ashley 2007).

It should be noted that so far it is still an open research question on how to design rational model based control systems analytically. It is hoped that the U-model framework explained in this paper will provide a concise analytical solution in future studies.

The rest of the paper is organised into the following sections. In section 2, rational model is defined to provide a framework for the following algorithm development. Subsequently an extended least squares parameter estimator is presented and a full step by step implementation in computation is listed as a guiding procedure for user's applications. In section 3, a back propagation algorithm is tailored for rational model parameter estimation together with a step by step implementation of the algorithm. In section 4, orthogonal computation is introduced to provide a solution for rational model structure detection (term selection) and parameter estimation. In section 5, the latest development of correlation based model validation procedure is presented for

validating identified linear and nonlinear models including neural networks and fuzzy logic models. In section 6, both U-model and U-control are explained conceptually, in terms of the U-transform, for control system design. In section 7, a brief conclusion is drawn to summarise the results.

2 Extended least squares (ELS) algorithm for parameter estimation (Billings and Zhu 1991)

2.1 Rational (total) model

Rational or total nonlinear dynamic systems can be mathematically described with a ratio of two polynomials, and are sometimes commonly known as rational models

$$y(t) = \hat{y}(t) + e(t) = \frac{a(t)}{b(t)} + e(t) = \frac{a\left[u(t-1),...u(t-n_{nu}), y(t-1),...y(t-n_{ny}), e(t-1),...e(t-n_{ne})\right]}{b\left[u(t-1),...u(t-n_{du}), y(t-1),...y(t-n_{dy}), e(t-1),...e(t-n_{de})\right]} + e(t) \tag{2.1}$$

where $y(t)$ and $\hat{y}(t)$ are the measured and one step ahead predicted model outputs respectively, $u(t)$ is the input, $e(t)$ is the model error, and t (= 1, 2, …) is the time index. Generally the numerator $a(t)$ and denominator $b(t)$ are functions of past inputs, outputs, and errors, and can be expressed in terms of polynomials.

$$\begin{aligned} a(t) &= \sum_{j=1}^{num} p_{nj}(t)\theta_{nj} \\ b(t) &= \sum_{j=1}^{den} p_{dj}(t)\theta_{dj} \end{aligned} \tag{2.2}$$

The regression terms $p_{nj}(t)$ and $p_{dj}(t)$ are the products of past inputs, outputs, and errors, such as $u(t-1)y(t-3)$, $u(t-1)e(t-2)$, $y^2(t-1)$, and θ_{nj}, θ_{dj} are the associated parameters. The task of model structure detection is to select candidate regression terms from a pre-assigned large candidate base. The task of parameter estimation, for a given or detected model structure, is to estimate the associated parameters from the measured inputs and outputs. Model validation is the final step to diagnose the feasibility of an identified model if it is a real representative of the underlying system. The correlation based model validation uses data sequences of input, output, and residual (difference between measured system output and model predicted output) to form a series of correlation functions to determine if the residual has been reduced to an uncorrelated sequence with zero mean and finite variance.

Several remarks relating to the characteristics of the rational model of (2.1) are noted below:

1) The rational model includes almost all the other **smooth** linear and nonlinear models as its subsets. For example the polynomial NARMAX model is a special case of rational model (2.1) by setting denominator polynomial $b(t) = 1$. Many neuro-fuzzy systems have been expressed as non-linear rational models (Wang 1994). For example fuzzy systems with a centre defuzzifier, product inference rule, singleton fuzzifier, and Gaussian membership function. The normalised radial basis function network is

also a type of rational model. When the centres and widths are estimated this can become a rational model parameter estimation problem.

2) The rational model can be much more concise than a polynomial expansion, for example

$$y(t) = \frac{u(t-1)}{1+y^2(t-1)} = u(t-1)\left[1 - y^2(t-1) + y^4(t-1)\cdots\right] \qquad (2.3)$$

3) The rational model can be frequently used to represent complicated system structures with a fairly low degree in both the numerator and denominator polynomials. For example, large and quick deviations in a system output, can be expressed as

$$y(t) = \frac{1}{1+u(t-1)} \qquad (2.4)$$

When $u(t\text{-}1)$ approaches –1, the model output response will be quickly increased. The power to capture quick and large changes is gained by introducing the denominator polynomial.

2.1 Bias analysis using the ordinary LS algorithm

Directly estimating parameters of model (2.1) is difficult in formulation and time consuming in computation even though a prediction error algorithm has been developed (Billing and Chen 1989). This is because the model is nonlinear in the parameters. An alternative approach is to multiply the denominator polynomial out (2.1) to yield a model expressed as linear in the parameters, then least squares algorithms can be applied with proper reformulation. To this end, multiplying the denominator $b(t)$ on both sides of (2.1) and then moving all the terms except the first term $y(t)p_{d1}(t)\theta_{d1}$ on left hand side to the right-hand side gives

$$\begin{aligned} Y(t) &= a(t) - y(t)\sum_{j=2}^{den} p_{dj}(t)\theta_{dj} + b(t)e(t) \\ &= \sum_{j=1}^{num} p_{nj}(t)\theta_{nj} - y(t)\sum_{j=2}^{den} p_{dj}(t)\theta_{dj} + \zeta(t) \\ &= \phi(t)\Theta + \zeta(t) \end{aligned} \qquad (2.5)$$

where

$$\begin{aligned} &Y(t) = y(t)p_{d1}(t)\theta_{d1} \quad \theta_{d1} = 1 \,(\text{without loosing genrality}) \\ &\varphi(t) = \left[\varphi_n(t) \quad \varphi_d(t)\right] = \left[p_{n1}(t) \quad \cdots \quad p_{nnum}(t) \quad -p_{d2}(t)y(t) \quad \cdots \quad -p_{dden}(t)y(t)\right] \\ &\Theta^T = \left[\theta_n \quad \theta_d\right] = \left[\theta_{n1} \quad \cdots \quad \theta_{nnum} \quad \theta_{d2} \quad \cdots \quad \theta_{dden}\right] \\ &\zeta(t) = b(t)e(t) \end{aligned} \qquad (2.6)$$

To understand the noise effects to the parameter estimation in applying least squares algorithms, from (2.5), in the case of noise free ($e(t) = 0$), the parameter estimates are

unbiased. If noise (even white noise) exists, a very practical situation, directly using least square algorithms will give biased estimates; simply the noise contaminated output appears in both the dependent variable and regression terms associated with the denominator polynomial *b*(*t*). This problem will be discussed and resolved in the next section.

2.2 ELS parameter estimation

This has been derived into following least squares formulation (Billings and Zhu 1991)

$$\hat{\Theta} = \left[\Phi^T \Phi - \sigma_e^2 \Psi\right]^{-1} \left[\Phi^T \mathrm{Y} - \sigma_e^2 \psi\right] \quad (2.7)$$

Where

$\hat{\Theta}$ is the estimate of parameter vector Θ, σ_e^2 is the model error variance,

Φ is the regressor matrix and formulated with data length *N*

$$\Phi^T = \begin{bmatrix} \varphi_n^T(1) & \cdots & \varphi_n^T(N) \\ \varphi_d^T(1) & \cdots & \varphi_d^T(N) \end{bmatrix} \quad (2.8)$$

Y is the dependent variable vector and composed of

$$\mathrm{Y} = \begin{bmatrix} Y(1) & \cdots & Y(N) \end{bmatrix}^T \quad (2.9)$$

Ψ is the error matrix associated with $\Phi^T \Phi$ and formulated as

$$\Psi = \begin{bmatrix} 0 & \cdots & 0 \\ 0 & \cdots & \sum_{k=1}^{N} p_d^T(t) p_d(t) \end{bmatrix} \qquad p_d(t) = \begin{bmatrix} p_{d2}(t) & \cdots & p_{ddden}(t) \end{bmatrix} \quad (2.10)$$

ψ is the error vector associated with $\Phi^T \mathrm{Y}$ and formulated as

$$\psi = \begin{bmatrix} 0 \\ \sum_{k=1}^{N} p_d^T(t) p_{d1}(t) \end{bmatrix} \quad (2.11)$$

An iterative procedure has been developed to obtain the unbiased parameter estimates, which is listed below:

Step 1, use an ordinary least squares algorithm (set to a null matrix Ψ and vector ψ in (2.7)) to compute parameter vector $\hat{\Theta}$. This estimate provides initial values for subsequent iterative computations.

Step 2, Compute the model error sequence by $e(t) = y(t) - \frac{a(\cdots, \hat{\Theta})}{b(\cdots, \hat{\Theta})}$. Then estimate the model error variance by $\sigma_e^2 = \frac{1}{N}\sum e^2(t)$.

Step 3, update matrices Φ and Ψ, vectors $\Phi^T Y$ and ψ through formulations (2.8) to (2.11).

Step 4, use the extended least squares algorithm (2.7) to estimate parameter vector $\hat{\Theta}$.

Step 5, Go back to step 2 and repeat until the parameter estimate vector $\hat{\Theta}$ and model error variance σ_e^2 converge to constant values.

2.3 Statistics of the estimator

The bias and variance of the estimates have been analysed (Zhu and Billings 1991) in case of model error *e*(*k*) is reduced to an uncorrelated noise sequence, which are given as follows.

$$
\begin{aligned}
&Bias(\hat{\Theta}) = 0 \\
&Cov(\hat{\Theta}) \approx \sigma_e^2 \sigma_b^2 \left[\Phi^T \Phi - \sigma_e^2 \Psi\right]^{-1}, \quad for\ \sigma_e^2 \prec 1, \qquad (2.12) \\
&\sigma_b^2 = E\left[b^2(t)\right], \quad E[*] \text{ is the expectation function}
\end{aligned}
$$

2.4 Summary

Expressing the rational model in terms of a linear in the parameters model makes it possible to use least squares algorithms for parameter estimation, on the other hand the expansion induces inherent correlated errors to give biased estimates while existing linear squares algorithms are applied directly. Therefore the extended least squares algorithm has been developed to set up a basis for concise and unbiased rational model parameter estimation from data sequences.

3 Back propagation (BP) algorithm for parameter estimation (Zhu 2003)

3.1 BP parameter estimation

The rational model can be treated as a neural network, as shown in Figure 1, which is structured with an input layer of regression terms $p_{nj}(t)$ and $p_{dj}(t)$, a hidden layer of *a*(*t*) and *b*(*t*) with linear activation functions, an output layer of *y*(*t*) with a ratio activation function. The activation function at the output layer is the division operation of the numerator polynomial divided by the denominator polynomial.

Leung and Haykin (1993) presented a rational function neural network. The shortcomings compared with the current study are that Leung and Haykin's network does not have a generalised rational model structure and correlated errors are not accommodated. Hence Leung and Haykin's parameter estimation algorithm cannot provide unbiased estimates with noise corrupted data, and is essentially a special implementation of the procedure of Billings and Zhu (1991) and Zhu and Billings (1991) in the case of noise free data.

To derive the parameter estimation algorithm, with reference to model expressions of (2.5) and (2.6), define the error between the measured output and the model output as

$$e(t) = \frac{1}{2}\left[y(t) - \hat{y}(t)\right]^2 \quad (3.1)$$

The task is to determine a set of parameters in model (2.5) so that the squared error in (3.1) is minimised. The parameter determination can be described as parameter training when the model is interpreted as a neural network structure. To train the parameters θ_{nj} associated with the numerator polynomial $a(t)$, the local gradients are given by

$$\frac{\partial e(t)}{\partial \theta_{nj}} = -\left[y(t) - \hat{y}(t)\right]\frac{\partial e(t)}{\partial a(t)}\frac{\partial a(t)}{\partial \theta_{nj}} = -\left[y(t) - \hat{y}(t)\right]\frac{1}{b(t)}\, p_{nj}(t) \quad (3.2)$$

To train the parameters θ_{dj} associated with the denominator polynomial $b(t)$, the local gradients are given by

$$\frac{\partial e(t)}{\partial \theta_{dj}} = -\left[y(t) - \hat{y}(t)\right]\frac{\partial e(t)}{\partial b(t)}\frac{\partial b(t)}{\partial \theta_{dj}} = \left[y(t) - \hat{y}(t)\right]\frac{a(t)}{b^2(t)}\, p_{dj}(t) \quad (3.3)$$

To carry out recursive computation, a time index k is introduced to the parameter sets $\theta_{nj}(k)$ and $\theta_{dj}(k)$ as they are updated following the time sequence. Convergence has been proved with $\lim_{k\to\infty}\theta_{nj}(k) \to \theta_{nj}$ $\lim_{k\to\infty}\theta_{dj}(k) \to \theta_{dj}$ (Zhu 2003). Therefore the parameter variation at time k+1 can be defined as follows

$$\begin{aligned}
\Delta\theta_{nj}(k+1) &= \theta_{nj}(k+1) - \theta_{nj}(k) = -\eta_{nj}\frac{\partial e(k+1)}{\partial \theta_{nj}(k+1)} \\
\Delta\theta_{dj}(k+1) &= \theta_{dj}(k+1) - \theta_{dj}(k) = -\eta_{dj}\frac{\partial e(k+1)}{\partial \theta_{dj}(k+1)}
\end{aligned} \quad (3.4)$$

where η_{nj} and η_{dj} are the learning rates.

The integrated parameter estimation procedure can be summarised in terms of the BP computation as follows:

I Initialisation
Set all model parameters to uniformly distributed random numbers with zero mean and small variance, or set up the initial parameters using the prior information extraction techniques, such as a least squares estimator.

II Parameter estimation
For l = 0 to L-1 (epoch number)
    For k = 1 to N (training data length)
    Forward computation

$$a(k+1)=\sum_{j=1}^{num} p_{nj}(k+1)\theta_{nj}(k) \quad b(k+1)=\sum_{j=1}^{den} p_{dj}(k+1)\theta_{dj}(k)$$
$$\hat{y}(k+1)=\frac{a(k+1)}{b(k+1)} \qquad e(k+1)=y(k+1)-\hat{y}(k+1) \tag{3.5}$$

Backward computation

$$\frac{\partial e(k+1)}{\partial \theta_{nj}(k+1)}=-e(k+1)\frac{1}{b(k+1)}p_{nj}(k+1)$$
$$\frac{\partial e(k+1)}{\partial \theta_{dj}(k+1)}=e(k+1)\frac{a(k+1)}{b^2(k+1)}p_{dj}(k+1) \tag{3.6}$$

$$\theta_{nj}(k+1)=\theta_{nj}(k)-\eta_{nj}\frac{\partial e(k+1)}{\partial \theta_{nj}(k+1)}$$
$$\theta_{dj}(k+1)=\theta_{dj}(k)-\eta_{dj}\frac{\partial e(k+1)}{\partial \theta_{dj}(k+1)} \tag{3.7}$$

end

end

Initialisation

The first step in the above computation is to initialise the parameters. A sensible choice will provide tremendous help for the resultant parameter estimation. A wrong choice of the initial parameters can lead to two critical phenomena, premature saturation and divergence. In the case of premature saturation, the instantaneous sum of squared errors remains almost constant for some period of time during the learning process. In the case of divergence, the sum of squared errors tends to become very large. A general initialisation technique (Haykin 1994) is to uniformly randomise the initial parameters with zero mean and small variance. In this study, the initial parameters will be determined using a prior information extraction technique. Since the model structure is assumed to be known in advance, an ordinary or extended least squares algorithm can be applied to obtain initial estimates. Although biased they should not be far away from the true parameters in Euclidean distance.

Learning rate

The BPE algorithm provides an approximation to the trajectory in parameter space computed in terms of steepest descent. The smaller the learning rate, the smaller the change to the model parameters will be from one iteration to the next and the smoother the trajectory will be in parameter space. However a small learning rate will possibly cause slower convergence of the parameter estimation. If the learning rate is set too large, the estimator may become unstable. The choice of learning rates is discussed below:

i) Constant

$$\eta_{nj}=\text{constant}_1 \quad \eta_{dj}=\text{constant}_2 \tag{3.8}$$

The disadvantage is that the learning rate cannot properly adjust in the early stage (large learning rate) and in the later stage (small learning rate) of the learning process.

ii) Inversely decayed sequence

$$\eta_{nj}(k) = \eta_{dj}(k) = \frac{1}{k} \qquad (3.9)$$

The disadvantage is that the learning rate decays too fast and this may not be suitable when long training data sequences are available.

iii) Generalised delta rule (Rumelhart, Hinton, and Williams 1986, Jacobs 1988)

$$\begin{aligned}\theta_{nj}(k+1) &= \theta_{nj}(k) - \eta_{nj}\frac{\partial e(k+1)}{\partial \theta_{nj}(k+1)} + \alpha_{nj}\left[\theta_{nj}(k) - \theta_{nj}(k-1)\right] \\ \theta_{dj}(k+1) &= \theta_{dj}(k) - \eta_{dj}\frac{\partial e(k+1)}{\partial \theta_{dj}(k+1)} + \alpha_{dj}\left[\theta_{dj}(k) - \theta_{dj}(k-1)\right]\end{aligned} \qquad (3.10)$$

where α_{nj} and α_{dj} are positive numbers called momentum constants. This has been extensively studied and can increase the learning rate and avoid instability. However, this choice involves more computation and requires the know-how to set up the parameters α_{nj} and α_{dj}.

iv) Linearly decayed sequence
To overcome the above disadvantages, a new learning rate sequence has been proposed as follows (Zhu 2003)

$$\eta_{nj}(k) = \eta_{dj}(k) = -\frac{\eta_0 - \eta_{end}}{LN}(lN + k) + \eta_0 \qquad (3.11)$$

where L is the number of iteration epoch, N is the training data length, l (= 0, …, L-1) for the lth epoch, and $\eta_0 \succ \eta_{end}$ are positive constants for the initial and end learning rates respectively. Figure 2 shows the learning rate variation.

Stopping criterion
Two criteria (Kramer and Sangiovanni-Vincentelli 1989) have been used to stop neural network training. These are formulated by means of local or global minimum of the error surface.

i) Gradient stopping criterion
This states that the BP converges when the Euclidean norm of the gradient vector reaches a sufficiently small gradient threshold. The drawback of this criterion is longer training and complexity in the computation of the gradient vector.

ii) Error measure stopping criterion
This states that the BP converges when the absolute rate of change in the mean squared error per epoch is sufficiently small. Typically 0.01 percent per epoch is used. The drawback is that the error may still be correlated for non-linear models, even though the mean squared error or its variation rate is very small (Billings and Woon 1983, Billings and Zhu 1994b).

To overcome the above drawbacks, a higher order correlation test (Billings and Zhu 1994b, 1995), introduced for non-linear model validation, can be used as a stopping criterion. The tests are described below:

$$R_{\chi e^2}(\tau)=\frac{\sum_{t=1}^{N}\left(\chi(t)-\bar{\chi}\right)\left(e^2(t-\tau)-\overline{e^2}\right)}{\sqrt{\left[\sum_{t=1}^{N}\left(\chi(t)-\bar{\chi}\right)^2\right]\left[\sum_{t=1}^{N}\left(e^2(t)-\overline{e^2}\right)^2\right]}}$$
$$R_{\chi u^2}(\tau)=\frac{\sum_{t=1}^{N}\left(\chi(t)-\bar{\chi}\right)\left(u^2(t-\tau)-\overline{u^2}\right)}{\sqrt{\left[\sum_{t=1}^{N}\left(\chi(t)-\bar{\chi}\right)^2\right]\left[\sum_{t=1}^{N}\left(u^2(t)-\overline{u^2}\right)^2\right]}} \qquad (3.12)$$

where

$$\chi(t)=y(t)e(t) \qquad \bar{\chi}=\frac{1}{N}\sum_{t=1}^{N}\chi(t)$$
$$\overline{u^2}=\frac{1}{N}\sum_{t=1}^{N}u^2(t) \qquad \overline{e^2}=\frac{1}{N}\sum_{t=1}^{N}e^2(t) \qquad (3.13)$$

When the higher order correlation functions $R_{\chi e^2}(\tau)$ and $R_{\chi u^2}(\tau)$ satisfy

$$R_{\chi e^2}(\tau)=\begin{cases}k\succ 0 & \tau=0\\ 0 & otherwise\end{cases} \qquad (3.14)$$
$$R_{\chi u^2}(\tau)=0 \quad \forall\tau$$

the estimated parameters are considered to be unbiased. Otherwise the training procedure will continue until the above conditions are satisfied. In practice the 95% confidence limits, $\pm 1.96/\sqrt{N}$, are used as the stopping thresholds.

3.2 Statistics of the estimator

For the bias

$$Bias\,\theta_{nj}=\theta_{nj}-E\left[\theta_{nj}(k+1)\right]=E\left[e(k+1)\right]E\left[\frac{1}{b(k+1)}\,p_{nj}(k+1)\right]=0$$
$$Bias\,\theta_{dj}=\theta_{dj}-E\left[\theta_{dj}(k+1)\right]=E\left[e(k+1)\right]E\left[\frac{a(k+1)}{b^2(k+1)}\,p_{nj}(k+1)\right]=0 \qquad (3.15)$$

Therefore the estimator will produce unbiased estimates of the non-linear rational model parameters. For the variance

$$Var\theta_{nj} = E\left[\left(\theta_{nj}(k+1) - E\left[\theta_{nj}(k+1)\right]\right)^2\right] = \sigma_e^2\sigma_{nj}^2$$
$$Var\theta_{dj} = E\left[\left(\theta_{dj}(k+1) - E\left[\theta_{dj}(k+1)\right]\right)^2\right] = \sigma_e^2\sigma_{dj}^2 \qquad (3.16)$$

where

$$\sigma_{nj}^2 = E\left[\left(\frac{1}{b(k+1)}\, p_{nj}(k+1)\right)^2\right] \quad \sigma_{dj}^2 = E\left[\left(\frac{a(k+1)}{b^2(k+1)}\, p_{dj}(k+1)\right)^2\right] \qquad (3.17)$$

Therefore the variance of each parameter estimate is associated with the noise variance and the second order moment of the regression term.

3.3 Summary

The BP based algorithm provides an alternative approach to estimate rational model parameters. Obviously there is no need to expand rational models into linear in the parameters expressions. Some techniques such as learning rate and stopping threshold play important roles in the iterative training and learning process, which need to be carefully considered in the computations.

4 Orthogonal Least Squares (OLS) algorithm for term selection and parameter estimation (Zhu and Billings 1993, Billings and Zhu 1994)

4.1 Orthogonal parameter estimation

Consider an orthogonal transformed expression of (2.5)

$$\begin{aligned} Y(t) &= \sum_{j=1}^{num} p_{nj}(t)\theta_{nj} - y(t)\sum_{j=2}^{den} p_{dj}(t)\theta_{dj} + \zeta(t) \\ &= \omega(t)G + \zeta(t) \\ &= \sum_{j=1}^{num} w_{nj}(t)g_{nj} - y(t)\sum_{j=2}^{den} w_{dj}(t)g_{dj} + \zeta(t) \end{aligned} \qquad (4.1)$$

where

$$\begin{aligned} \omega(t) &= \left[\omega_n(t) \quad \omega_d(t)\right] = \left[\omega_{n1}(t) \quad \cdots \quad \omega_{nnum}(t) \quad \omega_{d2}(t)y(t) \quad \cdots \quad \omega_{dden}(t)y(t)\right] \\ G^T &= \left[G_n \quad G_d\right] = \left[g_{n1} \quad \cdots \quad g_{nnum} \quad g_{d2} \quad \cdots \quad g_{dden}\right] \end{aligned} \qquad (4.2)$$

Define the orthogonal regression matrix W as

$$W = \Phi T^{-1}$$

$$W^T = \left[\omega^T(1) \quad \omega^T(N)\right] = \begin{bmatrix} \omega_n^T(1) & \cdots & \omega_n^T(N) \\ \omega_d^T(1) & \cdots & \omega_d^T(N) \end{bmatrix} \qquad (4.3)$$

$$T = \begin{bmatrix} 1 & t_{12} & . & . & . & t_{1n} \\ 0 & 1 & . & . & . & . \\ 0 & 0 & 1 & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & t_{n-1n} \\ 0 & 0 & . & . & . & 1 \end{bmatrix}$$

where Φ is the regressor matrix defined in (2.8). There are several typical algorithms to compute the elements of *T*, such as Gram-Schmidt, Householder, or Givens transforms (Billings and Zhu 1994a). The orthogonality of the matrix *W* gives

$$W^T W = D = diag\{d_{n1} \quad \cdots \quad d_{nnum} \quad d_{d2} \quad \cdots \quad d_{dden}\} \qquad (4.4)$$

The unbiased parameter vector *G* is estimated with

$$\hat{G} = \left[W^T W - \sigma_e^2 \Psi_{orth}\right]^{-1} \left[W^T \mathrm{Y} - \sigma_e^2 \psi_{orth}\right] \quad (4.5)$$

where

$$\Psi_{orth} = diag\left\{\overline{e_{n1}^2} \quad \cdots \quad \overline{e_{nnum}^2} \quad \overline{e_{d2}^2} \quad \cdots \quad \overline{e_{dden}^2}\right\}$$
$$\psi_{orth} = \left[\overline{p_{d1}e_{n1}} \quad \cdots \quad \overline{p_{d1}e_{nnum}} \quad \overline{p_{d1}e_{d2}} \quad \cdots \quad \overline{p_{d1}e_{dden}}\right]^T \qquad (4.6)$$

where over bar denotes time average value. The computation of the elements in (4.6) can be found in (Zhu and Billings 1993). Accordingly the original parameter vector in (2.5) can be recovered by

$$\Theta = T^{-1} G \qquad (4.7)$$

4.2 Structure detection (term selection)

The orthogonal property can be applied to select regression terms and estimate the associated parameters simultaneously. A criterion of Error Reduction Ratio (ERR) has been introduced to select the most important regression terms to contribute to reduce estimation errors from a large candidate term base (Billings and Zhu 1994a). In definition, individual ERR from numerator and denominator polynomials is calculated respectively by

$$err_{nj} = \frac{g_{nj}^2 \overline{w_{nj}^2}}{\sigma_Y^2 \sigma_b^2} - \frac{g_{nj}^2 \overline{e_{nj}^2} \sigma_e^2 + 2 g_{nj} \overline{e_{nj} b} \sigma_e^2}{\sigma_Y^2 \sigma_b^2}$$

$$err_{dj} = \frac{g_{dj}^2 \overline{w_{dj}^2}}{\sigma_Y^2 \sigma_b^2} - \frac{g_{dj}^2 \overline{e_{dj}^2} \sigma_e^2 + 2 g_{dj} \overline{e_{dj} b} \sigma_e^2}{\sigma_Y^2 \sigma_b^2} \quad (4.8)$$

$$\sigma_Y^2 = \frac{1}{N} \sum Y^2(k) \quad \sigma_b^2 = \frac{1}{N} \sum b^2(k)$$

It should be noticed that the larger the ERR, the more contribution of its regression term to reduce the estimation errors. Therefore the candidate term with the largest ERR in each selection sequence should be taken in as the model regression terms.

An iterative learning procedure for rational structure detection has been developed as follows:

Step 1, Fit a deterministic rational model (that is no noise model) initially with 3 to 4 regression terms (choice of the terms comes from experience, actually it is not critical) in numerator and denominator polynomials respectively. Therefore the model error sequence $e(t)$ and its variance σ_e^2 can be approximately obtained.

Step 2, Set up a series of ascending weights of the model variance σ_e^2 (such as 0.1, 0.25, 0.5 0.75, 0.95, 1, 1, …, 1). Set up a cut-off point (COP), which can be determined by trail and error approach, as

$$COP \geq 1 - \frac{1}{\sigma_b^2} + \sum bias[err] + \frac{\sigma_e^2}{\sigma_Y^2}$$

$$bias[err] = \begin{cases} \dfrac{g_{nj}^2 \overline{e_{nj}^2} + 2 g_{nj} \overline{e_{nj} b} \sigma_e^2}{\sigma_Y^2 \sigma_b^2} & \text{numerator terms} \\ \dfrac{g_{dj}^2 \overline{e_{dj}^2} + 2 g_{dj} \overline{e_{dj} b} \sigma_e^2}{\sigma_Y^2 \sigma_b^2} & \text{denonimator terms} \end{cases} \quad (4.9)$$

Step 3, Do a search through a pre-set-up full rational model candidate term base to select significant terms according to the ERR values and estimate their associated parameters.

Step 4, Repeat steps 2 and 3 until the computations converge, the preset maximum number of the iterations exceeded, or a specified number of terms are selected in the model.

4.3 Summary

This algorithm delivers a more realistic procedure to fit models to data in case of which terms should be included in the models before estimating their associated parameters.

5 Model validation (Zhang, Zhu, and Longden 2007, 2009, Zhu, Zhang, and Longden 2007)

5.1 Introduction to correlation based model validation

Consider a generalised single input and single output (SISO) nonlinear parametric model.

$$y(t) = \hat{y}(t) + \varepsilon(t) = \hat{f}(\mathbf{y}^{t-1}, \mathbf{u}^{t-1}, \boldsymbol{\varepsilon}^{t-1}) + \varepsilon(t) \qquad (5.1)$$

where $\hat{f}(\cdot)$ is the identified nonlinear model. $\mathbf{y}^{t-1}$, $\mathbf{u}^{t-1}$, and $\boldsymbol{\varepsilon}^{t-1}$ are measured output, input and residual (the difference between the measured output and the one step ahead model predicted output) vectors with delayed elements from 1 to r respectively. It should be noticed that when above model is properly identified, the residual $\varepsilon(t)$ should be reduced to a random noise sequence denoted by $e(t)$ in this section (without being confused with those used in other sections) with zero mean and finite variance (Billings and Zhu 1994). In other words, the residual sequence from a properly identified model should be uncorrelated to the delayed residuals, inputs and outputs (Ljung, 1999).

It should be noticed that all validation methods developed based on nonlinear models have included all linear model validation as their simplified cases. However the linear model based validation test methods can and often do fail when applied to nonlinear model validation.

5.2 Combined ODACF and ODCCF for model validation

Combined omni-directional auto-correlation functions (ODACF) is defined as follows,

$$\begin{cases} if \ \ |\max(R_{\varepsilon\varepsilon})| > |\min(R_{\varepsilon\varepsilon})| \\ \quad \rho_{\varepsilon\varepsilon}(\tau) = \max(R_{\varepsilon\varepsilon}) \\ else \\ \quad \rho_{\varepsilon\varepsilon}(\tau) = \min(R_{\varepsilon\varepsilon}) \end{cases} \qquad (5.2)$$

Combined omni-directional cross-correlation functions (ODACF) is defined as follows,

$$\begin{cases} if \ \ |\max(R_{u\varepsilon})| > |\min(R_{u\varepsilon})| \\ \quad \rho_{u\varepsilon}(\tau) = \max(R_{u\varepsilon}) \\ else \\ \quad \rho_{u\varepsilon}(\tau) = \min(R_{u\varepsilon}) \end{cases} \qquad (5.3)$$

In the above definition

$$\begin{cases} R_{\varepsilon\varepsilon} = [r_{\alpha'\alpha'}(\tau), r_{\alpha'\varepsilon'}(\tau), r_{\varepsilon'\varepsilon'}(\tau), r_{\varepsilon'\alpha'}(\tau)] \\ R_{u\varepsilon} = [r_{\beta'\alpha'}(\tau), r_{\beta'\varepsilon'}(\tau), r_{u'\varepsilon'}(\tau), r_{u'\alpha'}(\tau)] \end{cases} \qquad (5.4)$$

$$\begin{cases} \alpha'(k) = |\varepsilon'(k)| - \dfrac{1}{N}\sum_{k=1}^{N}|\varepsilon'(k)| \\ \beta'(k) = |u'(k)| - \dfrac{1}{N}\sum_{k=1}^{N}|u'(k)| \end{cases} \qquad (5.5)$$

where $r_{**}(\tau)$ denotes correlation function and the prime ' in (5.4) to (5.5) denotes that the mean level has been removed from the corresponding data sequence.

Then the validity tests for a properly identified model are derived as

$$\begin{cases} \rho_{\varepsilon\varepsilon}(\tau) = 1, \tau = 0 \\ \rho_{\varepsilon\varepsilon}(\tau) = 0, otherwise \end{cases} \quad (5.6)$$

$$\rho_{u\varepsilon}(\tau) = 0, \forall \tau \quad (5.7)$$

Compared with the other correlation tests based methodologies, this approach enhances the power of nonlinear model validity tests and significantly reduces the number of correlation plots. For large N the correlation function estimates given in (5.2) and (5.3) are still asymptotically normal with zero mean and finite variance in accordance with the central limit theorem (Bowker and Lieberman 1972). The standard deviations are $1/\sqrt{N}$ and the 95% confidence limits are therefore approximately at $\pm 1.95/\sqrt{N}$.

5.3 Summary

It should be mentioned that the method is not only applicable to rational models, but also for validation of identified neural networks and fuzzy logic models (Zhang, Zhu, and Ashley 2009).

6 U-model and U-control (Zhu 1989, Zhu and Guo 2002)

6.1 Introduction

So far almost all control system design approaches, no matter if they are linear or nonlinear plant based, have taken a unique procedure, set up a specified control system performance/index and then by inverting the integrated function of the plant and design performance in some way, obtain the controller output (that is the input to the plant). There is no problem at all with such procedures for all linear plant based control system design subject to stability considerations, because the linear superposition principle makes the inverse function easily resolved. However when the plants are subject to nonlinear dynamic equations (particularly non-affine models), the inverse functions are more complicated, intractable, or even impossible with analytical solutions, except step by step numerical computations.

Research question one: Can a class of nonlinear dynamic plants, described by smooth nonlinear models in terms of polynomials, be designed directly using the approaches developed from linear control systems?

Research question two: Is there a general approach to resolve the inverse function of nonlinear plants?

With such insight, the U-model methodology has been proposed, where the origin, but not in general U-model expression appeared in one of the authors PhD thesis (Zhu 1989). In progression along this route, Zhu, Warwick and Douce (1991) further proposed to use a Newton-Raphson algorithm (Gerald 1987) root solver to obtain the controller output, which makes the U-model type of control system feasible. The first

time the U-model was named in a study of pole placement controller design for nonlinear plants was by Zhu and Guo (2002), which is a simple mapping from ordinary linear and nonlinear difference equations to time-varying polynomials in terms of the plant input $u(t\text{-}1)$ (that is the controller output). The U-model covers almost all existing smooth nonlinear discrete time models as subsets.

Since then, within the first decade, the U-model has been used in designing feedforward control of MIMO nonlinear systems (Ali, Fouad, Muhammad, and Jamil, 2010), Nonlinear leaking minimum square algorithm for inverse adaptive control (Butt, Muhammad, and Tahir, 2005), minimum square error IMC (internal model control) (Muhammad and Haseebiddon, 2005) and General Predictive Control of nonlinear plants (Du, Wu and Zhu, 2012) have also been investigated. It should be noted that the U-model associated publications are still in a very beginning stage. These have had no rigorous analytical description and top journal publication till today.

The other research question: How is the U-model design procedure different from classical procedures in terms of efficiency and effectiveness?

The questions have been the justification for initiating the study. Xu, Zhu, Zhao and Li (2013) have presented a comprehensive survey for the first decade of development on U-model based control system design.

6.2 U-model

Consider a general single input and single output (SISO) discrete time dynamic plant described by the following Polynomial model (P-model).

$$\begin{aligned} y(t) &= f(P(*),\ \Theta) \\ P(*) &= P(y(t-1)\ ,...,\ y(t-n),\ u(t-1)\ ,...,\ u(t-n)) \end{aligned} \tag{6.1}$$

where $y(t) \in R$ and $u(t-1) \in R$ are the output and input signals of the plant respectively at discrete time instant $t(1,\ 2,\ \ldots)$, n is the plant order. . $P(*) = P(y(t-1)\ ,...,\ y(t-n),\ u(t-1)\ ,...,\ u(t-n)) \in R^{L+1}$ is the regression variable vector spanned from the delayed outputs and inputs and $\Theta = [\theta_0 \quad \cdots \quad \theta_L] \in R^{L+1}$ is the associated parameter vector. Function $f(\cdot)$ is a smooth linear or non-linear function. The P-model can be further expressed in terms of regression equation as below

$$y(t) = \sum_{l=0}^{L} p_l(t)\theta_l$$

Where the regression terms $p_l(t)$ are the products of past inputs and outputs such as $u(t-1)y(t-3)$, $u(t-1)u(t-2)$, $y^2(t-1)$, and θ_l are the associated parameters. Typically, for example, linear time invariant difference equation based plant models and NARMA (nonlinear auto-regressive moving average) models.

The U-model is defined as, under a U mapping from P-model, a control oriented model expression below

$$y(t) = f(\Upsilon(*),\ \mathrm{U}(t-1))$$
$$\Upsilon(*) = \Upsilon(y(t-1)\ ,...,\ y(t-n),\ u(t-2)\ ,...,\ u(t-n), \Theta)$$
$$\mathrm{U}(t-1) = \begin{pmatrix} const & u(t-1) & u^2(t-1) & \cdots & u^M(t-1) \end{pmatrix}$$

Correspondingly its regression equation is given as below

$$y(t) = \sum_{j=0}^{M} \lambda_j(t) u^j(t-1) \tag{6.2}$$

This is expanded from the above nonlinear function *f*(.) as a polynomial with respect to *u*(*t*-1). where *M* is the degree of model input $u(t-1)$, parameter $\lambda_j(t)$ is a function of past inputs, outputs (*u*(*t*-2), …, *u*(*t*-*n*), *y*(*t*-1), …, *y*(*t*-*n*)), and the parameters $(\theta_0 \quad \cdots \quad \theta_L)$.

6.3 U-control --- U-model based control system design procedure

Define the desired plant output as $U(t)$, it is clear that setting

$$y(t) = U(t) \tag{6.3}$$

then

$$U(t) = \sum_{j=0}^{M} \lambda_j(t) u^j(t-1) \tag{6.4}$$

Accordingly the task of the design is to determine the desired plant output $U(t)$ according to a specified performance index, for example,

Pole Placement Control (PPC) (Zhu and Guo 2002)

$$RU(t) = Tw(t) - Sy(t)$$

General Predictive Control (GPC) (Du, Wu, and Zhu 2012)

$$J_p = E\left[Y_d^T Q Y_d + \Delta U^T \lambda \Delta U \right]$$

then by resolving one of the roots of (6.4) to obtain the controller output. The block diagram, Figure 3 shows a general U-model based closed loop control system structure. It should be noted that so far it is still an open research question of how to use the U-model approach to design rational model based control systems. Therefore fundamental research should be performed in the future research development.

6.4 Summary

Design of nonlinear dynamic control systems has been a widely recognised as a challenging issue. The key point in the design is a general model prototype with

conciseness, flexibility and manipulability while keeping little loss of precision. This is the origin of the U-model with insight and procedure. In terms of time-varying parameter polynomials, the U-model almost covers all existing smooth nonlinear discrete time models as its subsets. Based on the U prototype, without complex transformation to its original model and no need for linearization at all, a nonlinear control system can be directly designed by linear control system design approaches, , such as Pole Placement Control (PPC), general predictive control (GPC), Sliding Mode Control (SMC), and so on. Even for linear control system design, the U-model provides another procedure for solutions.

The major contribution of the U-model based design procedure can be listed in order

1) In methodology, those well-known approaches developed from linear systems can be directly applied to nonlinear control system design, which significantly reduces the design complexity and effectively provides straight forward computational algorithms.
2) In design, it obtains desired plant output first (compared to designing desired controller output in classical framework) and then works out the controller output from U-model in a relaxed root-resolving routine (compared to resolving complex solutions from the inverse of the whole designed systems).
3) For linear control system design it provides new insight and solutions within a more general and effective frame work.

7 Conclusions

With regard to the presented algorithms, the ELS algorithm sets up a solid and concise basis for expanding this classical approach into nonlinear rational model term selection and parameter estimation. The insight presented in the study could be a useful indication to stimulate other complex model identification with proper extension of classical least squares algorithms. The BP algorithm provides a different angle to study the model parameter estimation issues. However the model structure detection via PB computation (i.e. regression term selection) still remains unresolved. The OLS algorithm provides a step by step procedure to select model regression terms and estimate their associated parameters simultaneously. The new model validation method provides a generic routine to effectively examine the model validity with significantly reduced correlation plots. In control system design, the U-model and hence U-control provide a new procedure from a classical framework, which could provide a concise solution for rational model based control systems design in the future studies. It should be noted again that the structure of the rational model brings various advantages, which cannot be easily contributed through polynomial model sets. On the other hand the rational model presents a number of new challenges for research, demonstration and applications.

Acknowledgement

This work is partially supported by the National Nature Science Foundation of China under Grant 61004080, 61273188, Shandong Provincial Natural Science Foundation under Grant ZR2011FM003, China and the Fundamental Research Funds for the Central Universities of China, Development of key technologies project of Qingdao Economic and Technological Development Zone under Grant 2011-2-52, Taishan Scholar Construction Engineering Special funding. Finally the authors are grateful to the editors and the anonymous reviewers for their helpful comments and constructive suggestions with regard to the revision of the paper.

References

Ali, S.S.A., Fouad, M.A., Muhammad, S., and Jamil, M.B., U-model based learning feedforward control of MIMO nonlinear systems, Electrical Engineering, 91(8), pp. 405-415, 2010.

Billings, S.A. and Chen, S., Identification of non-linear rational systems using a prediction error estimation algorithm, Int. J. Sys., Science, Vol. 20, pp. 467-494. 1989.

Billings, S.A. and Mao, K.Z., Structure detection for nonlinear rational models using genetic algorithms, Int. J. Sys. Science, Vol. 29, pp. 223-231, 1998.

Billings and Woon, W.S.F., Structure detection and model validity tests in the identification of nonlinear systems, IEE proceedings, PtD, Vol. 130, pp. 190-199, 1983.

Billings, S.A. and Zhu, Q.M., Rational model identification using extended least squares algorithm, Int. J. Control, Vol. 54, pp. 529-546, 1991; A structure detection algorithm for nonlinear dynamic rational models, ibid., Vol. 59, pp. 1439-1463, 1994a; Nonlinear model validation using correlation tests, ibid., Vol. 60, pp. 1107-1120, 1994b; Model validation tests for multivariable nonlinear models including neural networks, ibid., Vol. 62, pp. 749-766, 1995.

Bowker, A. H., and Lieberman, G. J., Engineering Statistics, Prentice Hall, 1972.

Butt, N.R., Muhammad, S., and Tahir, K. An adaptive root-solving controller for tracking of nonlinear dynamic plants, Proceeding of the 2005 IEEE International Conference on Industrial Electronics and Control Applications, Ecuador, pp. 144-149, 2005.

Chen, S. and Billings, S.A., Representations of nonlinear system: the NARMAX model, Int. J. Control, Vol. 48, pp.1013-1032, 1989.

Dimitrov, S.D. and Kamenski, D.I., A parameter estimation method for rational functions, Computers and Chemical. Engineering, Vol. 15, pp. 657-662, 1991.

Du, W.X., Wu, X.L, and Zhu, Q.M., Direct design of U-Model based generalized predictive controller (UMGPC) for a class of nonlinear (polynomial) dynamic plants, Proc. Instn. Mech. Enger, Part I: Journal of Systems and Control Engineering, V 226, pp. 27-42, 2012.

Ford, I., Titterington, D.M., and Kitsos, C.P., Recent advances in non-linear experimental design, Technometrics, Vol. 31, pp. 49-60, 1989.

Gerald, C.F., Applied Numerical Analysis, Addison-Wesley, Reading, Massachusetts, 1978.

Haber, R. and Unbehauen, H., Structure identification of nonlinear dynamic system --- a survey on input/output approaches, Automatica, Vol. 26, pp.651-677, 1990.

Haykin, S., Neural networks, Macmillan College Publishing Company, Oxford, 1994.

Jacobs, R.A., Increased rates of convergence through learning rate adaptation, Neural Networks, Vol. 1, pp.295-307, 1988.

Kambhampati, C., Mason, J.D., and Warwick, K., A stable one-step-ahead predictive control of nonlinear systems, Automatica, Vol. 36, pp. 485-495, 2000.

Kamenski, D.I. Dimitrovl, S.D., Parameter estimation in differential equation by application of rational functions, Comput. Chem. Eng., Vol. 17, pp.643–651, 1993.

Kramer, A.H. and Sangiovanni-Vincentelli, A., Efficient parallel learning algorithms for neural networks, In Advances in Neural Information Processing Systems (Touretzky, D.S., ed.), pp. 40-48, San Mateo, CA: Morgan Kaufmann, 1989.

Leung, H. and Haykin, S., Rational function neural network, Neural Computation, Vol. 5, pp. 928-938, 1993.

Ljung, L, Analysis of recursive stochastic algorithms, IEEE, Transactions on Automatic Control, Vol. 22, pp.551-575, 1977.

Ljung, L., System Identification Theory for the User 2nd Edition, Prentice Hall, Upper Saddle River, NJ, 1999.

Muhammad, S. and Haseebiddon, M., U-model-based internal model control for non-linear dynamic plants, Proceedings of the Institution of Mechanical Engineers. Part I: Journal of Systems and Control Engineering, 219(6), pp. 449-458, 2005.

Narendra, K.S. and Parthasapathy, K., Identification and control of dynamical systems using neural networks, IEEE Transactions on Neural Networks, Vol. 1, pp. 4-27, 1990.

Ponton, J.W., The use of multivariable rational functions for non-linear data presentation and classification, Computers and Chemical Engineering, Vol. 17, pp.1047-1052, 1993.

Proll, T. and Karim, M.N., Model predictive PH control using real time NARX approach, AIChe J., Vol. 40, pp. 269-282, 1994.

Rumelhart, D.E., Hinton, G.E., and Williams, R.J., Learning representations by back propagating errors, Nature (London), Vol. 323, pp.533-536, 1986.

Sontag, E.D., Realization theory of discrete time nonlinear systems: Part I --- the bounded case, IEEE transaction on Circuits and Systems, Vol. 26, pp. 342-356, 1978; Polynomial response maps --- lecture notes in control and information sciences, Vol. 13, Springer-Verlag, Berlin, 1979.

Wang, L.X, Adaptive fuzzy systems and control, Prentice Hall, Englewood Cliffs, NJ, 1994.

Xu, F.X., Zhu, Q.M., Zhao, D.Y., and Li, S.Y., U-Model based design methods for nonlinear control systems–A survey of the development in the 1st decade, Control and Decision, Vol. 28, No. 7, pp. 961-971, 2013 (in Chinese).

Zhang, L.F., Zhu, Q.M., and Longden, A., A set of novel correlation tests for nonlinear system variables, Int. J. Systems Science, Vol. 38, pp. 47-60, 2007.

Zhang, L.F., Zhu, Q.M., and Longden, A., A correlation-test-based validation procedure for identified neural networks, IEEE Transactions on Neural Networks, Vol. 20, No. 1, pp. 1-13, 2009.

Zhu, Q.M., Identification and Control of nonlinear systems, PhD thesis, University of warwick, UK, 1989.

Zhu, Q.M., Warwick, K., and Douce, J.L., Adaptive general predictive controller for nonlinear systems, IEE Proc. – Control Theory Appl., Vol. 138, No 1, pp. 33-40, 1991.

Zhu, Q.M. and Billings, S.A, Recursive parameter estimation for nonlinear rational models, J. Sys. Engineering, Vol. 1, pp. 63-67, 1991; Parameter estimation for stochastic nonlinear models, Int. J. Control, Vol. 57, pp. 309-333, 1993.

Zhu Q.M. and Guo L.Z., A pole placement controller for nonlinear dynamic plants, Journal of Systems and Control Engineering, Proceedings of the Institution of Mechanical Engineers Part I, Vol. 216, No. 6, pp. 467-476, 2002.

Zhu, Q.M., A back propagation algorithm to estimate the parameters of nonlinear dynamic rational models, Applied Mathematical Modelling, Vol. 27, pp. 169-187, 2003.

Zhu, Q.M., An implicit least squares algorithm for nonlinear rational model parameter estimation, Applied Mathematical Modelling, Vol. 29, pp. 673-689, 2005.

Zhu, Q.M., Zhang, L.F., and Longden, A., Development of omni-directional correlation functions for nonlinear model validation, Vol. 43, pp. 1519-1531, Automatica, 2007.

Zhu, Q.M., Zhang, L.F., and Longden, A., A correlation test-based validity monitoring procedure for online detecting the quality of nonlinear adaptive noise cancellation, Int. J. of Systems Science, Vol. 41 pp. 1043-1055, 2010.

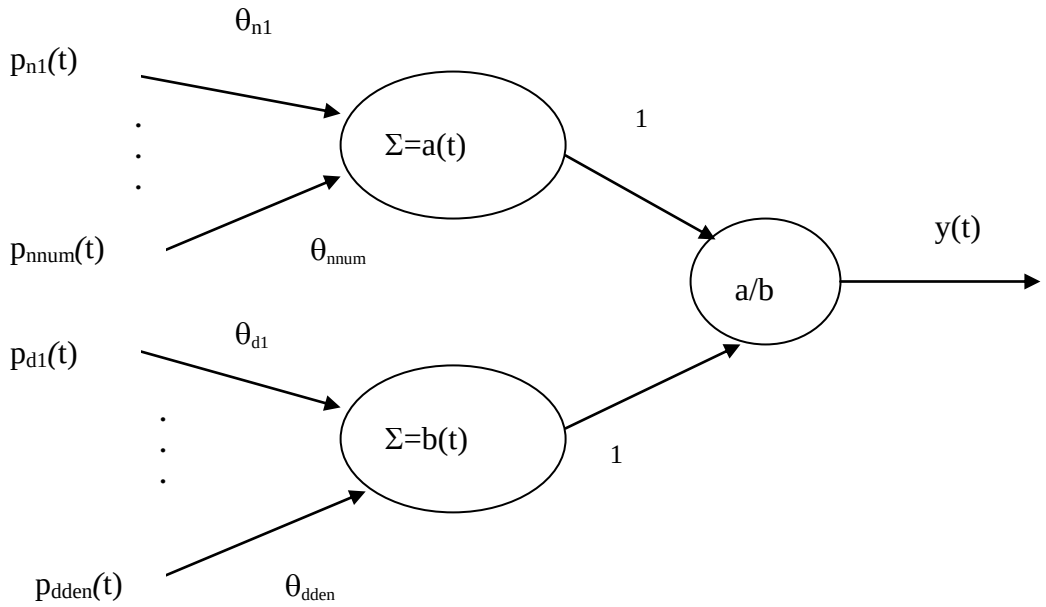


Figure 1 Rational neural network

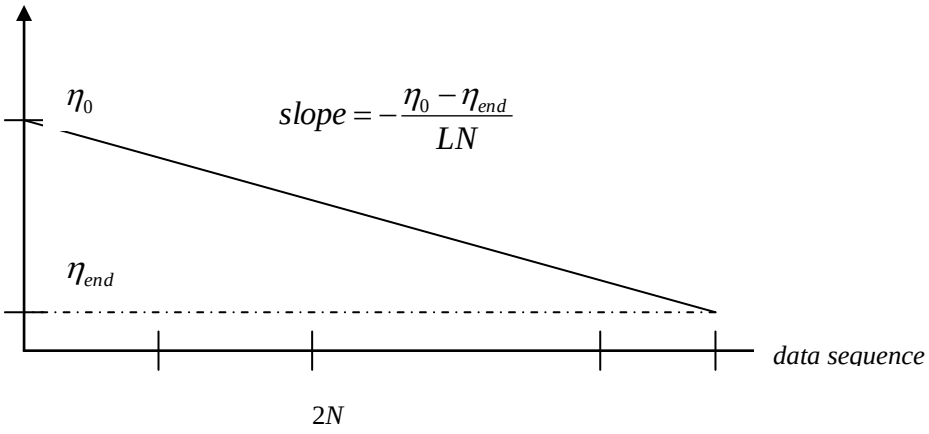


Figure 2 Linearly decayed learning rate

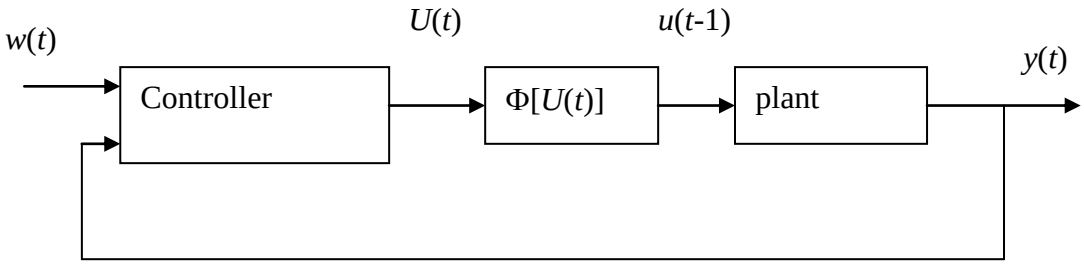
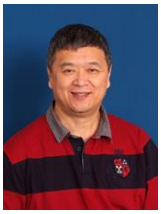


Figure 3 A general U-model based closed loop control system



Quanmin Zhu is Professor in control systems at Faculty of Environment and Technology, University of the West of England, Bristol, UK. He obtained his MSc in Harbin Institute of Technology, China in 1983 and PhD in Faculty of Engineering, University of Warwick, UK in 1989. His main research interest is in the area of nonlinear system modelling, identification, and control. His other research interest is in investigating electrodynamics of acupuncture points and sensory stimulation effects in human body, modelling of human meridian systems, and building up electro-acupuncture instruments. Currently Professor Zhu is acting as Member of Editorial Committee of Chinese Journal of Scientific Instrument, Editor (and Founder) of International Journal of Modelling, Identification and Control, and Editor of International Journal of Computer Applications in Technology.

Yongji Wang received the undergraduate degree in Electrical Engineering from Shanghai Railway University, Shanghai, P.R. China, the M.S. degree and the Ph. D degree in automation from Huazhong University of Science and Technology, Wuhan, P.R. China, in 1982, 1984 and 1990, respectively. He has been with Huazhong University of Science and Technology, Wuhan, P.R. China Since 1984, where he is currently a Professor of Control Engineering. His main interest is in intelligent control and autonomous mobile robot, and he has done research in neural network control, predictive control, adaptive control, and most recently, flight vehicle optimal control. Dr. Wang is a member of IEEE, the President of Hubei Automation Association, China, a member of council of the Chinese Automation Association，Standing member of council of Intelligent Robot Committee of Chinese Artificial Intelligence Society. He is an Regional editor (Asia and Pacific) of *Int. J. of Modelling Identification and Control*.

Dongya Zhao received BEng from Shandong University, Jinan, China, in 1998, MSc from Tianhua Institute of Chemical Machinery & Automation, Lanzhou, China, in 2002 and PhD from Shanghai Jiao Tong University, Shanghai, China, in 2009.

He was a research fellow in Nanyang Technological University during 7/2011 to 7/2012. Since 2002, he has been with College of Chemical Engineering, China University of Petroleum, where he is currently an Associate Professor. His research interests include robot control, sliding mode control, process modeling and control, nonlinear system control and analysis.

Shaoyuan LI received the Ph.D degree from the Department of Computer and System Science, Nan Kai University in 1997, he was a Lecture in the Department of Electrical Engineering, Hebei University of Technology, Tianjin from 1994 to 1998, and was with Dept. Automation, Shanghai Jiao Tong University as the post doctorial form 1998 to 2000, since then, he has been with the Department of Automation, Shanghai Jiao Tong University, China, and he is currently full Professor in this Department, and in charge of the Head of this department from 2004 to 2013. His research interest covers system identification and controller tuning, adaptive control, satisfying optimization and predictive control and so on, he is the author or coauthor of four monographs and about two hundred technical papers.

Stephen A. Billings is a Professor in the Department of Automatic Control and Systems Engineering, University of Sheffield, UK and leads the Signal Processing and Complex Systems research group. His research interests include system identification and information processing for nonlinear systems, narmax methods, model validation, prediction, spectral analysis, adaptive systems, nonlinear systems analysis and design, neural networks, wavelets, fractals, machine vision, cellular automata, spatio-temporal systems, fMRI and optical imagery of the brain, synthetic biology and related fields.