

# A general U-Block model based design procedure for nonlinear polynomial control systems

Q.M. Zhu\*<sup>#1, 2</sup>, D.Y. Zhao<sup>#2</sup>, Jianhua Zhang<sup>#3</sup>

#1 Department of Engineering Design and Mathematics  
University of the West of England  
Frenchy Campus Coldharbour Lane  
Bristol, BS16 1QY, UK  
Email: [quan.zhu@uwe.ac.uk](mailto:quan.zhu@uwe.ac.uk)

#2 Department of Chemical Industrial Equipment and Control Engineering  
College of Chemical Engineering  
China University of Petroleum  
Qingdao, 266580, China  
Email: [dyzhao@upc.edu.cn](mailto:dyzhao@upc.edu.cn)

#3 Institute of Electrical Engineering  
Hebei University of Science and Technology  
Shijiazhuang, 050018, China  
Email: [zhangjianhua@hebust.edu.cn](mailto:zhangjianhua@hebust.edu.cn)

\* Corresponding author

**Abstract:** The proposition of U-model concept (in terms of “providing concise and applicable solutions for complex problems”) and a corresponding basic U-control design algorithm was originated in the first author’s PhD thesis (Zhu 1989). The term of U-model was the first time appeared (not rigorously defined) in the first author’s other journal paper (Zhu and Guo 2002), which established a framework for using linear polynomial control system design approaches to design nonlinear polynomial control systems (in brief, linear polynomial approaches → nonlinear polynomial plants). This paper represents the next milestone work --- using linear state space approaches to design nonlinear polynomial control systems (in brief, linear state space approaches → nonlinear polynomial plants). The overall aim of the study is to establish a framework, defined as U-Block model, which provides a generic prototype for using linear state space based approaches to design the control systems with smooth nonlinear plants/processes described by polynomial models. For analysing the feasibility and effectiveness, Sliding Mode Control (SMC) design approach is selected as an exemplary case study. Numerical simulation studies provide a user-friendly step by step procedure for the readers/users with interest in their ad hoc applications. In formality, this is the first paper to present U-model oriented control system design in formal way and to study the associated properties and theorems. The previous publications, in the main, have been algorithm based studies and simulation demonstrations. In some sense this paper can be treated as a landmark for U-model based research from intuitive/heuristic stage to rigour/formal/comprehensive studies.

**Keywords:** U-model, U-Block model, U-state space model, NARMAX model, U-model based controller design, pole placement assignor, feasibility demonstration of sliding mode control.

## 1 Introduction

The main objective of control engineering is invariably to design a controlled system to best match the performance or design specifications. Standard procedures of control system design can be broadly classified into two groups according to whether the dynamic plant is linear or nonlinear. Linear control is a mature subject with a variety of powerful methodologies and a long history of successful industrial applications. Comparatively speaking nonlinear control is a more difficult subject, although some design schemes have been developed with ad hoc model structures (Slotine and Li 1991). Generalisation of control system design for nonlinear dynamic plants has been a hot and challenging topic in research and applications. It has been the researcher’s belief that it is possible to use linear controller design methodologies to design a large class of smooth nonlinear control systems, and therefore to simplify and generalise nonlinear control system design. This is the insight of the study.

State space model based linear control system design approaches have been well studied in research and adopted in wide ranges of applications (Maciejowski 1994, Ogata 2009). The fundamental request is the linear state space model. For linear polynomial models there are many realisation approaches to convert them into state space expressions, such as typical controllable and observable realisations. However while a nonlinear polynomial model, such as NARMAX (Non-linear AutoRegressive Moving Average with eXogenous input) models (Billings 2013), a very practical and predominantly used model structure in practice (Billings 2015, Zhu, Wang, Zhao, Li, and Billings 2015), is appeared, it is very difficult to convert it into a state space expression, nevertheless almost impossible into an equivalent linear state space model. It should make clear at the beginning of the study that piecewise linearization and linear approximation approaches are not discussed in this study, simply because these are not the essence of nonlinear control system design problems.

There have been two main streams plus a less attended research area in the related fields. The first one is to convert nonlinear state space models into linear expressions. The approach is the feedback linearization (input-state linearization and input-output linearization) (Slotine and Li 1991, Isidori 1995), which the central idea of feedback linearization is to convert, by coordinate transform, the non-linear model into a linear form, in order to cancel the non-linear dynamics in the closed-loop. Then use linear state space approaches to design the corresponding control systems. However this is a case by case approach with certain degree of skills in manipulating differential equations and selecting coordinates. Further shortage with the state space linearization approaches is that there is no way to deal with nonlinear polynomial model based control system design.

The second one is to treat nonlinear polynomial models as time varying linear models. One of the typical approaches is the state-dependent parameter (SDP) transformation (Taylor, Chotai, and Young 2009, Çimen 2010), which in the main, reduces the closed-loop system to a linear transfer function with the specified (desired) poles. Hence, assuming pole assignability at each sample, global stability of the non-linear system is guaranteed at the design stage. It is clear from these studies that a common strategy for the control of non-linear systems involves attempting to bring the original system into a quasilinear domain, before subsequently designing an appropriate linear control algorithm. In model structure, the parameters of this quasi-linear state-dependent parameter (SDP) are functionally dependent on other variables in the system (Young 2000). Although this provides a bridge from nonlinear polynomial model to linear time-varying state space expression, the transform is not unique and selection of SDP is subjective, personal and no clear rule to follow, which has been the main barrier for a wide range of users.

Although not widely attended, the third one is U-model methodology, which is a generic systematic approach to convert nonlinear polynomial model into a controller output  $u(t-1)$  based time-varying polynomial model. This has been studied (Zhu and Guo 2002, Muhammad and Haseebiddon 2005, Du, Wu, and Zhu 2012) for facilitating nonlinear control system design over the last decade. Consequently linear polynomial model based design approaches (such as pole placement and general predictive control can be directly used to design such nonlinear control systems (Zhu and Guo 2002, Du, Wu, and Zhu 2012). 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. It should be noted that this new approach doesn't request to linearize nonlinear plant models in design. It is just to use linear design approaches to directly design nonlinear plant model based control systems.
- 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 framework.

It should be noted that the U-model associated publications are still in a very beginning stage. Although various algorithms have been developed and bench tested with simulations and applications, they have not had rigorous analytical description and top journal publication till today. It is the author's belief that the next research progression should be the development to make U-model approach available for using linear state space based design approaches to design the control of nonlinear polynomial models described plants. The key issue in the on-going study is how to systematically obtain an equivalent linear state space models from its nonlinear polynomial model origin.

This paper takes the above described issues into study and proposes a general U-Block model based procedure for using linear state space based approaches to design the control systems with smooth nonlinear plants/processes described by polynomial models. It should be noted that this procedure is not the way using linear approximation to obtain linear polynomials and then convert to the corresponding state space realisations. For a nonlinear plant polynomial model, to design its control system with linear state space based approaches, the procedure can be ordered into the following step by step implementation.

- 1) Convert the nonlinear polynomial model into U-model expression (NB, not linear approximation).
- 2) By pole assignor (a pole placement algorithm), convert the U-model into an input-output linear closed loop model with assigned poles. This is defined as U-Block model, which is linear.
- 3) Convert the linear U-Block model into its state space realisation by commonly used realisation techniques. This is defined as U-state space model.
- 4) With reference to the U-state space model, using linear state space based design approaches, design the control system. For feasibility test, this study selects Sliding Mode Control (SMC) design. It should be noted, from the author's knowledge, so far there is no any existing approach to design SMC with nonlinear polynomial models. There are a few of representative publications using the terminology <SMC for nonlinear polynomial systems> (Basin and Calderon-Alvarez 2010, Basin and Rodriguez-Ramirez 2012), however all the publications have used <state space model with a nonlinear polynomial for each state variable>. Simply it is a common basis that **SMC need state space models**, no matter linear or nonlinear expressions.

The rest of the paper is organised into the following sections. In section 2, U-model is formally defined and the associated properties are presented to provide a proper description of the model structure and characteristics. Subsequently remarks with an example are presented for explanation and understanding of the conversion from classical polynomials to the U-model. In section 3, A U-model based control system design procedure is presented with step by step implementation, which provides a foundation for the development of Section 4. Remarks are used to show the difference and novelty compared with classical linear control system design approaches. In section 4, a U-Block model is proposed to establish a framework for using linear state space approaches to design nonlinear polynomial control systems. The related properties are analysed as well to provide assurance for applications. In section 5, Sliding Mode Control (SMC) design approach is selected as an exemplary case study to show the feasibility within U-Block model framework. In section 6, a series of computational experiments are conducted to demonstrate numerically the theoretically developed results. In section 7, a brief conclusion is drawn to summarise the results and the potential research in the future is briefly discussed as well.

## 2 U-model

### Definition of smooth and analytic function (Isidori 1995)

Let  $A$  be an open subset of  $R^n$  and  $f: A \rightarrow R$  a function. The value of  $f$  at  $x = (x_1 \cdots x_n)$  is denoted  $f(x) = f(x_1 \cdots x_n)$ . The function  $f$  is said to be a function of class  $C^\infty$  (or simply  $C^\infty$ , or, also smooth function) if its partial derivatives of any order with respect to  $x_1 \cdots x_n$  exist and are continuous. A function  $f$  is said to be analytic (sometimes denoted as  $C^\omega$  if it is  $C^\infty$  and for each point  $x^o \in A$  there exists a neighbourhood  $\chi$  of  $x^o$ , such that the Taylor series expansion of  $f$  at  $x^o$  converges to  $f(x)$  for all  $x \in \chi$ ).

**Remark 1:** This definition is well explained in classical calculus. In the contrast, nonsmooth systems are normally referred to those systems with switching or jumping functions, which are also called "hard nonlinearities" such as Coulomb friction, saturation, dead-zones, backlash, and hysteresis, and are frequently found in practical control systems (Slotine and Li 1991).

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

$$y(t) = f(P^*, \Theta) \quad (2.1)$$

$$P^* = P(y(t-1), \dots, y(t-n), u(t-1), \dots, u(t-n))$$

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

$$y(t) = \sum_{l=0}^L p_l(t) \theta_l \quad (2.2)$$

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  $\theta_l$  are the associated parameters.

It should be noted that this model set has been widely accepted as NARMAX (Non-linear AutoRegressive Moving Average with eXogenous input) models (Billings 2013).

#### Definition of U-model

U-model is defined as, under a U mapping from the above polynomial model, a controller output  $u(t-1)$  oriented polynomial below,

$$y(t) = f(Y^*, U(t-1))$$

$$Y^* = Y(y(t-1), \dots, y(t-n), u(t-2), \dots, u(t-n), \Theta) \quad (2.3)$$

$$U(t-1) = (const \ u(t-1) \ u^2(t-1) \ \dots \ u^M(t-1))$$

Correspondingly its regression equation is given as below,

$$y(t) = \sum_{j=0}^M \lambda_j(t) u^j(t-1) \quad (2.4)$$

This is expanded from the above nonlinear function  $f(\cdot)$  as a polynomial with respect to  $u(t-1)$ . where  $M$  is the degree of model input (controller output)  $u(t-1)$ , the time varying parameter vector  $\lambda(t) = [\lambda_0(t) \ \dots \ \lambda_M(t)] \in R^{M+1}$  is a function of past inputs and outputs ( $u(t-2), \dots, u(t-n), y(t-1), \dots, y(t-n)$ ), and the parameters  $(\theta_0 \ \dots \ \theta_L)$ .

**Property 1:** Let  $\varphi: R^{L+1} \rightarrow R^{M+1}$  be the map from polynomial model to U-model and its inverse be  $\varphi^{-1}$ , that is

$$f(P^*, \Theta) \xrightarrow{\varphi} f(Y^*, U(t-1)) \quad (2.5)$$

Then it has the following properties

- 1) The map is injective (one to one)
- 2) The map is surjective (onto)
- 3) Therefore the map is bijective as it is both injective and surjective.
- 4) The map is invertible.
- 5) The map does not change any both model characteristics, such as output response, stability, dynamics and statics.

**Proof:** In (2.2) the products of the term by parameter can be alternatively written as

$$p_l(t)\theta_l = r_l(t)u^j(t-1)\theta_l, \quad j \in 0,1,\dots,M \quad (2.6)$$

Accordingly (2.2) can be expressed as

$$y(t) = \sum_{l=0}^L p_l(t)\theta_l = \sum_{l=0}^L r_l(t)\theta_l u^j(t-1) = \sum_{j=0}^M \left( \sum_{l=0}^L (r_l(t)\theta_l) \right) u^j(t-1), \quad l \in 0,1,\dots,L \quad (2.7)$$

For each  $u^j(t-1)$ ,  $j \in 0,1,\dots,M$ , let  $\lambda_j(t) = \sum_{l=0}^L (r_l(t)\theta_l)$ , then it gives the U-model expression (2.4).  $\square$

**Remark 1:** With the U-model (2.4), there is nothing in change of the original polynomial model (2.2) properties such as dynamic, stability, and input and output external relationship. However by such mapping (re-organisation) of the model expression, U-model provides a generic prototype to facilitate control system design by using linear approaches. In model properties, U-model and NARMAX polynomial model are equivalent. In model expression, U-model is time varying in parameters and convenient for control systems design and NARMAX polynomial model is time constant in parameters and convenient for system identification.

**Remark 2:** Actually the above proof provides a model conversion algorithm.

**Remark 3:** Such an example, from polynomial model to the U-model conversion, is shown below. The polynomial model is

$$y(t) = 0.1y(t-1)y(t-2) - 0.5y(t-1)u^2(t-1) + 0.8u(t-1)u(t-2) \quad (2.8)$$

And the U-model can be determined in notation of (2.4)

$$y(t) = \lambda_0(t) + \lambda_1(t)u(t-1) + \lambda_2(t)u^2(t-1) \quad (2.9)$$

where  $\lambda_0(t) = 0.1y(t-1)y(t-2)$ ,  $\lambda_1(t) = 0.8u(t-2)$ , and  $\lambda_2(t) = -0.5y(t-1)$ .

### 3 U-control --- U-model based control system design procedure

**Hypothesis:** It is possible to use linear controller design methodologies to directly design a large class of smooth nonlinear plant based control systems, and therefore to simplify and generalise nonlinear control system design.

In order to use linear polynomial mode based design approaches, define the desired plant output as  $y_d(t)$ , which is specified either by designers or customers in advance. Accordingly the relationship between a specified plant output  $y_d(t)$  and the requested corresponding controller output  $u(t-1)$  can be expressed in terms of U-model

$$y_d(t) = \sum_{j=0}^M \lambda_j(t)u^j(t-1) \quad (3.1)$$

Accordingly this establishes a prototype for proposing a new two-step design procedure.

**Step 1 (design  $y_d(t)$ ):** The first task of the design is to determine the desired plant output  $y_d(t)$  according to a specified performance index, for example,

Pole Placement Control (PPC) (Zhu and Guo 2002) has been designed in terms of

$$Ry_d(t) = Tw(t) - Sy(t) \quad (3.2)$$

where  $y(t)$ ,  $y_d(t)$  and  $w(t)$  are the plant output, desired/designed plant output, and reference input respectively. The polynomials  $R$ ,  $S$ , and  $T$  are used to specify the desired plant output  $y_d(t)$ .

General Predictive Control (GPC) (Du, Wu, and Zhu 2012) has been designed in terms of

$$J_p = E[(Y - W)^T Q_1 (Y - W) + \Delta Y_d^T Q_2 \Delta Y_d] \quad (3.3)$$

where  $Y$ ,  $Y_d(t)$  and  $W(t)$  are the vectors of  $y(t)$ ,  $y_d(t)$  and  $w(t)$  respectively.  $Q_1$  and  $Q_2$  are the corresponding weighting coefficient matrices.

**Step 2 (work out  $u(t-1)$ ):** Then the remaining design task is to resolve one of the roots of (3.1) to obtain the controller output  $u(t-1)$ . That is

$$u(t-1) = \Psi \left[ y_d(t) - \sum_{j=0}^M \lambda_j(t) u^j(t-1) = 0 \right] \quad (3.4)$$

where  $\Psi[*]$  is a root-solving algorithm, such as Newton-Raphson algorithm (Chong and Zak 2013). A detailed analysis on the root solving issues has been presented (Zhu and Guo 2002).

**Remark 4:** It should be noted that any iterative algorithm takes time, however in the framework, the designed closed loop is stabilised, so that the controller output is stabilised. By feedback, at each step  $u(t-1)$  changes small which makes the root solver initial searching point is closed to its solution. According to a general proposition (Nash 2000, Chong and Zak 2013), many iteration algorithms are converged with good convergent rate while its starting searching point is sufficiently close to its solution point. In regarding to Newton-Raphson algorithm, it requests calculations of Jacobian and particularly inverse of Hessian matrix in iteration. If this involves multi decision variable search, inverse of Hessian matrix is very time consuming. For the U-model case, it is only a single variable  $u(t-1)$  search, therefore inverse of Hessian matrix is reduced to a single reciprocal calculation. A small number of iterations is sufficient to produce a 'good' step.

**Remark 5:** U-model is type of time varying polynomial, U-model based controller is actually an online algorithm. Because its controller must be updated from the time varying parameter vector  $\lambda(t) = [\lambda_0(t) \ \cdots \ \lambda_M(t)]$  in each sampling interval.

Block diagram Figure 1 shows a general U-model based pole placement control system structure with the proposed design procedure.

**Remark 6:** In methodology, the difference compared with linear control system design is the way to find controller output  $u(t-1)$  from resolving a root with (3.1).

**Remark 7:** Comparison of classical design procedure with the new procedure, let design criterion or performance index

$$J(C_1, C_2, \dots, C_n) \quad (3.5)$$

where  $C_1, C_2, \dots, C_n$  are the polynomials specifying the control system performance, such a typical example is the pole placement scheme (Astrom and Wittenmark 1995) with three polynomials  $R, S, T$  to specify the desired plant output, plant output, and reference respectively.

**In classical design procedure (work out controller output  $u(t-1)$ ),** resolve the criterion function to obtain the **controller output** (that is the plant input)  $u(t-1)$ . With this procedure, it, even for a linear system, requests plant model to obtain the solution from inversion of the equations. The more difficulty case is the nonlinear systems, which the inversion is almost impossible in the most of the time.

**In the new design procedure (work out desired plant output  $y_d(t)$  and then controller output  $u(t-1)$ ),** firstly resolve the criterion function to obtain the designed/desired **plant output**  $y_d(t)$  (NB not the controller output  $u(t-1)$ !). Then obtain controller output  $u(t-1)$  from the designed  $y_d(t)$  through U-model (3.1), that is, resolve one of the roots from the U-model (3.1). With this procedure, it does not request plant model (no matter line or nonlinear) in the first stage. It only uses the plant model, in U-model expression, to obtain controller

output  $u(t-1)$  in the second stage. For a nonlinear plant model, the calculation is merely to resolve one of the roots from the U-model (3.1). For a linear plant model, it has, from (3.1)

$$u(t-1) = \frac{y_d(t) - \lambda_0(t)}{\lambda_1(t)} \quad (3.6)$$

where  $\lambda_1(t)$  is the coefficient associated with  $u(t-1)$  (for linear time invariant models,  $\lambda_1(t)$  is a constant).  $\lambda_0(t)$  (none zero) is the summation of the rest of the terms in the linear model (Zhu and Guo 2002).

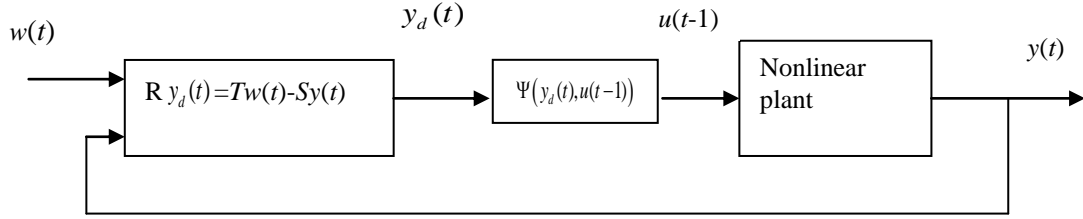


Figure 1 U-model based pole placement control system

#### 4 U-Block model and its state space realisation

##### 4.1 U-Block model

U-Block model is defined as a general linearised model from nonlinear polynomial model (2.2) through the conversion to U-model (2.4) and then assigned with required poles through a linear feedback control algorithm. In this study the pole assignor is formulated by a revised U-model based pole placement controller design algorithm. Figure 2 shows the U-Block model structure

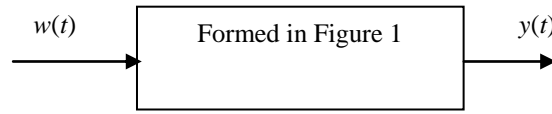


Figure 2 U-Block model

##### Pole assignor --- design of pole placement controller

A standard reference (Astrom and Wittenmark 1995) has been used to develop the following pole placement controller (Zhu and Guo 2002). Consider U-model of (2.4), a general controller can be structured by

$$Ru(t) = Tw(t) - Sy(t) \quad (4.1)$$

where  $w(t)$  is the reference for output target and  $R$ ,  $S$ , and  $T$  are the polynomials of the forward shift operator  $q$ , which are described by

$$\begin{aligned} R &= q^n + r_1 q^{n-1} + \dots + r_n \\ T &= t_0 q^m + t_1 q^{m-1} + \dots + t_m \\ S &= s_0 q^l + s_1 q^{l-1} + \dots + s_l \end{aligned} \quad (4.2)$$

where  $q$  is the forward operator,  $n$ ,  $m$ , and  $l$  are the orders of the polynomials  $R$ ,  $T$ , and  $S$  respectively. A designed controller must satisfy the causality conditions  $order(S) < order(R)$   $order(T) \leq order(R)$ , That is  $l < n$  and  $m \leq n$ .

The control law of (4.1) represents a negative feedback with transfer function  $-S/R$  and a feedforward with transfer function  $T/R$ . It thus has two degree of freedom.

For the proposed design procedure, as shown in Figure 1, rewrite (4.1) as

$$Ry_d(t) = Tw(t) - Sy(t) \quad (4.3)$$

By letting  $y(t) = y_d(t)$ , the designed U block model can be linked to the reference  $w(t)$  as

$$y_d(t) = \frac{T}{R+S} w(t) = \frac{T}{A_c} w(t) \quad (4.4)$$

where polynomial  $A_c$  is the closed loop characteristic equation and specified in advance, that is

$$R+S = A_c \quad (4.5)$$

To cancel the possible output offset in steady state, i.e., to make steady state error equal to zero at the controlled output, polynomial  $T$  is specified with

$$T = A_c(1) \quad (4.6)$$

The key idea of the design is to specify the desired closed loop characteristic polynomial  $A_c$ , then resolve the polynomials  $R$  and  $S$  through a Diophantine equation (the algorithm is given in Appendix). After the desired plant output  $y_d(t)$  designed, controller output  $u(t-1)$  can be determined by resolving one the root of U-model (3.1), which two step design procedure has been formulated in section 3.

**Theorem 1:** U-model based pole placement design procedure does not depend on plant model. Only the solution of the designed controller output involves in the plant model.

**Proof:** This directly comes from the relationship of (4.5). In classical approach (Astrom and Wittenmark 1995), the corresponding relationship is given by

$$AR + BS = A_c \quad (4.7)$$

Where  $A$  and  $B$  are the numerator polynomial and denominator polynomial of a plant model respectively, which indicate the classical design depending on plant model.  $\square$

**Remark 8:** Actually Theorem 1 provides a new design framework, using U-control design procedure once, then it can be applied to many different plant models, which only calculates the corresponding controller output each time from a given plant model.

**Pole assignability:** For linear systems this has been well studied (Young, Behzadi, Wang, and Chotai 1987, Talor, Chotai, and Young 2009). In principle, for linear case, model input variable (that is the controller output) should have some influence on the system output. For the nonlinear case, in terms of SDP model, Talor, Chotai, and Young (2009) derived similar condition. In regarding to the U-model structure, the condition of the pole assignability can be stated in the following proposition.

**Proposition:** For U-model (3.1), pole assignment (placement) algorithm described by (4.1) through to (4.6), can be resolved if and only if

$$\exists \lambda_j(t) \neq 0, j \in [1 \ \dots \ M] \Rightarrow u(t-1) \text{ has some influence on the system output (4.8).}$$

This is a direct inference of the condition for SDP based pole assignment algorithm (Talor, Chotai, and Young 2009).

#### 4.2 U-state space model

U-state space model is defined as the state space realisation of the U-Block model. Let the assigned poles as  $\alpha_1 \ \dots \ \alpha_k$ , then it is corresponding characteristic equation is



$$A_c = (q - \alpha_1) \cdots (q - \alpha_k) = q^k + a_{c1}q^{k-1} + \cdots + a_{ck} = 0 \quad (4.8)$$

With reference to (4.4) and (4.6), the assigned numerator polynomial of the U-Bock model is

$$T = A_c(1) \quad (4.9)$$

Correspondingly the U-Block model transfer function is

$$\frac{Y(q)}{W(q)} = \frac{A_c(1)}{q^k + a_{c1}q^{k-1} + \cdots + a_{ck}} \quad (4.10)$$

Consequently the U-state space model, in controllable realisation, can be expressed as

$$\begin{aligned} x(t+1) &= Ax(t) + Bw(t) \\ y(t) &= Cx(t) \end{aligned} \quad (4.11)$$

where the state equation

$$\begin{bmatrix} x_1(t+1) \\ x_2(t+1) \\ \vdots \\ x_{k-1}(t+1) \\ x_k(t+1) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & & & & \\ & & 0 & 0 & \cdots & 1 \\ -a_{ck} & -a_{c(k-1)} & -a_{c(k-2)} & \cdots & -a_{c1} \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_{k-1}(t) \\ x_k(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} w(t) \quad (4.12)$$

and the output equation

$$y(t) = [A_c(1), 0, \dots, 0, 0] \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_{k-1}(t) \\ x_k(t) \end{bmatrix} \quad (4.13)$$

**Remarks 9:** The state variable measurement can be obtained by standard observers (Ogata 2009) from the available input  $w(t)$  and output  $y(t)$ , while U state space model is assigned with the poles to make it completely observable.

**Remark 10:** The whole framework in using linear state space approaches to design control systems with smooth nonlinear polynomial plant models is shown in Figure 3.

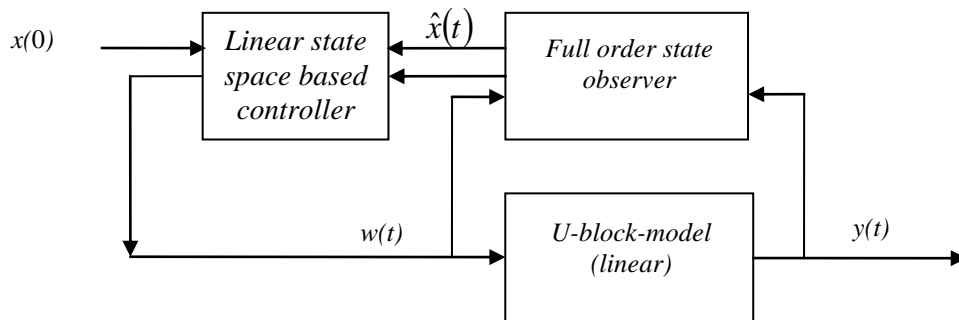


Figure 3 Diagram of U-block-model based control systems

### 4.3 Feasibility studies --- design of sliding mode control (SMC)

As this is the first time to establish such framework for using linear state space approaches to design control systems with nonlinear polynomial plant models, it should make clear at the beginning of this section that selecting SMC design approach is, in the main as an example, to demonstrate the feasibility of the developed design framework. Subsequently the associated comprehensive studies on SMC design will be described in the future work.

**Theorem 2 (feasibility theorem):** It is feasible to design SMC for nonlinear polynomial plant models, while all the following conditions are satisfied.

- C1) A corresponding U-Block model is existed.
- C2) A completely observable observer is existed.
- C3) An asymptotic stable sliding manifold (mode) is existed.

**Proof:** C1) provides a necessary control oriented model structure. Otherwise SMC cannot be used without state space model. The existence of a U-Block model can be achieved from the assignability presented in Section 4. C2) provides a channel to obtain the necessary state variable signal for design. Otherwise SMC cannot be used in practice. The existence can be achieved from assigning characteristic equation of U-Block model. C3) provides stable control functionality. Otherwise simply a designed system is out of control. The existence can be achieved from the following step by step algorithm (Utkin, Guldner, and Shi 2009).

**Step 1:** With reference to U-state space model of (4.11) and (4.12), design a discrete time Sliding Mode (SM). By choosing a matrix  $\tilde{C} \in R^{k \times 1}$  ( $k$  is the dimension of dynamic matrix (4.11) or the order of characteristic equation of U-Block model (4.9), assign a stable manifold

$$s(t) = \tilde{C}x(t) \quad (4.14)$$

where  $s(t)$  is the so called sliding mode (SM).

**Step 2:** Design an equivalent control  $w_{eq}(t)$

$$w_{eq}(t) = -(\tilde{C}B)^{-1} s(t) - (\tilde{C}B)^{-1} (\tilde{C}A - \tilde{C})x(t) \quad (4.15)$$

**Step 3:** Design SMC

$$w(t) = \begin{cases} w_{eq}(t) & \|w_{eq}(t)\| \leq w_0 \\ \frac{w_{eq}(t)}{\|w_{eq}(t)\|} w_0 & \|w_{eq}(t)\| > w_0 \end{cases} \quad (4.16)$$

where  $w_0$  is the controller output bound, that is, makes  $\|w(t)\| \leq w_0$ .

It should be noted that the Sliding Mode Controller  $w(t)$  is a standard form and the stability analysis can be examined by using  $\|s(t+1)\| < \|s(t)\|$  (Utkin, Guldner, and Shi 2009).  $\square$

## 5 Case studies --- design and simulation

Consider a nonlinear dynamic plant described by the following Hammerstein model

$$\begin{aligned} y(t) &= 0.5y(t-1) + x(t-1) + 0.1x(t-2) \\ x(t) &= 1 + u(t) - u^2(t) + 0.2u^3(t) \end{aligned} \quad (5.1)$$

### 5.1 Design the U-Block model

The first step is to convert the Hammerstein model into U-model, with reference to (2.7), it gives

$$y(t) = \lambda_0(t) + \lambda_1(t)u(t-1) + \lambda_2(t)u^2(t-1) + \lambda_3(t)u^3(t-1) \quad (5.2)$$

where

$$\begin{aligned} \lambda_0(t) &= 0.5y(t-1) + 1 + 0.3x(t-2) & \alpha_1(t) &= 1 \\ \alpha_2(t) &= -1 & \alpha_3(t) &= 0.2 \end{aligned} \quad (5.3)$$

As explained in remark 5, U-model based design procedure does not request plant model (no matter line or nonlinear) in the first stage. To assign the obtained U-model into its U-Block model, specify the closed loop characteristic equation with

$$A_c = q^2 - 1.3205q + 0.4966 \quad (5.4)$$

Therefore the closed loop poles are a complex conjugate pair of  $-0.6603 \pm j0.2463$ . This design specification corresponds to a natural frequency of 1 rad/sec and a damping ratio of 0.7. To achieve zero steady state error, specify

$$T = A_c(1) = 1 - 1.3205 + 0.4966 = 0.1761 \quad (5.5)$$

For the polynomials  $R$  and  $S$ , specify

$$\begin{aligned} R &= q^2 + r_1q + r_2 \\ S &= s_0q + s_1 \end{aligned} \quad (5.6)$$

Substitute the specification of (5.5) into Diophantine equation of (4.5), to formulate the coefficients in polynomials  $R$  and  $S$  (Zhu and Guo 2002) with

$$\begin{aligned} r_2 + s_1 &= 0.4966 \\ r_1 + s_0 &= -1.3205 \end{aligned} \quad (5.7)$$

To guarantee the computation convergence of the sequence  $U(t)$ , that is to keep the difference equation with stable dynamic, let  $r_1 = -0.9$   $r_2 = 0.009$ . This assignment corresponds to the characteristic equation of  $U(t)$  as  $(q - 0.89)(q - 0.01) = 0$ . Then the coefficients in polynomial  $S$  can be determined from the Diophantine equation of (6.7)

$$s_0 = -0.4205 \quad s_1 = 0.4876 \quad (5.8)$$

Substitute the coefficients of the polynomials  $R$  and  $S$  into controller of (3.1), gives rise to

$$y_d(t+1) = 0.9(t)y_d(t) - 0.009y_d(t-1) + 0.1761w(t-1) + 0.4205y(t) - 0.4876y(t-1) \quad (5.9)$$

Therefore the controller output  $u(t)$  can be determined from resolving (3.5).

The output response of the designed U-Block model with assigned poles and steady state property is shown in Figure 4.1 and the pole placement controller output is shown in Figure 4.2 (Zhu and Guo, 2009), which is similar to that of a linear system.

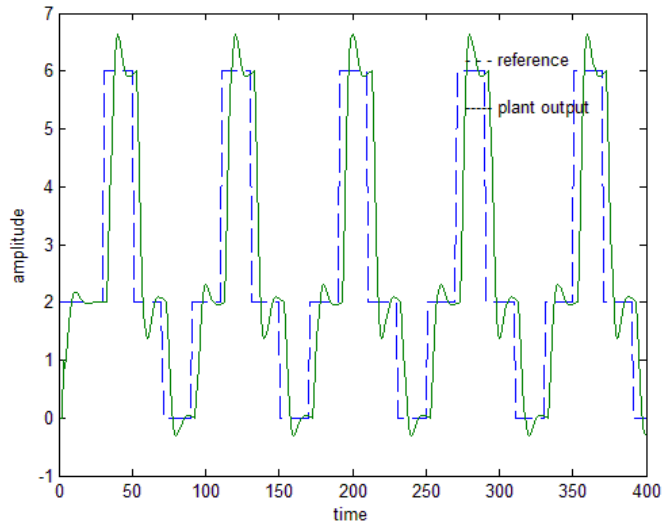


Figure 4.1 Output response of designed U-Block model

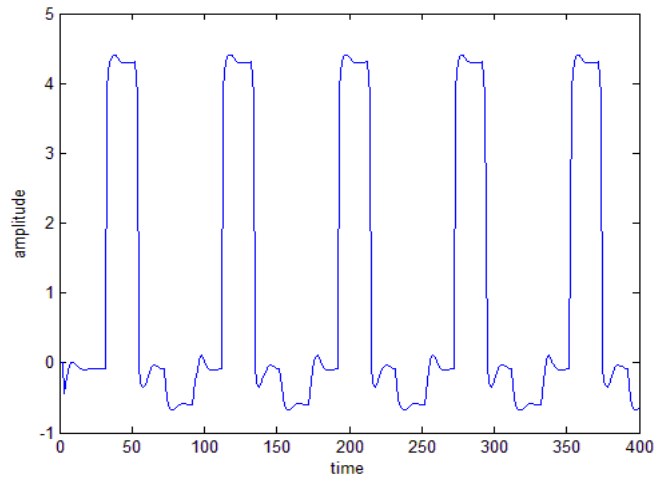


Figure 4.2 Pole placement controller output

### 5.2 Convert the U-Block model into its controllable U-state space model

With reference to the designed U-Block model (5.4) (the numerator of the transfer function) and (5.5) (the denominator of the transfer function), the controllable form in terms of state space model is

$$\begin{aligned} x(t+1) &= Ax(t) + Bw(t) \\ y(t) &= Cx(t) \end{aligned} \quad (5.10)$$

where

$$A = \begin{bmatrix} 0 & 1 \\ -0.4966 & 1.3205 \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad C = [0.1761 \quad 0] \quad (5.11)$$

As the original polynomial plant model only has input and output available for design, a full order state observer (Ogata 2009) with two poles at  $[-0.1 \quad -0.2]$  is designed to accommodate state space type controller, which is given by

$$\hat{x}(t+1) = L_{obs} \hat{x}(t) + M_{obs} u(t) + N_{obs} y(t) \quad (5.12)$$

where  $\hat{x}(t)$  is the estimated state and

$$L_{obs} = \begin{bmatrix} -1.0205 & 1 \\ -1.3676 & 1.3205 \end{bmatrix} \quad M_{obs} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad N_{obs} = \begin{bmatrix} 5.7950 \\ 4.9459 \end{bmatrix} \quad (5.13)$$

### 5.3 Design SMC

**Step 1:** Design a time Sliding Mode (SM). By choosing a matrix  $\tilde{C} \in R^{1 \times 2} = [1 \quad 2]$  to makes the SM be Hurwitz polynomial. Therefore this choice assigns a stable manifold

$$s(t) = \tilde{C}x(t) \quad (5.14)$$

where  $s(t)$  is the so called SM.

**Step 2:** Design an equivalent control  $w_{eq}(t)$

$$w_{eq}(t) = -(\tilde{C}B)^{-1} s(t) - (\tilde{C}B)^{-1} (\tilde{C}A - \tilde{C})x(t) \quad (6.15)$$

**Step 3:** Design SMC

$$w(t) = \begin{cases} w_{eq}(t) & \|w_{eq}(t)\| \leq w_0 \\ \frac{w_{eq}(t)}{\|w_{eq}(t)\|} w_0 & \|w_{eq}(t)\| > w_0 \end{cases} \quad (5.16)$$

where  $w_0 = 0.05$  is the bound to specify the controller output saturation, that is, makes  $\|w(t)\| \leq w_0$ .

The designed performances with initial state  $x_0 = [1 \quad 0.5]^T$  are shown in Figure 5.

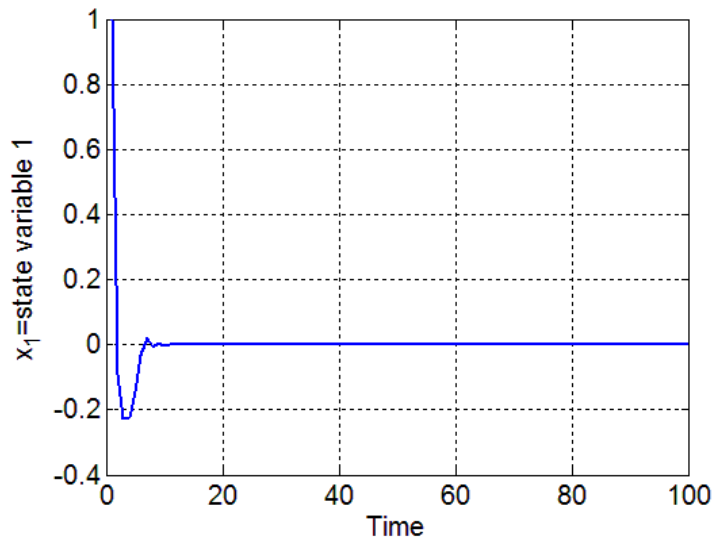


Figure 5.1 Response of state variable 1

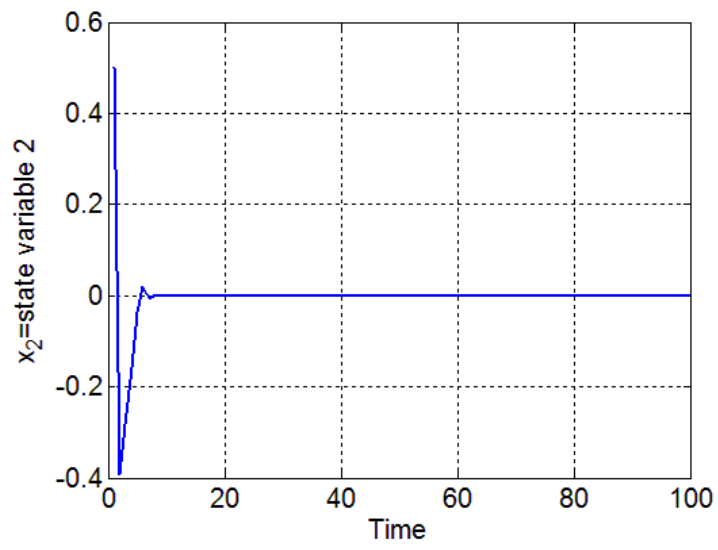


Figure 5.2 Response of state variable 2

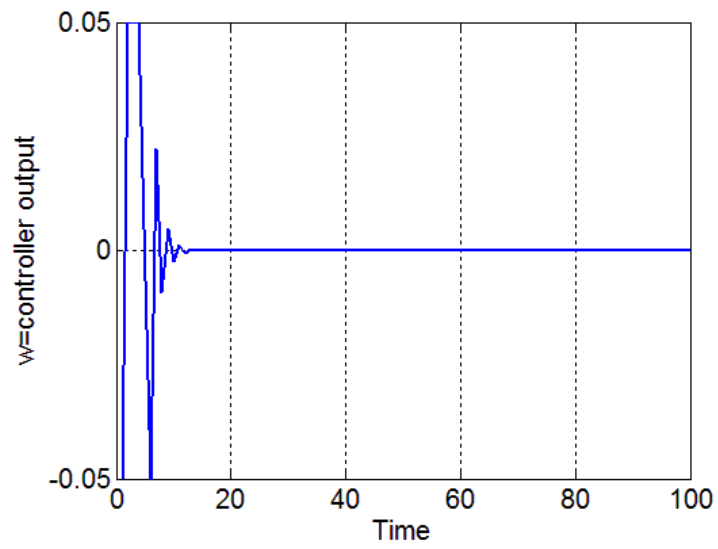


Figure 5.3 Controller output

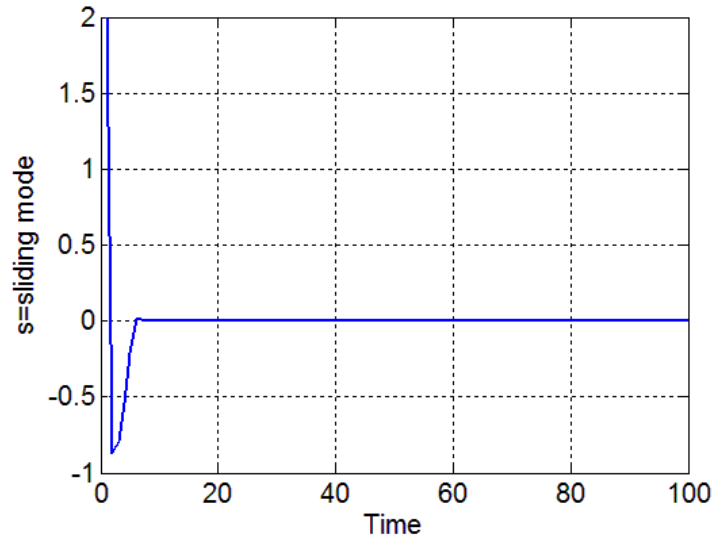


Figure 5.4 Sliding model

#### 5.4 Discussion of the case studies

This is the first study with such proposed framework and procedure, the main purpose of the case study is to demonstrate the feasibility and effectiveness of the developed methodology. For classical comparative demonstrations with the other approaches will be conducted in the future studies. The second motivation in the case study is to give an example for the potential users to expand their studies in this new research domain.

In regarding to section 5.1, the plant has the first order dynamic with nonlinear linear input. To design its U-Block model, a linear closed loop transfer function was specified with second order, decade oscillatory response. The assignability is satisfied to confirm the existence of the U-Block model. An algorithm for resolving Diophantine equation was used to obtain the requested polynomial coefficients linked to the designed poles. To reduce the computational burden, the algorithm could be programmed, for example, Matlab coded. Inspection of Figures 4.1 and 4.2, the U-Block model does behave with the designed response and the root solver gives convergent results. It should be mentioned that the relevant comprehensive studies have been presented in the author's previous work (Zhu and Guo 2002). The example picked here is used as a foundation for the following development.

In regarding to section 5.2, by checking through U-state space model (5.10), it is completely controllable and observable, this provides an assurance to design observer (5.13).

In regarding to section 5.3, again it is not a comprehensive comparative study. SMC has been well matured topic. There is nothing new in this SMC design, therefore it has no need to compare with the other leading SMC results. However its significance is the demonstration of the feasibility in using linear state space approaches to design control systems with nonlinear polynomial plants, which had no other results reported before even though the bottle neck issue has been commonly widely recognised for long time. Figures 5.1 to 5.4 show the designed performance.

In summary, with the simulation demonstration or so called computational experiment tests, the case study supports the following insight.

- 1) It is feasible to combine the linear sliding mode control design with U-model structures to analytically design control systems nonlinear polynomial plant models. In the other terms, the plant is nonlinear (Hammerstein nonlinear polynomial model), the designs are linear approaches (pole placement and sliding mode), and the bridge to connect is U-model (direct re-expression of the original nonlinear model) and U-Block model (designed into liner state space model).
- 2) It has demonstrated the effectiveness of the idea by using simple exiting linear state space control design approaches for nonlinear polynomial plant models, which could be expanded to cover the other strong nonlinear systems (such as those represented by nonlinear rational models (Zhu, Wang, Zhao, Li, and

Billings 2015)) design which is very challenging or unsolved systematically by using exiting control design approaches.

- 3) The algorithms developed could be conveniently converted into online recursive implementation for the control of those complex nonlinear systems in process industry.
- 4) In some sense, a complex problem is split into a simple step by step procedure for solutions.

## 6 Conclusions and future directions

As a natural research progression from U-model based controller design for nonlinear polynomial plants using linear polynomial based approaches, this study establishes a U-Block model framework to facilitate using linear state space based design approaches for nonlinear polynomial models. The concept is novel and design procedure is straightforward and effective without complicated mathematical manipulation. On the other hand, this study reflects the deep understanding and comprehensive skills in dealing with control oriented model structure and control system design, which is a good example to provide easy and general solutions to complicated nonlinear dynamic problems. It is believed that this study will bring forward a new direction in using linear approaches to design nonlinear control systems.

For future directions, some of potential studies associated with the current work are recommended below.

- 1) Even though the bottle neck problem has been resolved in converting nonlinear polynomial into state space model, the SMC selected in the study is just for feasibility test, further comprehensive studies and simulation bench tests should be conducted in the future with reference to many leading research results (Bartolini and Punta 2012, Kalsi, Lian, Hui, and Zak 2010, Moreno and Osorio 2012).
- 2) Within such U-model framework, proper stability analysis methods and stabilisation control should be considered in conjunction with Lyapunov stability analysis theorems (Johansen 2000, Li and Khalil 2012, Zhou, Zheng, and Duan 2011).
- 3) With U-Block realisation, nonlinear polynomial models can be easily converted into state space models, therefore this provides a framework for robust control system design using linear matrix inequality (LMI) and Matlab software. Surely some of the exiting well known results in robust control (Zemouche and Boutayeb 2013, ElBsat and Yaz 2013) will give useful reference for the new development.
- 4) New topics for iterative learning control (ILC), from the author understanding, ILC with linear dynamic plants has been well studied into maturity. However for nonlinear dynamic plants, ILC are still predominantly using linearization approaches (such as Newton algorithms and scant algorithms) in design, which are not accurate and nonlinear plant model structure are not properly accommodated. U-model framework will obviously reduce the complexity in nonlinear type ILC schemes, particularly in the proof of convergent rate and widening of range of learning (Xu and Tan 2003, Tan, Dai, Huang, and Xu 2012, Xu and Xu 2013, Meng, Jia, Du, and Zhang 2014).

## 7 Acknowledgements

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; Development of key technologies project of Qingdao Economic and Technological Development Zone under Grant 2011-2-52; and Taishan Scholar Construction Engineering Special Funding, Shandong, China. 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

- Astrom, K.J. and Wittenmark, B., Adaptive control (2<sup>nd</sup> ed), Addison Wesley, Reading, 1995.
- Bartolini, G. and Punta, E., Sliding mode output-feedback stabilization of uncertain nonlinear nonaffine systems, *Automatica*, Vol. 48, No. 12, pp. 3106-3113, 2012.
- Basin, M. and Calderon-Alvarez, D., Sliding mode regulator as solution to optimal control problem for nonlinear polynomial systems, *Journal of the Franklin Institute* Vol. 347, pp. 910-922, 2010.
- Basin, M. and Rodriguez-Ramirez, P.C., Sliding mode filter design for nonlinear polynomial systems with unmeasured states, *Information Sciences*, Vol. 204, pp. 82-91, 2012.
- Billings, S.A., *Nonlinear System Identification: NARMAX Methods in the Time, Frequency, and Spatio-Temporal Domains*, Wiley, John & Sons, Chichester, West Sussex, 2013.
- Chong, E.K.P. and Zak, S.H., *An Introduction to Optimization*, (4<sup>th</sup> ed), Wiley, Purdue University, 2013.
- Çimen, T., Systematic and effective design of nonlinear feedback controllers via the state-dependent Riccati equation (SDRE) method, *Annual Reviews in Control*, Vol. 34, No. 1, pp. 32-51, 2010.
- 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. Eng., Part I: Journal of Systems and Control Engineering*, V 226, pp. 27-42, 2012.
- ElBsat, M.N. and Yaz, E.E., Robust and resilient finite-time bounded control of discrete-time uncertain nonlinear systems, *Automatica*, Vol. 49, No. 7, pp. 2292 - 2296, 2013.
- Isidori, A., *Nonlinear control systems* (3rd ed.), Berlin: Springer-Verlag, 1995.
- Johansen, T.A., Computation of Lyapunov functions for smooth nonlinear systems using convex optimization, *Automatica*, Vol.36, No. 11, pp. 1617-1626, 2000.
- Kalsi, K., Lian, J.M., Hui, S., and Zak, S.H., Sliding-mode observers for systems with unknown inputs: A high-gain approach, *Automatica* Vol. 46, No. 2, pp. 347-353, 2010.
- Li, R.R. and Khalil, H.K., Nonlinear output regulation with adaptive conditional servocompensator, *Automatica* Vol. 48, No. 10, pp. 2550 - 2559, 2012.
- Maciejowski, J.M., *Multivariable feedback design*, Addison Wesley, Reading, 1994.
- Meng, D.Y., Jia, Y.M., Du, J.P., and Zhang, J., On iterative learning algorithms for the formation control of nonlinear multi-agent systems, *Automatica*, Vol. 50, No. 1, pp. 291-295, 2014.
- Moreno, J.A., and Osorio, M., Strict Lyapunov Functions for the Super-Twisting Algorithm, *Automatic Control*, *IEEE Transactions*, Vol. 57 No. 4, pp. 1035-1040, 2012.
- 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*, Vol. 219, No. 6, pp. 449-458, 2005.
- Nash, S., A survey of truncated-Newton methods, *Journal of Computational and Applied Mathematics*, Vol. 124, pp. 45-59, 2000.
- Ogata, K., *Modern Control Engineering* (5<sup>th</sup> ed), Prentice-Hall, 2009.
- Slotine, J.E. and Li, W., *Applied nonlinear control*, Prentice Hall, London, 1991.

- Tan Y., Dai, H.H., Huang, D.Q., and Xu, J.X., Unified iterative learning control schemes for nonlinear dynamic systems with nonlinear input uncertainties, *Automatica*, Vol. 48, No.12, pp. 3173-3182, 2012.
- Taylor, C.J., Chotai, A. and Young, P.C., Non-linear control by input-output state variable feedback pole assignment, *International Journal of Control*, Vol.82, No. 6, pp. 1029 — 1044, 2009.
- Utkin, V., Guldner, J., and Shi. J., *Sliding Mode Control in Electro-Mechanical Systems* (2nd ed), New York: Taylor & Francis, 2009.
- Xu, J.X. and Tan, Y., *Linear and nonlinear iterative learning control*, Lecture notes in control and information sciences, Springer, Berlin, 2003.
- Xu, J.X. and Xu, J., State-Constrained Iterative Learning Control for a Class Of MIMO Systems, *Automatic Control, IEEE Transactions*, Vol. 58, No. 5, pp. 1322 – 1327, 2013.
- Young, P.C., Behzadi, M.A., Wang, C.L., and Chotai, A., Direct digital and adaptive control by input-output, state variable feedback pole assignment, *International Journal of Control*, Vol. 46, pp. 1867–1881, 1987.
- Young, P.C., Stochastic, dynamic modelling and signal processing: time variable and state dependent parameter estimation, in *Nonlinear and Nonstationary Signal Processing*, ed. Fitzgerald, W.J., Cambridge: Cambridge University Press, 2000.
- Zemouche, A., and Boutayeb, M., On LMI conditions to design observers for Lipschitz nonlinear systems, *Automatica* Vol. 49, No. 2, pp.585 – 591, 2013.
- Zhou, B., Zheng, W.X, and Duan, G.R., Stability and stabilization of discrete-time periodic linear systems with actuator saturation, *Automatica*, Vol. 47, No. 8, pp. 1813-1820, 2011.
- Zhu, Q.M., *Identification and Control of nonlinear systems*, PhD thesis, University of warwick, UK, 1989.
- 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., Wang, Y.J., Zhao, D.Y., Li, S.Y., and Billings, S.A., Review of rational (total) nonlinear dynamic system modelling, identification and control, *Int. J. of Systems Science*, 2015 (in press).