

U-model based predictive control for nonlinear processes with input delay

Xinpeng Geng^a, Quanmin Zhu^{b,*}, Tao Liu^{a,*}, Jing Na^c

a Institute of Advanced Control Technology, Dalian University of Technology, Dalian, 116024, P. R. China

b Department of Engineering Design and Mathematics, University of the West of England, Frenchy Campus
Coldharbour Lane, Bristol, BS16 1QY, UK

c Faculty of Mechanical & Electrical Engineering, Kunming University of Science & Technology, Kunming,
650500, P.R. China

* Corresponding authors. Tel: +86-411-84706465; Fax: +86-411-84706706

Emails: xpgengdlut@hotmail.com (X. Geng); quan.zhu@uwe.ac.uk (Q. Zhu); liurouter@ieee.org (T. Liu);
najing25@163.com (J. Na)

Abstract: In this paper, a general control scheme is proposed for nonlinear dynamic processes with input delay described by different models, including polynomial models, state-space models, nonlinear autoregressive moving average with eXogenous inputs (NARMAX) models, Hammerstein or Wiener type models. To tackle the input delay and nonlinear dynamics involved with the control system design, it integrates the classical Smith predictor and a U-model based controller into a U-model based predictive control scheme, which gives a general solution of two-degree-of-freedom (2DOF) control for the set-point tracking and disturbance rejection, respectively. Both controllers are analytically designed by proposing the desired transfer functions for the above objectives in terms of a linear system expression with the U-model, and therefor are independent of the process model for implementation. Meanwhile, the control system robust stability is analyzed in the presence of process uncertainties. To demonstrate the control performance and advantage, three examples from the literature are conducted with a user-friendly step by step procedure for the ease of understanding by readers.

Keywords: nonlinear dynamic control, input delay, Smith Predictor, U-model, U-Smith predictor controller, robust stability

1. Introduction

Time delay appears in various industrial operations associated with transmissions of material or energy between interconnected systems, data transmission in communication systems and networked control systems. The presence of time delay may lead to a sluggish response, limit the achievable control performance, or even provoke instability of the closed-loop systems. The main challenge for controlling such a process is to avoid overshoot in tracking a desired set-point profile and to accommodate stability and robustness against process uncertainties [1]. In particular, the input delay is a notorious barrier for the control of industrial processes with nonlinear dynamics [2, 3].

For a general review of the existing references contributing to nonlinear systems with input delay, the research development is divided in terms of continuous-time domain and discrete-time domain. In continuous-time domain, backstepping based designs were developed to deal with time-invariant input delay for control implementation [4-6]. The backstepping strategy was further extended to stabilize nonlinear systems with time-varying input delay [7], by tuning the controllers with respect to the output bound. In contrast, the finite spectrum assignment (FSA) approach was explored for retarded nonlinear systems by transforming such a nonlinear system into a delay-free form for control design [8]. A truncated predictor feedback control design based on state estimation or delay-free output estimation were developed for nonlinear processes with input delay [9, 10]. By using a high-gain output predictor, a feedback linearization design was proposed to maintain asymptotic stability of the closed-loop control system for a nonlinear process with time-varying input delay and output delay [11]. By comparison, approximate predictors and high-gain observers were adopted for estimating the delay-free state and output for control design by means of numerical computation on the ordinary differential equations of nonlinear systems [12]. To reduce the high gain of such an observer, a chain of observation algorithms was proposed to reconstruct the system state and output at different delayed time instants [13]. In contrast, a nonlinear filtered Smith predictor structure [14] was proposed for predicting the delay-free output of a nonlinear process, based on the Hammerstein type or Volterra model for describing the system dynamics. For sampled control systems implemented in

discrete-time domain, the approximated implementation of continuous-time control design may affect or even destroy the system stability [1, 15]. Only a few papers, however, presented discrete-time control designs for nonlinear systems with input delay. Based on the delay-free output prediction, an adaptive neural network control scheme was presented to approximate the desired control input for implementation [16]. By representing a nonlinear time-delay system into a T-S fuzzy system comprised of linear delay difference inclusions, a nonlinear model predictive control (MPC) method [17] was proposed to improve system performance for disturbance rejection. A dynamic output feedback linearization control algorithm was developed by using the additive nonlinear autoregressive moving average with eXogenous inputs (NARMAX) models established by training a neural network with a specific connectivity structure [18]. Note that the robust stability under process uncertainties was left open in the above references. To deal with input delay uncertainty, a predictor based robust control design was given which could allow the input delay to be varying in a range [19]. Another predictor based control design was proposed to deal with state-dependent input delay [20]. To improve disturbance rejection performance in the presence of input delay uncertainty, a modified active disturbance rejection control (ADRC) design based on the extended state observer (ESO) was proposed [21], which could be applied to linear or nonlinear systems based on a low-order model description of the fundamental dynamics of such a system. This approach was recently extended to systems with longer input delay, by using a filtered SP in combination with ESO [22].

It should be noted that most of the existing references for nonlinear systems included the aforementioned almost gave up the Smith Predictor (SP) based control structure [23] that had been successfully used for linear systems with input delay [24]-[30], probably due to the difficulty in dealing with the related nonlinear models. It was noted [31] that SP could not be directly used for nonlinear control design because nonlinear systems generally could not be expressed by linear transfer function models. Besides, it had been recognized that the SP could not be directly used in terms of a state-space description of the system under control.

To deal with the above issues, a universal framework is proposed in this paper for controlling nonlinear processes with input delay described by different models like polynomial models,

state-space models, NARMAX models, Hammerstein or Wiener type models, by extending the U-model enabled design for delay-free nonlinear systems [32]. This paper will concentrate on the control design in the presence of time-invariant input delay, which is the foundation for solving the next step challenge of time-varying input delay. By integrating the SP with the delay-free U-model based control structure, a two-degree-of-freedom (2DOF) control scheme with output prediction is proposed for application to various nonlinear systems, based on a general U-model representation of different plant models for control design. By specifying the desired transfer functions for the set-point tracking and disturbance rejection, both controllers are analytically derived, respectively. They can be separately tuned for control performance optimization. A notable advantage of the proposed control design is that both controllers could be tuned relatively independent of the plant model or its variation, therefore facilitating practical applications. Moreover, the system performance could be monotonically tuned by the single adjustable parameter of each controller. For clarity, the paper is organised as follows. Section 2 presents the U-model representations of different classical models of nonlinear processes with input delay, which lays a foundation for the general framework of U-model based control design. Section 3 proposes a general U-model based control system design with a step by step implementation procedure. Section 4 presents robust stability analysis of the proposed control scheme. Section 5 shows three illustrative examples to demonstrate the performance of the proposed control scheme. Section 6 draws some conclusions along with potential research topics for study in the future.

Notations: Throughout the study, a series of conventional notations are used, R , R^n , and $R^{n \times m}$ for sets of real number, n -dimension real factor, and $n \times m$ real matrix. Denote by $t \in 1, 2, \dots$ for sampling time instance.

2. U-model description of nonlinear processes with input delay

Consider a general single-input single-output (SISO) nonlinear process with input delay described by the discrete-time domain state-space model,

$$\begin{aligned} X(t+1) &= F(X(t), u(t-\tau)) \\ y(t) &= h(X(t)) \end{aligned} \tag{1}$$

where $X \in R^n$ is the state vector, $u \in R$ is the control input, $\tau > 0$ is time delay in the control input, $y \in R$ is the system output. $F \in R^n$ is a smooth vector function describing the model dynamics and $h \in R$ is a smooth function relating the system states to the output. Throughout the study, assume the system relative degree r equals to the system order n and has stable zero dynamics (i.e., the system model has a stable inverse), and the state vector X is available by measurement or observer.

Equivalently, the corresponding input-output polynomial model of (1) is written by

$$y(t) = f(y(t-1), \dots, y(t-n), u(t-1-\tau), \dots, u(t-n-\tau)) \quad (2)$$

where $y(t) \in R$ and $u(t-1) \in R$ are the output and input (also known as the controller output in control system design) signals of the plant, respectively, at the discrete-time instance $t \in 1, 2, \dots$. $f \in R$ is a smooth linear or nonlinear function.

For delay-free nonlinear systems, it has been explored that the U-model based control strategy could establish a universal framework to design a control system based on the plant polynomial or state-space model [32, 33]. Figure 1 shows the U-model based control structure, where G_p denotes the plant, G_p^{-1} (U-model) an inverse of the plant model represented by the U-model, G_c the closed-loop controller. The U-model is defined as a polynomial of $u(t)$ with time-varying coefficients. With the control oriented model structure, the U-model transforms a smooth (polynomial) nonlinear model of the plant into a class of polynomials, so as to make a dynamic inversion of the plant for deriving the control implementation, which is resolved by finding one of the roots of the U-model. Given a state-space model of the plant, the classical backstepping algorithm could be expanded to recursively resolve the multi-layer roots. In this way, the U-model based control scheme is capable of radically relieving the dependence on the plant model that has been the foundation of classical control system designs in the literature. The role of plant model is therefore reduced as a reference for converting to U-model and determining the next step controller output. Note that the U-model based control design takes a hypothesis on the feasibility using linear methodologies to directly design nonlinear control systems within a universal framework.

To extend the U-model based control strategy for nonlinear processes with input delay, the

first step is to transform the plant model of polynomial, state-space, or else with time delay into the U-model, as detailed in the following subsections.

2.1 Single-layer U-model in polynomial

2.1.1 Polynomial U-model set

The general polynomial model in (2) can be alternatively written in terms of a parametric form,

$$\begin{aligned} y(t) &= f(y(t-1), \dots, y(t-n), u(t-1-\tau), \dots, u(t-n-\tau), \Theta) \\ &= f(p(t), \Theta) \end{aligned} \quad (3)$$

where $\Theta = [\theta_0 \ \dots \ \theta_L] \in R^{L+1}$ is the associated parameter vector, L is the number of regression terms, $p \in R^{L+1}$ is a regression term composed of the past inputs and outputs.

Definition [34]: Assign a triplet (X, f, h) , X is an irreducible real affine variety, (f, h) are mapping functions. A discrete-time process model Σ with delayed input space $U(t-\tau) \in R^m, t, \tau \in \mathbb{Z}^+$ and output $Y(t) \in R^r$ is defined as polynomial/rational, while the functions $f = \{f_\alpha | \alpha \in U\}$ and $h: X \rightarrow R^r$ both on X are mappings from input space to state space and from state space to output space polynomial/rational, respectively. That is, for polynomial systems, $h_i \in A$ for all $i=1, \dots, r$ where A is the algebra of all polynomials on the variety X , and for rational systems, $h_i \in Q$ for all $i=1, \dots, r$ where Q is the algebra of all rational functions on the variety X .

Assume a mapping $\varphi: R^{L+1} \rightarrow R^{M+1}$ and its inverse φ^{-1} exist, that is $f(p_i, \theta_i) \xrightarrow{\varphi} f(u^j, \lambda_j)$, it defines the corresponding U-model structure of (3) as an input oriented polynomial,

$$y(t) = \sum_{j=0}^M \lambda_j(t) u^j(t-1-\tau) \quad (4)$$

where M is the degree of model input, 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 vector Θ .

Property 1: For the above manifolds $f_p(p_i, \theta_i)$, $f_u(u^j, \lambda_j)$, the differentiable mapping $f(p_i, \theta_i) \xrightarrow{\varphi} f(u^j, \lambda_j)$ is diffeomorphism, which is bijection and its inverse $f(u^j, \lambda_j) \xrightarrow{\varphi^{-1}} f(p_i, \theta_i)$ is differentiable as well [32].

For illustration, consider a nonlinear process described by a polynomial model,

$$y(t) = 0.5y(t-1)y(t-2) + 0.7u(t-3-\tau)u(t-1-\tau) + 0.9y(t-2)u^2(t-1-\tau) - 0.1u^3(t-1-\tau) \quad (5)$$

With the above U mapping, the corresponding U-model is given by

$$y(t) = \lambda_0(t) + \lambda_1(t)u(t-1-\tau) + \lambda_2(t)u^2(t-1-\tau) + \lambda_3(t)u^3(t-1-\tau) \quad (6)$$

where $\lambda_0(t) = 0.5y(t-1)y(t-2)$, $\lambda_1(t) = 0.7u(t-3-\tau)$, $\lambda_2(t) = 0.9y(t-2)$, and $\lambda_3(t) = -0.1$.

Clearly, $\lambda_j(t)$ is a time-varying function, absorbing the past variables and parameters of the original polynomial in (5) associated with $u^j(t-1-\tau)$.

Remark 1: Concerning the U-model in (4), an input oriented prototype, there is no change of model properties compared with the classical polynomial models in representation of (3). Therefore, the U-model and classical polynomial models are equivalent to each other. However, the U-model is linear-like with time-varying parameters $\lambda(t)$ as shown in (4) and therefore, is convenient for applying the developed linear control methods based on real-time estimation of $\lambda(t)$, compared to the nonlinear polynomial model in (3) used for exploring specific nonlinear control designs in the literature (e.g. [4-8, 10-12]). Hence, the U-model can bridge various types of polynomial models for nonlinear systems with the developed linear control system designs within a universal framework. \diamond

2.1.2 Rational U-model set

This model set is defined as

$$y(t)D_p(t) = N_p(t) \quad (7)$$

where $D_p(t)$ and $N_p(t)$ are two U-polynomials, as expressed in the parametric form of

$$y(t) \sum_{i=0}^L \gamma_i(t) u^i(t-1-\tau) = \sum_{j=0}^M \lambda_j(t) u^j(t-1-\tau) \quad (8)$$

This is a representation of the classical rational model set, which is expressed as a ratio of the numerator to denominator polynomials. For illustration, a rational model

$$y(t) = y(t-1) \frac{1 + 0.8u(t-1-\tau)}{1 + 0.5u^2(t-1-\tau)} \quad \text{may be converted into a U-model}$$

$$y(t)[\gamma_0 + \gamma_2(t)u^2(t-1-\tau)] = \lambda_0(t) + \lambda_1(t)u(t-1-\tau) \quad \text{where} \quad \gamma_0(t) = 1 \quad , \quad \gamma_2(t) = 0.5 \quad ,$$

$$\lambda_0(t) = y(t-1) \quad , \quad \lambda_1(t) = 0.8y(t-1).$$

2.1.3 Extended U-model set

This model set is defined as

$$y(k)f_b(u(t-1-\tau)) = f_a(u(t-1-\tau)) \quad (9)$$

where $f_b(u(t-1-\tau)) \in R$ and $f_a(u(t-1-\tau)) \in R$ are smooth functions that may be generally expressed by

$$\begin{aligned} f_b(u(t-1-\tau)) &= \sum_j f_{bj}(u(t-1-\tau)) \\ f_a(u(t-1-\tau)) &= \sum_j f_{aj}(u(t-1-\tau)) \end{aligned} \quad (10)$$

For instance, consider a nonlinear process, $y(t) = \frac{y(t-1)\sin(u(t-1-\tau))}{1+0.5\cos^2(u(t-1-\tau))}$, using the extended U-model defined by (9), there follows $f_a(u(t-1-\tau)) = \lambda_1(t)\sin(u(t-1-\tau))$, $f_b(u(t-1-\tau)) = \gamma_0(t) + \gamma_1(t)\cos^2(u(t-1-\tau))$, where $\gamma_0(t) = 1$, $\gamma_1(t) = 0.5$, $\lambda_1(t) = y(t-1)$.

2.2 Multi-layer U-model in state space

With reference to the U model in (4), the state-space model in (1) can be converted into a multi-layer U model as

$$\begin{cases} x_1(t+1) = \sum_{j=0}^{M_1} \lambda_{1j}(t)x_2^j(t) \\ \vdots \\ x_{n-1}(t+1) = \sum_{j=0}^{M_{n-1}} \lambda_{n-1j}(t)x_n^j(t) \\ x_n(t+1) = \sum_{j=0}^{M_n} \lambda_{nj}(t)u^j(t-\tau) \\ y(t) = h(X(t)) \end{cases} \quad (11)$$

where, for each line, M_j is the degree of next state variable $x_{j+1}(t)$, the time-varying parameter vector $\tilde{\lambda}_i(t) = [\lambda_{j0}(t) \quad \cdots \quad \lambda_{jM_j}(t)] \in R^{M_j+1}$, $i = 1 \cdots n$ is a function of the other state variables. In the penultimate line, M_n is the degree of the model input (controller output) $u(t-1)$, the

time-varying parameter vector $\tilde{\lambda}_n(t) = [\lambda_{n0}(t) \ \cdots \ \lambda_{nM_n}(t)] \in R^{M_n+1}$ is a function of all system states.

Remark 2: In the multi-layer U-model shown in (11), the controller output $u(t-\tau)$ can be derived by using a backstepping routine as long as $x_1(t+1)$ is determined. \diamond

2.3 U-model representations for Hammerstein and Wiener models

Considering that Hammerstein type models, Wiener type models, and NAMAX models have also been widely used for describing various nonlinear system dynamics, the corresponding U-model representations are given below for the convenience of control design.

Hammerstein type models

The Hammerstein type model is a cascade structure of a nonlinear static input block and a linear dynamic block. For describing a nonlinear process with input delay, it is in the form of

$$\begin{aligned} X_H(t-1) &= f_H(u(t-1-\tau)) \\ y(t) &= \sum_{i=0}^m b_i(t) X_H(t-1-i) - \sum_{i=1}^n a_i(t) y(t-i) \end{aligned} \quad (12)$$

where $y(t)$ and $u(t-1-\tau)$ are the output and input of the plant at the discrete-time instant $t=1,2,\dots$, respectively, $X_H(t)$ is the output of the nonlinear input block, and $f_H(\cdot)$ is a nonlinear function of the input $u(t-1-\tau)$.

The corresponding equivalent U-model can be expressed as

$$y(t) = \lambda_0(t) + \lambda_1(t) X_H(t-1) = \lambda_0(t) + \lambda_1(t) f_H(u(t-1-\tau)) \quad (13)$$

where

$$\lambda_0(t) = \sum_{i=1}^m b_i(t) X_H(t-1-i) - \sum_{i=1}^n a_i(t) y(t-i), \quad \lambda_1(t) = b_0 \quad (14)$$

Wiener type models

The Wiener type model is a cascade structure of a linear dynamic block and a nonlinear static output block. For a nonlinear process with input delay, it has the form of

$$\begin{aligned} X_W(t) &= \sum_{i=0}^m b_i(t) u(t-1-\tau-i) - \sum_{i=1}^n a_i(t) X_W(t-i) \\ y(t) &= f_W(X_W(t)) \end{aligned} \quad (15)$$

where $y(t)$ and $u(t-1-\tau)$ denote the output and input of the plant at the discrete-time instant $t = 1, 2, \dots$, respectively, $X_w(t)$ is the output of the linear dynamics, and $f_w(\cdot)$ is a nonlinear function of $X_w(t)$.

The corresponding equivalent U-model can be expressed as

$$\begin{aligned} X_w(t) &= \lambda_0(t) + \lambda_1 u(t-1-\tau) \\ y(t) &= f_w(X_w(t)) \end{aligned} \quad (16)$$

where

$$\lambda_0(t) = \sum_{i=1}^m b_i(t) X_H(t-1-i) - \sum_{i=1}^n a_i(t) y(t-i), \quad \lambda_1(t) = b_0 \quad (17)$$

NARMAX models

Generally, an NARMAX model has the form,

$$y(t) = \sum_{l=0}^L p_l(t) \theta_l \quad (18)$$

where the regression terms $p_l(t)$ are the products of past inputs and outputs such as $u(t-1-\tau)y(t-3)$, $u(t-1-\tau)u(t-2-\tau)$, $y^2(t-1)$, and θ_l ($l = 0, 1, \dots, L$) are the associated parameters.

The corresponding equivalent U-model can be straightforwardly expressed as

$$y(t) = \sum_{j=0}^M \lambda_j(t) u^j(t-1-\tau) \quad (19)$$

where $\lambda_j(t)$ can be viewed as a time-varying parameter absorbing $p_l(t)$ and θ_l associated with $u^j(t-1-\tau)$ in (18).

3. U-model based predictive control design

3.1 Framework of U-model based control design

Without loss of generality, consider a single input ($u \in R$) and single output ($y \in R$) linear feedback and delay-free control system structured with a triplet

$$\Sigma = (F_{\text{fbc}} \quad G_{\text{c1}} \quad G_{\text{ip}}) \quad (20)$$

where the linear invariant controller $G_{\text{c1}} : y \rightarrow u$ and the constant unit $G_{\text{ip}} = G_{\text{p}}^{-1} G_{\text{p}} = 1 : u \rightarrow y$ are both within a linear feedback control framework, F_{fbc} .

Denote by \hat{G}_p the process model, which is in the form of

$$\hat{G}_p: \begin{cases} X(t+1) = F(X(t), u(t)) \\ y(t) = h(X(t)) \end{cases} \quad (21)$$

It is generally assumed that \hat{G}_p and its inverse \hat{G}_p^{-1} implemented by u shown in Figure 1 are diffeomorphism and globally uniform Lipschitz in R^n , that is

$$\|\hat{G}_p(x_1) - \hat{G}_p(x_2)\| \leq \gamma_1 \|x_1 - x_2\|, \quad \forall x_1, x_2 \in R^n \quad (22)$$

$$\|\hat{G}_p^{-1}(x_1) - \hat{G}_p^{-1}(x_2)\| \leq \gamma_2 \|x_1 - x_2\|, \quad \forall x_1, x_2 \in R^n \quad (23)$$

where γ_1, γ_2 are Lipschitz coefficients.

From (20), two separate designs are in parallel: 1) the linear invariant controller G_{c1} is designed with $\sum = (F_{\text{libc}} \quad G_{c1} \quad 1)$, given a linear feedback structure F_{libc} ; 2) determine u to realize $G_{ip} = \hat{G}_p^{-1}G_p = 1$. Note that if the process model is modified, re-design of the control system can be simply conducted by re-doing the model inverse \hat{G}_p^{-1} , with no change on G_{c1} (corresponding to G_c in Figure 1).

Hence, the structured control system in (20) is generally applicable to all the classical linear process models by letting $G_{ip} = \hat{G}_p^{-1}G_p = 1$, if the process model inverse \hat{G}_p^{-1} exists.

Remark 3: The U-model based control design is independent of the process model, owing to the above separate designs. Hence, there is no need to change the controller for implementation when the process model is modified, compared to the existing nonlinear control methods dependent on the process model (e.g. [4, 5, 8, 10, 11, 15, 20]). The U-model based control takes effort in dynamic inversion as well as the existing control methodologies such as the internal model control (IMC) [35]. This is generally conducted by defining an inverse function Ψ in the literature. Note that it is only a function of the process model, i.e., $\Psi(\hat{G}_p)$, in the U-model based control design, compared to those of many existing control methods which are involved with both G_{c1} and \hat{G}_p , i.e., $\Psi(G_{c1}, \hat{G}_p)$. Hence, the computation effort could be significantly reduced for the dynamic inversion, in particular for nonlinear control systems.

3.2 U-model based predictive control design for input delay systems

As explained in Section 3.1, the U-model can be integrated with many existing control system frameworks. To cope with the input delay, the well-known Smith predictor may be adopted in combination with the delay-free U-model control structure shown in Figure 1, which is shown in Figure 2. Considering that $\hat{G}_{p0}^{-1}\hat{G}_{p0} = 1$, the combined control structure in Figure 2 is further simplified, while a set-point filter is introduced for tuning the set-point tracking performance. Hence, the proposed U-model based predictive control scheme is a two-degree-of-freedom (2DOF) control structure, as shown in Figure 3, where \hat{G}_{p0}^{-1} denotes the inverse of a delay-free U-model (\hat{G}_{p0}) of the plant G_p as shown in (4), G_r is a desired transfer function from the set-point to the system output, which may be regarded as the set-point tracking controller, the desired output without delay is denoted by y_r , G_c is a closed-loop controller which is used to eliminate the output error arising from process uncertainties and load disturbance denoted by d . Hence, the nonlinear dynamics involved in the system can be virtually regarded as a linear plant, owing to that the proposed U-model based control could eliminate the effect of nonlinearities in the system. Moreover, both G_r and G_c can be designed in terms of the desired transfer functions for the set-point tracking and load disturbance rejection, respectively, rather than depending on the nonlinear process model.

The control objective is, for a desired output trajectory $y_r(t)$, to find a manipulated variable $u(t)$ to drive the process output $y(t)$ to track the delayed trajectory $y_r(t-\tau)$ with an acceptable performance (such as transient response and steady-state error), while all the inputs and outputs of the control system are bounded in the permitted ranges for system operation.

There are three steps for the U-model based predictive control design as below.

Step 1: Design the set-point tracking controller G_r by proposing the desired transfer function, and accordingly, determine the desired output trajectory $y_r(t)$.

Step 2: Design the closed-loop controller G_c to eliminate the error between the desired output and the process output, by proposing the desired closed-loop transfer function for disturbance rejection.

Step 3: Derive the process inversion by \hat{G}_{p0}^{-1} (assuming the inverse of the delay-free plant

model exists) in terms of the U-model, and then resolve a proper root of the U-model expression as the manipulated variable $u(t-1)$ for implementation.

For Step 1, to realize fast and smooth set-point tracking without overshoot in the nominal case of perfect model match with the plant (i.e., $G_p = \hat{G}_p = \hat{G}_{p0}z^{-\tau}$, which means there is open-loop control for the set-point tracking as can be seen from Figure 3), the desired transfer function for the set-point tracking, which is exactly from r to y_r in Figure 3, is therefore proposed via the controller $G_r(z)$ for discrete-time implementation as

$$G_r(z) = \frac{(1-\lambda_r)^{n_r} z^{n_r}}{(z-\lambda_r)^{n_r}} \quad (24)$$

where λ_r is a tuning parameter satisfying $|\lambda_r| < 1$ in the z -plane, and n_r is a specified order of the transfer function that may be chosen with respect to the measurement noise level in engineering application [2]. When $\lambda_r \rightarrow 0$, there is $G_r(z) \rightarrow 1$, which means the transfer function recovers the ideal form of one for the set-point tracking.

It can be seen from (24) that tuning a smaller value of λ_r could obtain a faster set-point tracking speed with a larger control effort and vice versa, similar to the internal model control (IMC) design for set-point tracking [35]. It is suggested to initially take $\lambda_r \in [0.5, 0.9]$, and then by monotonically increasing or decreasing it, a desirable trade-off between the tracking performance and the control effort (along with its robustness against process uncertainties) could be achieved.

For Step 2, consider the closed-loop control structure outlined by the dash-line box in Figure 3, it can be easily derived in the nominal case that

$$\frac{y}{d} = \frac{1}{1+G_c z^{-\tau}} = 1 - T_d \quad (25)$$

$$\frac{\hat{y}}{d} = T_d = \frac{G_c z^{-\tau}}{1+G_c z^{-\tau}} \quad (26)$$

Ideally, it is desired that $T_d = 1$, such that the controller G_c could drive the process to yield an equivalent output \hat{y} to counteract d when it is detected by the output measurement error denoted by e in Figure 3. Consider practical constraints and time delay for control

implementation, the desired transfer function for disturbance rejection is proposed as

$$T_d = \frac{(1 - \lambda_c)z}{z - \lambda_c} z^{-\tau} \quad (27)$$

where λ_c is a tuning parameter. Note that when $\lambda_c \rightarrow 0$, T_d recovers the ideal form of $z^{-\tau}$.

Correspondingly, the closed-loop controller can be inversely derived from (26) as

$$G_c = \frac{T_d z^\tau}{1 - T_d} = \frac{(1 - \lambda_c)z}{z(1 - z^{-\tau}) + \lambda_c(z^{-\tau+1} - 1)} \quad (28)$$

It can be easily verified from Figure 3 that

$$\begin{aligned} \lim_{z \rightarrow 1} e(z) &= \lim_{z \rightarrow 1} \frac{1}{G_c \hat{G}_{p0}^{-1} G_p} [y_r(z) \hat{G}_{p0}^{-1} G_p - y(z)] \\ &= \lim_{z \rightarrow 1} \frac{z(1 - z^{-\tau}) + \lambda_c(z^{-\tau+1} - 1)}{(1 - \lambda_c)z \hat{G}_{p0}^{-1} G_p} [y_r(z) \hat{G}_{p0}^{-1} G_p - y(z)] \\ &= 0 \end{aligned} \quad (29)$$

Hence, no steady-state output error could be ensured by the above controller design, in the presence of process uncertainties or load disturbance.

Remark 4: *The closed-loop controller design in (28) is generally applicable to deal with load disturbance without integral or unstable property, such as a constant, asymptotically stable, or sinusoidal type disturbance. For a ramp type or unstable disturbance, the desired form of T_d need to be specifically designed with constraints according to the transfer function approach developed in [2], so that the closed-loop controller could be inversely determined to ensure no steady-state output error.* \diamond

For Step 3, in the nominal case, there stands $e(z) = 0$ for $z \in (0, 1)$. Based on the designed G_r and the desired output trajectory $y_r(t)$, the manipulated variable $u(t-1)$ can be determined from the delay-free U-model (similar to the input delayed U-model in (4)) by letting

$$y_r(t) = \sum_{j=0}^M \lambda_j(t) u^j(t-1) \quad (30)$$

Then a proper solution of $u(t-1)$ could be found from the roots of

$$y_r(t) - \sum_{j=0}^M \lambda_j(t) u^j(t-1) = 0 \quad (31)$$

Based on the initial conditions of the nonlinear system under control and a *priori* knowledge of the system operation, a proper solution can be easily determined from possible multiple roots of (31). For instance, it may be taken as the root nearest to the control signal in the previous step, or alternatively, the one corresponding to the minimal control effort or energy consumption for implementation. For numerical computation of the roots of (31), there are some alternative algorithms such as the Newton Raphson algorithm, the Matlab function ‘*roots*’, or other computation procedures for real-time application [36].

Note that with the process state-space model in (1), a multi-layer U-model in (11) can be equivalently established to apply the above procedure to recursively resolve the roots layer by layer. A backstepping root solving procedure is given below.

- 1) From the first line of (11), assign $x_1(t+1) = y_r(t)$. Resolve a proper root of $x_2(t)$ from
$$y_r(t) = \sum_{j=0}^{M_1} \lambda_{1j}(t) x_2^j(t).$$
- 2) Assign $x_2(t+1) = x_2(t)$, and similarly determine $x_3(t)$.
- 3) Repeat the above recursion till the last line of (11) to determine the manipulated variable $u(t-1)$.

4. Robust stability analysis

To address the robust stability of the proposed control scheme in the presence of process uncertainties, we lump all of these uncertainties into a multiplicative form,

$$\Delta(z) = \frac{G_p(z) - \hat{G}_p(z)}{\hat{G}_p(z)} \quad (32)$$

where $\hat{G}_p = \hat{G}_{p0} z^{-\tau}$ is a model of the process G_p as shown in Figure 3.

According to the $M - \Delta$ structure for robust stability analysis, the transfer function from the output to the input of $\Delta(z)$ can be derived from Figure 3 as

$$M(z) = -T_d(z) = -\frac{G_c z^{-\tau}}{1 + G_c z^{-\tau}} \quad (33)$$

Hence, it follows from the small gain theorem that the closed-loop structure in Figure 3 holds robust stability if

$$\|T_d(z)\Delta(z)\|_\infty < 1 \quad (34)$$

or expressed as

$$\|T_d(z)\|_\infty < \frac{1}{\|\Delta(z)\|_\infty} \quad (35)$$

Owing to that the U-model based control design specifying the closed-loop transfer function without involving the plant model, the corresponding stability condition can be relatively simplified, given the upper bound of the multiplicative uncertainty of the overall system as specified or estimated in practice. Substituting the desired transfer function in (27) into (35), the robust stability condition is therefore obtained as

$$\left\| \frac{(1-\lambda_c)z^{-\tau+1}}{z-\lambda_c} \right\|_\infty < \frac{1}{\|\Delta(z)\|_\infty} \quad (36)$$

Since a rational z-transform, $z = e^{j\omega T_s}$, is a periodic function with respect to ω , by defining $z = e^{j\theta}$, ($0 < \theta < 2\pi$) and substituting it into (36), the robust stability constraint is derived as

$$\left\| \frac{(1-\lambda_c)\sqrt{\cos^2(\tau-1)\theta + \sin^2(\tau-1)\theta}}{\sqrt{(\cos\theta - \lambda_c)^2 + \sin^2\theta}} \right\|_\infty < \frac{1}{\|\Delta(z)\|_\infty} \quad (37)$$

which may be simplified as

$$\left\| \frac{1-\lambda_c}{\sqrt{\alpha_1^2 + \alpha_2^2}} \right\|_\infty < \left\| \frac{1}{\Delta(z)} \right\|_\infty \quad (38)$$

where

$$\alpha_1 = \cos\theta - \lambda_c, \quad \alpha_2 = \sin\theta.$$

It is seen that the above robust stability constraint is related to the tuning parameter λ_c in G_c , which however could not be analytically solved with respect to a specified bound of $\Delta(z)$. Based on extensive simulations and numerical computations, it is suggested to initially take $\lambda_c \in [0.4, 0.9]$, and then by monotonically increasing or decreasing the value of λ_c , a good trade-off between the closed-loop control performance and its robust stability can be obtained. Note that a smaller value of λ_c could lead to a faster response in eliminating the output error, but in exchange for degraded robustness in the presence of process uncertainties, and vice versa.

In practice, the above robust stability condition can be used to check if the tuned value of λ_c could guarantee the system stability in terms of a prespecified uncertainty bound.

5. Illustration

Three examples studied in the literature are used to perform computational experiments, along with detailed U-model formulations and controller designs for reference, respectively, in order to demonstrate 1) the generality in designing an invariant controller for different examples with polynomial and state-space models, and 2) the superiority in control performance.

Example 1: Consider a Hammerstein system studied in the recent reference [32],

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

Converting it into the U-model gives

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

where $\lambda_0(t) = 0.5y(t-1) + 1 + 0.3x(t-2)$, $\lambda_1(t) = 1$, $\lambda_2(t) = -1$, $\lambda_3(t) = 0.2$.

Using the design formulae in (24) and (28) with $\lambda_r = 0.5$, $n_r = 2$ and $\lambda_c = 0.9$, the controllers in the proposed control scheme are determined as

$$G_r(z) = \frac{0.25z^2}{(z-0.5)^2}, \quad G_c = \frac{0.1z}{z(1-z^{-\tau}) + 0.9(z^{-\tau+1} - 1)}$$

Suppose there is an input delay ($\tau=10$) for the above Hammerstein system. For fair comparison with the U-model control reference [32] that used a pole placement design for the controller tuning, the similar predictive control structure is applied for the reference [32] where the delay-free output obtained by the SP structure is used instead of the real output. The set-point tracking test used in [32] is performed, along with a step load disturbance with a magnitude of 0.5 added to the process output at $t = 225$ (sample period). Note that both methods take the minimum of the absolute values of the roots solved from the U-model for control implementation. The control results are shown in Figure 4. It is seen that the proposed method obtains evidently improved tracking performance with no overshoot. Note that the U-model control scheme given in ref.[32] could not eliminate the output error arising from the load disturbance. For illustration,

the control results by taking two different parameter settings, i.e., $\lambda_r = 0.7$ and $\lambda_c = 0.9$; $\lambda_r = 0.7$ and $\lambda_c = 0.7$, in the proposed control scheme, are also shown in Figure 4. It is seen that a smaller value of $\lambda_r = 0.5$ in G_r leads to a faster set-point tracking speed compared with the choice of $\lambda_r = 0.7$, while the change of λ_c in G_c does not affect the set-point tracking performance.

Assume that the input delay is actually 10% larger and there exists multiplicative uncertainty of the process output, $\Delta(s) = (0.5s + 0.1) / (2s + 1)$, corresponding to a discrete-domain form of $\Delta(z) = (0.25z - 0.2475) / (z - 0.9753)$, which may be roughly regarded as the output measurement increased by up to 25% uncertainty at high frequencies and by almost 10% uncertainty in the low-frequency range. Figure 5 shows the perturbed system responses. It is seen that the proposed method maintains good robustness against these uncertainties. It is also demonstrated that tuning a larger λ_c (i.e., $\lambda_c = 0.9$) gives less oscillatory output response than the choice of $\lambda_c = 0.7$, well in accordance with the tuning guideline given in Section 4.

Then assume that the output measurement is blurred by a Gaussian white noise with zero mean and a variance of 0.15, causing the signal-to-noise ratio about 30%. Figure 6 shows the control results, which demonstrates good robustness of the proposed method against measurement noise.

Example 2: Consider a nonlinear system with input delay studied in [5],

$$\begin{cases} \dot{x} = z + \omega_1 \\ \dot{z} = a \sin x - bz + u(t-d) + \omega_2 \end{cases}$$

where $a = b = 1$ was assumed in [5].

By taking a sampling period of $T_s = 0.01$ (s) and using a zero-order holder for discretization, a state-space model is obtained as

$$\begin{cases} x(t+1) = x(t) + T_s z(t) \\ z(t+1) = z(t) + T_s (a \sin(x(t)) - bz(t)) - \frac{T_s^2 b}{2} + T_s u(t-\tau) \\ y(t) = z(t) \end{cases}$$

Assume that the input delay is $\tau = 18$ (sample period), the corresponding U-model is expressed as

$$y(t+1) = z(t+1) = \lambda_0 + \lambda_1 u(t-\tau)$$

where

$$\lambda_0 = z(t) + T_s (a \sin(x(t)) - bz(t)) - \frac{T_s^2 b}{2}, \lambda_1 = T_s$$

The control objective studied in [5] is performed for illustration, i.e., the system should be stabilized from any initial condition (ϕ_x, ϕ_z) satisfying $(\phi_x(m), \phi_z(m)) = (1, 1)$ for all $m \in [-\tau, 0]$.

To compare with the nonlinear control method given in [5] with the tuned controller parameters of $r = 0.8879$ and $L = 0.1121$ therein, the proposed controller design formulae in (24) and (28) are applied with $\lambda_r = 0.99$, $n_r = 2$ and $\lambda_c = 0.986$, obtaining

$$G_r(z) = \frac{0.0001z^2}{(z-0.99)^2}, G_c = \frac{0.014z}{z(1-z^{-\tau}) + 0.986(z^{-\tau+1} - 1)}$$

which could result in the similar response peak with that of [5] for fair comparison.

Meanwhile, a load disturbance with pulse width of 5(s) and magnitude of 0.2 is added to the process input at $t = 30$ (s). The control results are shown in Figure 7. It is seen that the proposed method gives obviously improved control performance compared to the backstepping control method given in [5].

Then assume that there actually exists multiplicative uncertainty of the process input described by $\Delta(s) = (0.5s + 0.2) / (2s + 1)$, corresponding to the discrete-domain form of $\Delta(z) = (0.25z - 0.2451) / (z - 0.9753)$, which can be roughly regarded as the input actuator contains up to 25% uncertainty at high frequencies and almost 20% uncertainty in the low-frequency range. The perturbed system responses are shown in Figure 8. It is once again seen that the proposed control scheme well maintains robust stability.

Example 3: Consider a nonisothermal CSTR for polymerization reaction with dynamics described by [37],

$$\begin{cases} V \frac{dc_A}{dt'} = \lambda F c_{A_f} + F(1-\lambda)c_A - Fc_A - V k_0 \exp\left(-\frac{E}{RT}\right) c_A \\ VC_p \rho \frac{dT}{dt'} = \rho C_p F [\lambda T_f + (1-\lambda)T - T] + V(-\Delta H) k_0 \exp\left(-\frac{E}{RT}\right) c_A - UA(T - T_c) \end{cases}$$

where C_A is the concentration of A for an exothermic irreversible reaction ($A \rightarrow B$), T is the temperature of the reactor, and T_c is the temperature of the coolant stream which is taken as the manipulated variable.

The CSTR exhibits highly nonlinear behavior in the normal operating regime. By defining some dimensionless variables, $D_a = 0.072$, $\gamma = 20$, $B = 8$, $\beta = 0.3$, the corresponding model studied in [38] is adopted here,

$$\begin{cases} \dot{x}_1 = -x_1 + D_a(1-x_1)\exp\left(-\frac{x_2}{1+x_2/\gamma}\right) \\ \dot{x}_2 = -x_2 + BD_a(1-x_1)\exp\left(-\frac{x_2}{1+x_2/\gamma}\right) + \beta(u-x_2) \\ y = x_2 \end{cases}$$

where x_1 , x_2 and u denotes the dimensionless reagent conversion, solution temperature, and coolant temperature, respectively, expressed by

$$x_1 = \frac{c_{Af} - c_A}{c_{Af}}, \quad x_2 = \frac{T - T_f}{T_f} \left(\frac{E}{RT_f} \right), \quad D_a = \frac{k_0 e^{-\gamma V}}{F \lambda},$$

$$B = \frac{(-\Delta H)c_{Af}}{\rho T_f C_p} \left(\frac{E}{RT_f} \right), \quad \beta = \frac{UA}{F \rho C_p}, \quad \gamma = \frac{E}{RT_f},$$

By taking a sampling period of $T_s = 0.05$ (s) and using a zero-order holder for discretization, a state-space model is obtained as

$$\begin{cases} x_1(t+1) = x_1(t) + T_s \delta_1 + \frac{T_s^2}{2} \left[-\delta_1 - D_a \exp\left(-\frac{x_2(t)}{1+x_2(t)/\gamma}\right) \delta_1 \right] + \\ \frac{T_s^3}{6} \left[\delta_1 + 2D_a \exp\left(-\frac{x_2(t)}{1+x_2(t)/\gamma}\right) \delta_1 + [D_a \exp\left(-\frac{x_2(t)}{1+x_2(t)/\gamma}\right)]^2 \delta_1 \right] \\ x_2(t+1) = x_2(t) + T_s \delta_2 + \frac{T_s^2}{2} \delta_3 + \frac{T_s^3}{6} \left\{ \delta_2 + BD_a \exp\left(-\frac{x_2(t)}{1+x_2(t)/\gamma}\right) \delta_2 \right. \\ \left. - BD_a \left[\exp\left(-\frac{x_2(t)}{1+x_2(t)/\gamma}\right) \left(\frac{\delta_2}{(1+x_2(t)/\gamma)^2} \right)^2 + \exp\left(-\frac{x_2(t)}{1+x_2(t)/\gamma}\right) \delta_3 (1+x_2(t)/\gamma)^2 - \frac{2\delta_2^2(1+x_2(t)/\gamma)}{\gamma} \right] \right. \\ \left. + B\delta_2 - \frac{\beta BD_a \exp\left(-\frac{x_2(t)}{1+x_2(t)/\gamma}\right) \delta_2}{1+x_2(t)/\gamma} + \beta^2 \delta_2 \right\} \\ y(t) = x_2(t) \end{cases}$$

where

$$\delta_1 = -x_1(t) + D_a(1 - x_1(t)) \exp\left(-\frac{x_2(t)}{1 + x_2(t)/\gamma}\right)$$

$$\delta_2 = -x_2(t) + BD_a(1 - x_1(t)) \exp\left(-\frac{x_2(t)}{1 + x_2(t)/\gamma}\right) + \beta[u(t - \tau) - x_2(t)]$$

$$\delta_3 = -\delta_2 - \frac{BD_a(1 - x_1(t)) \exp\left(-\frac{x_2(t)}{1 + x_2(t)/\gamma}\right) \delta_2}{(1 + x_2(t)/\gamma)^2} - \beta \delta_2$$

Assume that the input delay is $\tau = 15$ (sample period), the corresponding U-model is expressed as

$$y(t+1) = x_2(t+1) = \lambda_0 + \lambda_1 u(t - \tau) + \lambda_2 u^2(t - \tau)$$

where

$$\begin{aligned} \lambda_0 = & x_2(t) + T_s \delta'_2 + \frac{T_s^2}{2} \delta_3'' + \frac{T_s^3}{6} \left\{ \delta'_2 + BD_a \exp\left(-\frac{x_2(t)}{1 + x_2(t)/\gamma}\right) \delta'_2 \right. \\ & - BD_a \left[\exp\left(-\frac{x_2(t)}{1 + x_2(t)/\gamma}\right) \frac{\delta'_2}{(1 + x_2(t)/\gamma)^4} + \exp\left(-\frac{x_2(t)}{1 + x_2(t)/\gamma}\right) \delta_3'' (1 + x_2(t)/\gamma)^2 - \frac{2\delta_2'^2 (1 + x_2(t)/\gamma)}{\gamma} \right] \\ & \left. + B \delta_2' - \frac{\beta BD_a \exp\left(-\frac{x_2(t)}{1 + x_2(t)/\gamma}\right) \delta'_2}{1 + x_2(t)/\gamma} + \beta^2 \delta_2' \right\} \\ \lambda_1 = & T_s \beta + \frac{T_s^2}{2} \delta_3' + \frac{T_s^3}{6} \left\{ \beta + BD_a \exp\left(-\frac{x_2(t)}{1 + x_2(t)/\gamma}\right) \beta \right. \\ & - BD_a \left[\exp\left(-\frac{x_2(t)}{1 + x_2(t)/\gamma}\right) \frac{2\delta_2' \beta}{(1 + x_2(t)/\gamma)^4} + \exp\left(-\frac{x_2(t)}{1 + x_2(t)/\gamma}\right) (1 + x_2(t)/\gamma)^2 \delta_3' - \frac{2\delta_2' (1 + x_2(t)/\gamma)}{\gamma} \right] \\ & \left. + B \beta - \frac{\beta BD_a \exp\left(-\frac{x_2(t)}{1 + x_2(t)/\gamma}\right) \beta}{1 + x_2(t)/\gamma} + \beta^3 \right\} \\ \lambda_2 = & -\frac{T_s^3}{6} BD_a \left[\exp\left(-\frac{x_2(t)}{1 + x_2(t)/\gamma}\right) \frac{\beta^2}{(1 + x_2(t)/\gamma)^4} - \exp\left(-\frac{x_2(t)}{1 + x_2(t)/\gamma}\right) \frac{2\beta^2 (1 + x_2(t)/\gamma)}{\gamma} \right]; \end{aligned}$$

$$\delta_2' = -x_2(t) + BD_a [1 - x_1(t)] \exp\left(-\frac{x_2(t)}{1 + x_2(t)/\gamma}\right) - \beta x_2(t)$$

$$\delta_3' = -\beta - \frac{BD_a [1 - x_1(t)] \exp\left(-\frac{x_2(t)}{1 + x_2(t)/\gamma}\right) \beta}{[1 + x_2(t)/\gamma]^2} - \beta^2$$

$$\delta_3'' = -\delta_2' - \frac{BD_a[1-x_1(t)]\exp(-\frac{x_2(t)}{1+x_2(t)/\gamma})\delta_2'}{[1+x_2(t)/\gamma]^2} - \beta\delta_2'$$

The control objective studied in [38] is to transfer the system from the initial stable equilibrium point $(x_{10}, x_{20}) = (0.0504, 0.31)$ to the desired equilibrium point $(0.0064, 2.75)$.

To compare with the LADRC method given in [38] with the controller parameters of $b=0.35$, $\omega_c=2.5$, $\omega_0=1.9$, the proposed controller design formulae in (24) and (28) with $\lambda_r=0.8$, $n_r=2$ and $\lambda_c=0.9$ give

$$G_r(z) = \frac{0.04z^2}{(z-0.8)^2}, \quad G_c = \frac{0.1z}{z(1-z^{-\tau})+0.9(z^{-\tau+1}-1)}$$

which could obtain the similar set-point tracking speed with that of LADRC for fair comparison. Also, the modified ADRC method given in [21] for time-delay systems is used for comparison, where the controller parameters are taken as $b=0.35$, $\omega_c=2.5$, $\omega_0=10$ according to the guidelines given therein.

A step load disturbance with a magnitude of 0.5 is added to the process output at $t=10$ (sample size). The control results are shown in Figure 7. It is seen that the proposed method gives obviously improved control performance compared to the LADRC method [38] and the modified ADRC method [21].

Assume that there actually exists multiplicative uncertainty of the process input described by $\Delta(s) = (0.5s+0.2)/(2s+1)$, corresponding to the discrete-domain form of $\Delta(z) = (0.25z-0.2451)/(z-0.9753)$, which can be roughly regarded as the input actuator contains up to 25% uncertainty at high frequencies and almost 20% uncertainty in the low-frequency range. The perturbed system responses are shown in Figure 8. It is seen that the proposed control scheme well maintains robust stability.

Assume that the output measurement is blurred by a Gaussian white noise with zero mean and a variance of 0.002, causing the signal-to-noise ratio about 20%. Figure 9 shows the control results, which demonstrates good robustness of the proposed method against measurement noise.

6. Conclusions

A general predictive control scheme has been proposed for controlling nonlinear processes with input delay, by using the U-model framework to represent various nonlinear process models of polynomial, NARMAX, state-space, Hammerstein or Wiener type. An important merit is that the proposed control structure including the controller forms are relatively independent of the process model, based on a linear-like system expression by the U-model. Moreover, the set-point tracking and load disturbance rejection can be separately tuned in the proposed control scheme. Both the set-point tracking controller and the closed-loop controller for disturbance rejection are analytically derived by proposing the desired transfer functions for an ideal ‘linear’ plant, based on using the U-model inverse. It is convenient to tune a single adjustable parameter of both controllers to obtain a good trade-off between the control performance and its robustness against process uncertainties. The applications to three illustrative examples from the literature with detailed procedures for the control designs have well demonstrated the effectiveness and advantage of the proposed method. As this is a model based approach, it could be sensitive to different modeling errors and external disturbances. The future work may consider to design specific controllers to enhance robust control performance under different types of process perturbations or disturbances.

Acknowledgements

This work is partially supported by high rank overseas expert project, Yunnan Province, China, and the NSF China Grants 61633006, 61473054, and 61573174. The authors would like to thank the anonymous reviewers for their constructive suggestions to improve the paper quality.

References

- [1] J.E. Normey-Rico, Control of dead-time processes, Springer, London, UK, 2007.
- [2] T. Liu, F. Gao, Industrial process identification and control design: step-test and relay-experiment-based methods, Springer, London, UK, 2012.
- [3] O. Boubaker, V.E. Balas, A. Benzaouia, M. Chaabane, M.S. Mahmoud, Q.M. Zhu, Time-delay systems: modeling, analysis, estimation, control, and synchronization, *Mathematical Problems in Engineering* 2017 (2017) 1398904-1-3.
- [4] F. Mazenc, P.A. Bliman, Backstepping design for time-delay nonlinear systems, *IEEE Trans. Autom. Control* 51 (2006) 149-154.
- [5] F. Mazenc, S.I. Niculescu, M. Bekaik, Backstepping for nonlinear systems with delay in the input revisited, *SIAM J. Control Optim.* 49 (2011) 2239-2262.
- [6] N. Bekiaris-Liberis, M. Krstic, Compensation of time-varying input and state delays for nonlinear systems, *J. Dyn. Sys. Meas. Control* 134 (2012) 011009-1-14.
- [7] F. Mazenc, M. Malisoff, New control design for bounded backstepping under input delays, *Automatica* 66 (2016) 48-55.
- [8] T. Ogushi, J.P. Richard, Sliding-mode control of retarded nonlinear systems via finite spectrum assignment approach, *IEEE Trans. Autom. Control* 51 (2006) 1527-1531.
- [9] B. Zhou, Z. Lin, G. Duan, Truncated predictor feedback for linear systems with long time-varying input delay, *Automatica* 48 (2012) 2387-2399.
- [10] Z. Zuo, Z. Lin, Z. Ding, Truncated prediction output feedback control of a class of Lipschitz nonlinear systems with input delay, *IEEE Trans. Circuits Syst. II: Brief Papers* 63 (2016) 788-792.
- [11] J. Lei, H.K. Khalil, Feedback linearization for nonlinear systems with time-varying input and output delays by using high-gain predictors, *IEEE Trans. Autom. Control* 61 (2016) 2262-2268.
- [12] I. Karafyllis, M. Krstic, Stabilization of nonlinear delay systems using approximate predictors and high-gain observers, *Automatica* 49 (2013) 3623-3631.
- [13] Germani, A., Manes, C., & Pepe, P. (2002). A new approach to state observation of nonlinear systems with delayed output. *IEEE Transactions on Automatic Control*, 47(1), 96-101.
- [14] Lima, D. M., Santos, T. L. M., & Normey-Rico, J. E. (2015). Robust nonlinear predictor for dead-time systems with input nonlinearities. *Journal of Process Control*, 27, 1-14.

- [15] I. Karafyllis, M. Krstic, Approximate and sampled-data predictors for control of nonlinear delay systems, *Annu. Rev. Control* 41 (2016) 2-12.
- [16] J. Na, X. Ren, C. Shang, Y. Guo, Adaptive neural network predictive control for nonlinear pure feedback systems with input delay, *J. Process Control* 22 (2012) 194-206.
- [17] Teng, L., Wang, Y., Cai, W., & Li, H. (2017). Robust model predictive control of discrete nonlinear systems with time delays and disturbances via T-S fuzzy approach. *Journal of Process Control*, 53, 70-79.
- [18] J. Belikov, E. Petlenkov, NN-ANARX model based control of nonlinear discrete-time systems with input delay, in: *Proceedings of the 3rd IEEE Multi-conference on Systems and Control* (2009)1039-1044.
- [19] J. Choi, M. Krstic, Compensation of time-varying input delay for discrete-time nonlinear systems, *Int. J. Robust and Nonlinear Control* 26 (2016) 1755-1776.
- [20] X. Cai, L. Liao, Y. Liu, C. Lin, Predictor-based stabilisation for discrete nonlinear systems with state-dependent input delays, *Int. J. of Systems Science* 48 (2017) 769-777.
- [21] Zhao, S., & Gao, Z. (2014). Modified active disturbance rejection control for time-delay systems. *ISA transactions*, 53(4), 882-888.
- [22] Liu, T., Hao, S., Li, D., Chen, W. H., & Wang, Q. G. (2018). Predictor-based disturbance rejection control for sampled systems with input delay. *IEEE Transactions on Control Systems Technology*, published on line, <http://dx.doi.org/10.1109/TCST.2017.2781651>.
- [23] O. Smith, Closer control of loops with dead time, *Chem. Eng. Progr.* 53 (1957) 217-219.
- [24] K.J. Åström, C.C. Hang, B.C. Lim, A new Smith predictor for controlling a process with an integrator and long dead time, *IEEE Trans. Autom. Control* 39 (1994) 343-345.
- [25] H.J. Kwak, S.W. Sung, I.B. Lee, J.Y. Park. A modified Smith predictor with a new structure for unstable processes, *Ind. Eng. Chem. Res.* 38 (1999) 405-411.
- [26] C.C. Hang, Q.G. Wang, X.P. Yang. A modified Smith predictor for a process with an integrator and long dead time. *Ind. Eng. Chem. Res.* 42 (2003) 484-489.
- [27] A.S. Rao, M. Chidambaram, Analytical design of modified Smith predictor in a two-degrees-of-freedom control scheme for second order unstable processes with time delay, *ISA Trans.* 47 (2008) 407-419.
- [28] P. García, P. Albertos, Robust control design for long time-delay systems. *J. Process Control* 19 (2009) 1640-8.
- [29] P. García, T. Santos, J.E. Normey-Rico, et al. Smith predictor-based control schemes for

- dead-time unstable cascade processes. *Ind. Eng. Chem. Res.* 49 (2010) 11471-11481.
- [30] J.E. Normey-Rico, P. Garcia, A. Gonzalez, Robust stability analysis of filtered Smith predictor for time-varying delay processes, *J. Process Control* 22 (2012) 1975-1984.
- [31] Kravaris, C. and Wright, R.A., Deadtime compensation for nonlinear processes, *AIChE J.* 35 (1989) 1537-1542.
- [32] Q. Zhu, D. Zhao, J. Zhang, A general U-Block model based design procedure for nonlinear polynomial control systems, *Int. J. of Systems Science* 47 (2016) 3465-3475.
- [33] Q. Zhu, D. Zhao, S. Zhang, P. Narayan. U-model enhanced dynamic control of a heavy oil pyrolysis/cracking furnace, *IEEE/CAA Journal of Automatica Sinica*, 5 (2018) 577–586.
- [34] Q. Zhu, L. Liu, W. Zhang, S. Li. Control of complex nonlinear dynamic rational systems, *Complexity* 2018 (2018) 1–12.
- [35] M. Morari, E. Zafiriou, *Robust process control*, Prentice Hall, New Jersey, 1989.
- [36] W.H. Press, *Numerical recipes 3rd edition: The art of scientific computing*, Cambridge University Press, 2007.
- [37] A. Uppal, W.H. Ray, A.B. Poore, On the dynamic behavior of continuous stirred tank reactors, *Chem. Eng. Sci.* 29 (1974) 967-985.
- [38] W. Tan, C.F. Fu, Linear active disturbance-rejection control: analysis and tuning via IMC, *IEEE Trans. Ind. Electron.* 63 (2016) 2350-2359.

List of Figure Captions

Figure 1. A general framework of U-model based control system

Figure 2. Smith predictor based U-model control structure

Figure 3. Proposed U-model based predictive control scheme

Figure 4. Control results for Example 1: (a) output response, (b) control signal

Figure 5. Perturbed system responses of Example 1: (a) output response, (b) control signal

Figure 6. Control results under measurement noise for Example 1: (a) output response, (b) control signal

Figure 7. Control results for Example 2: (a) output response, (b) control signal

Figure 8. Perturbed system responses of Example 2: (a) output response, (b) control signal

Figure 9. Control results for Example 3: (a) output response, (b) control signal

Figure 10. Perturbed system responses of Example 3: (a) output response, (b) control signal

Figure 11. Control results under measurement noise for Example 3: (a) output response, (b) control signal

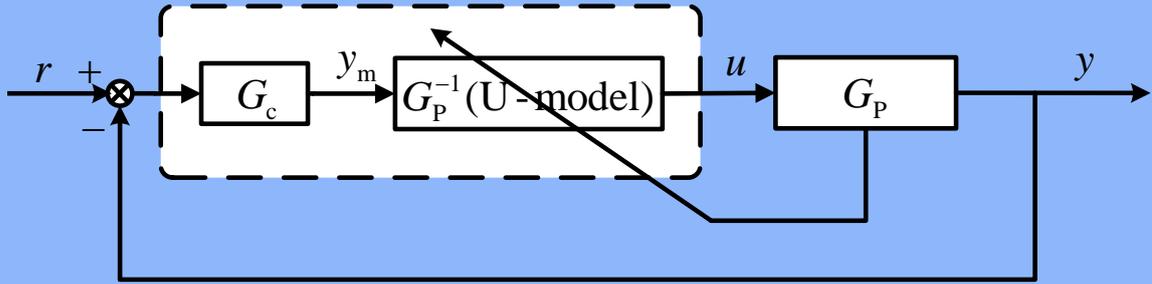


Figure 1. A general framework of U-model based control system

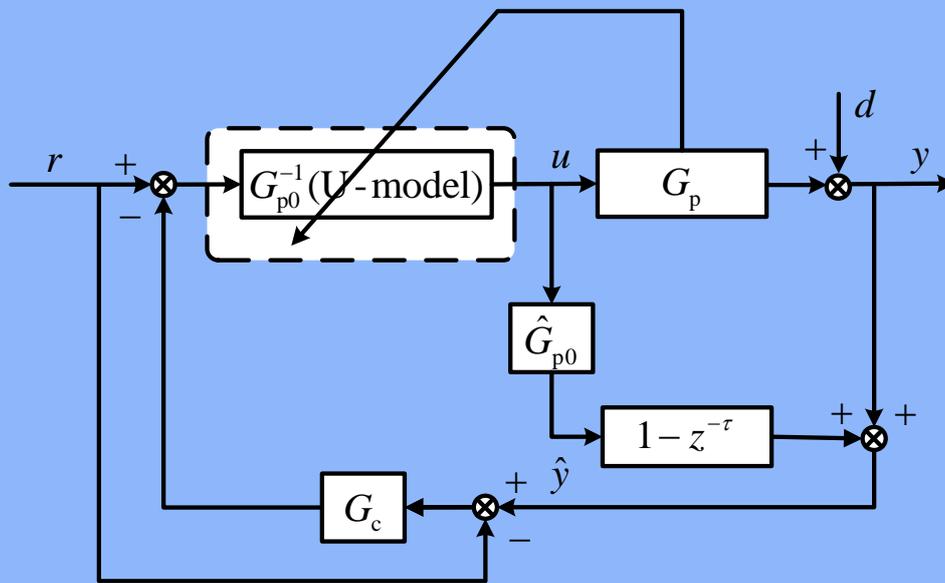


Figure 2. Smith predictor based U-model control structure

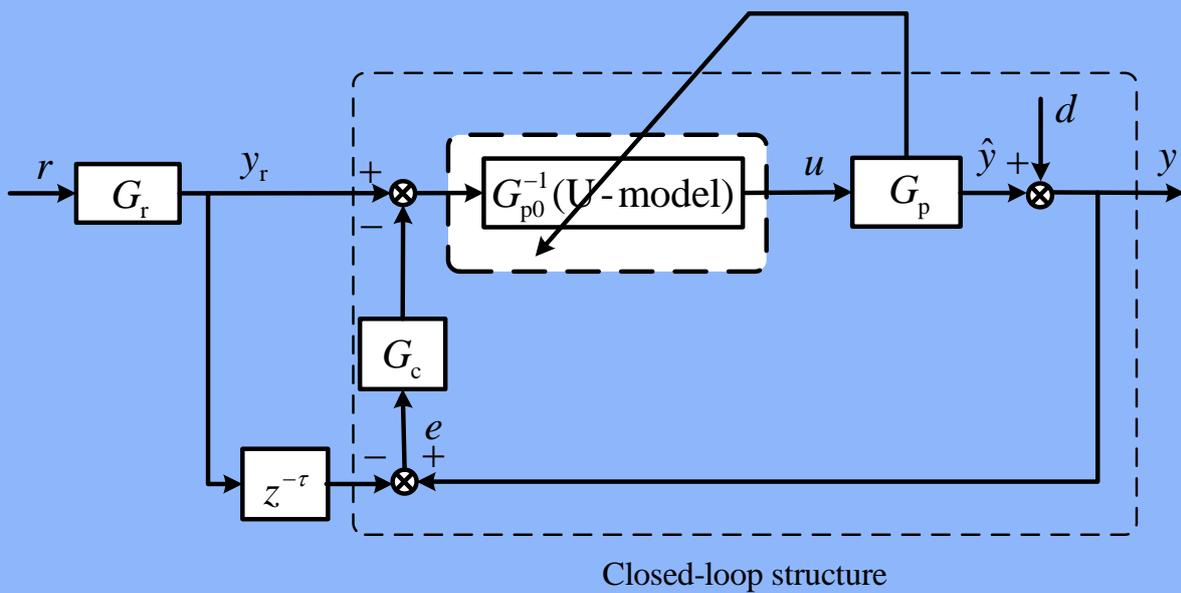


Figure 3. Proposed U-model based predictive control scheme

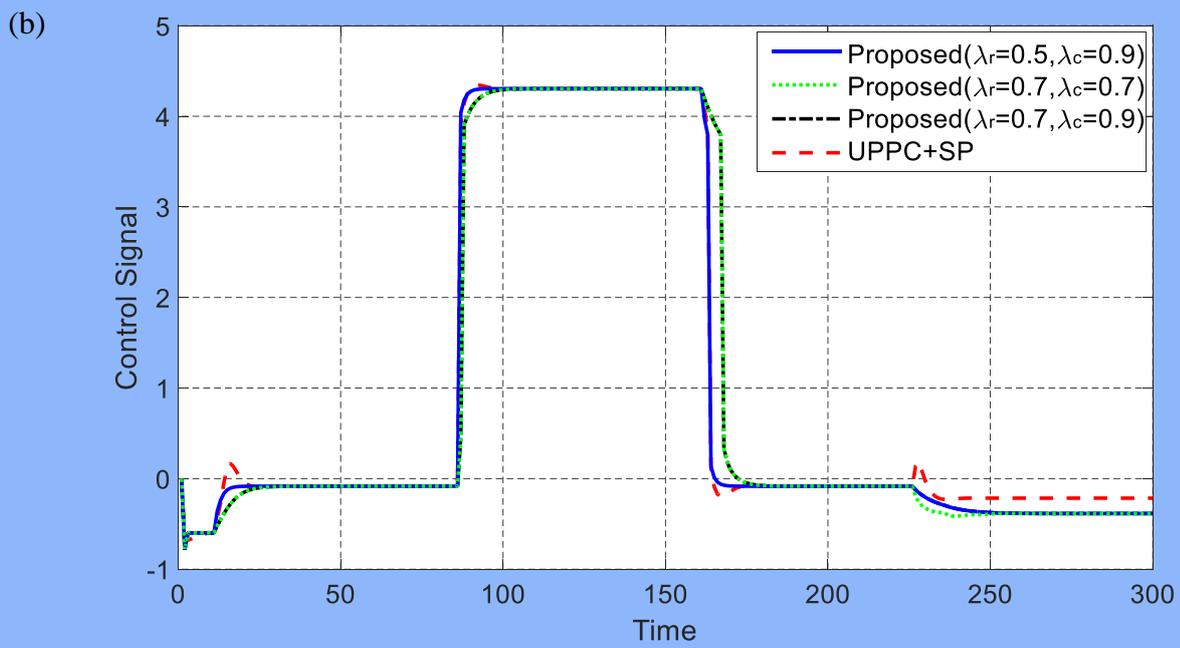
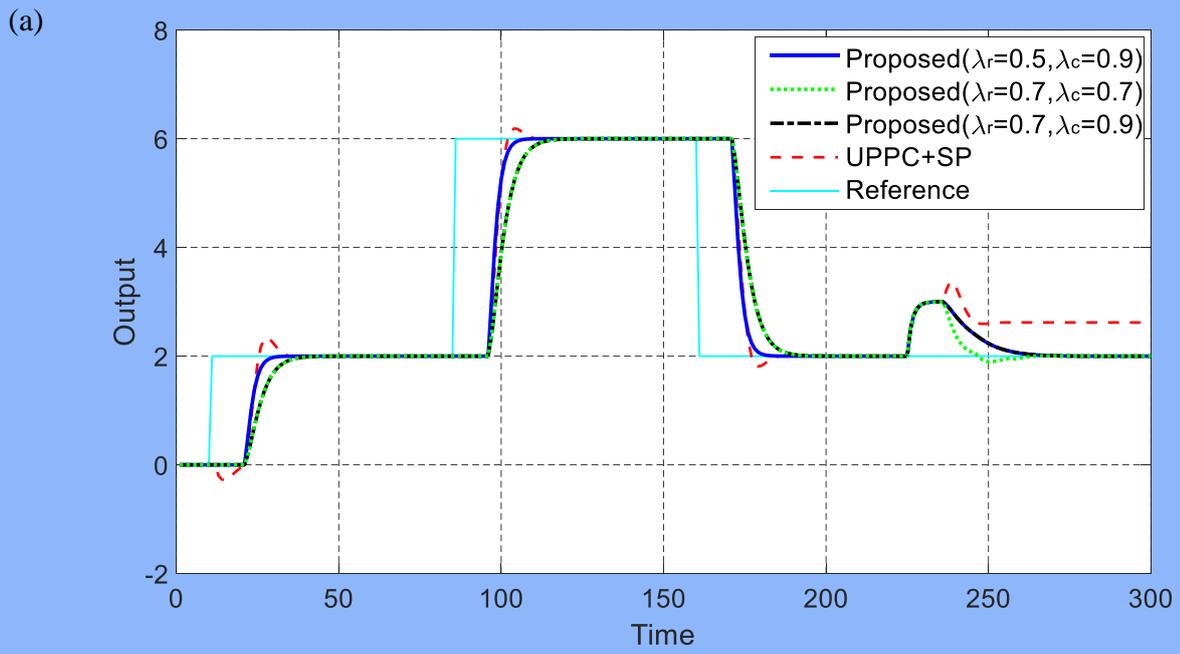


Figure 4. Control results for Example 1: (a) output response, (b) control signal

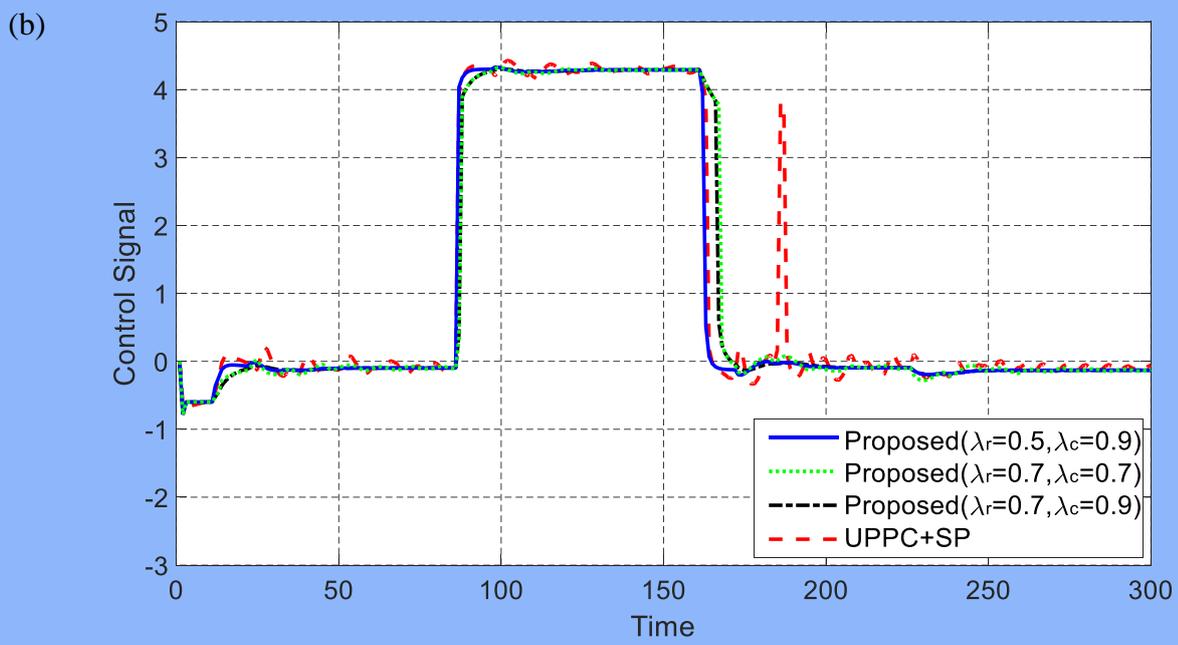
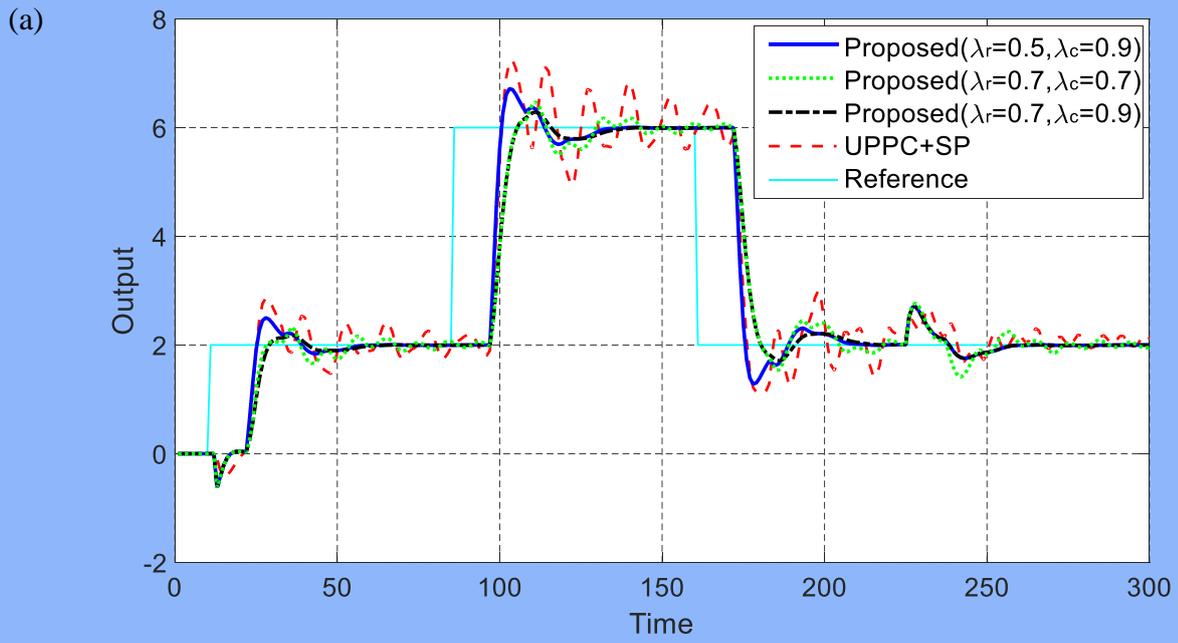


Figure 5. Perturbed system responses of Example 1: (a) output response, (b) control signal

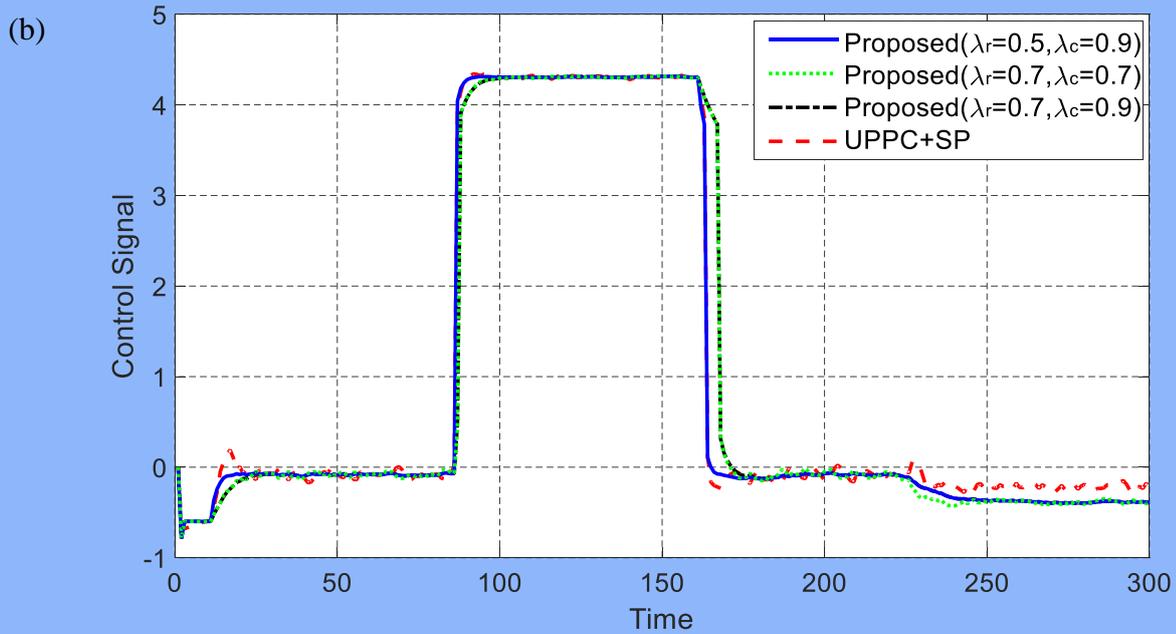
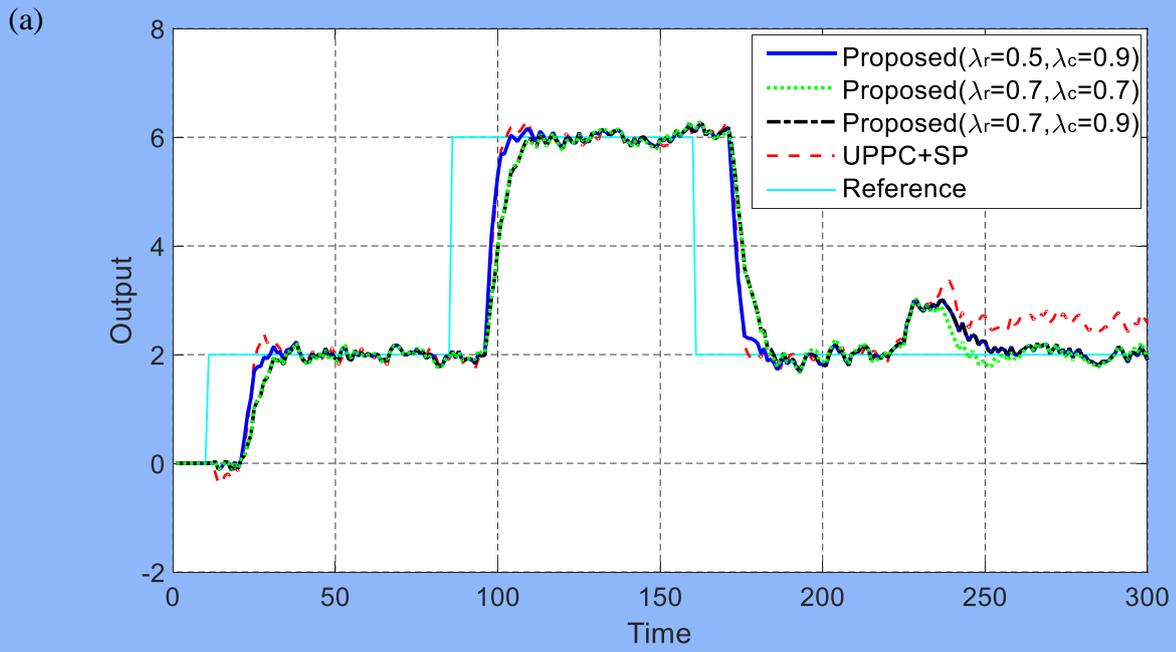


Figure 6. Control results under measurement noise for Example 1: (a) output response, (b) control signal

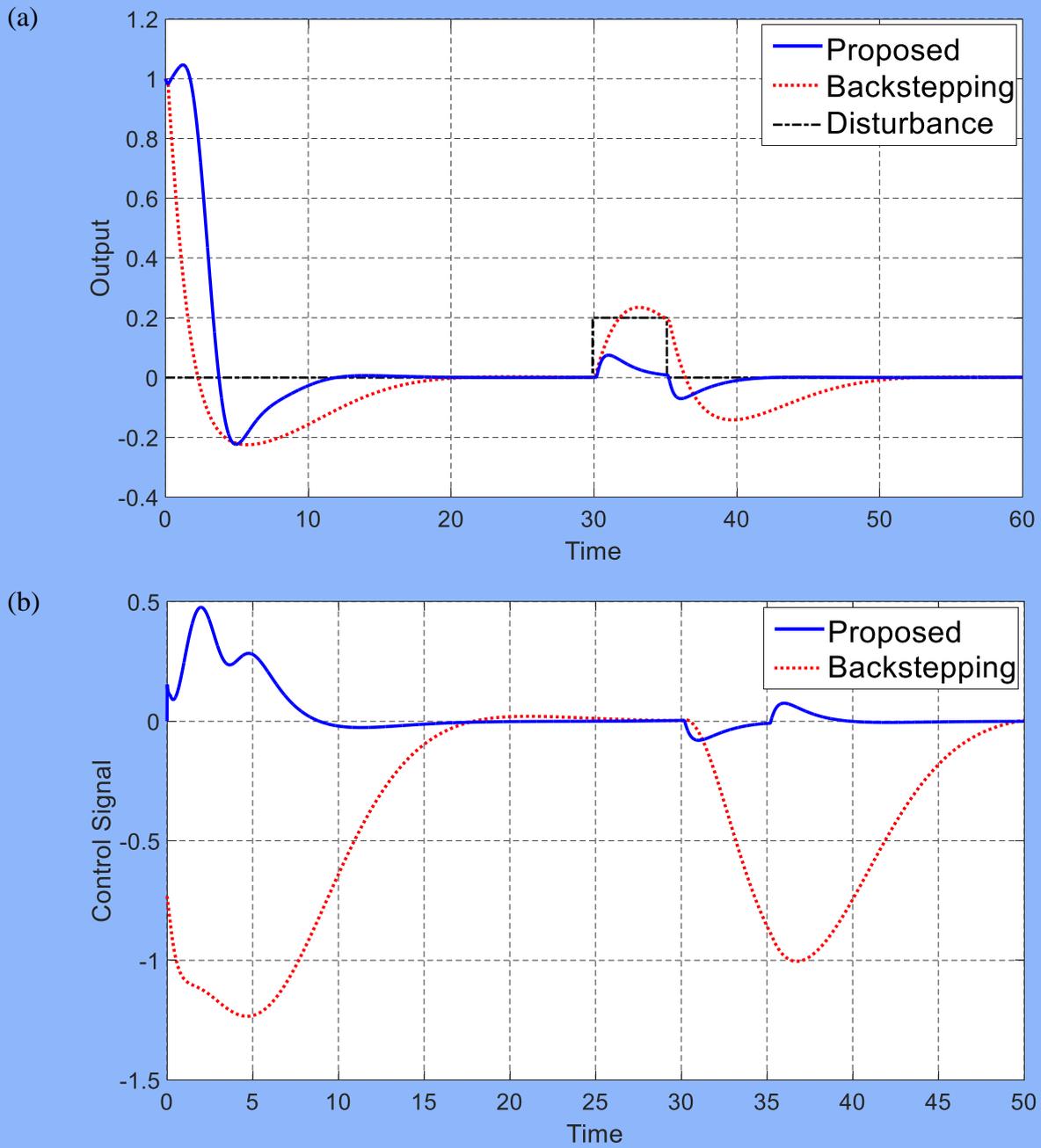


Figure 7. Control results for Example 2: (a) output response, (b) control signal

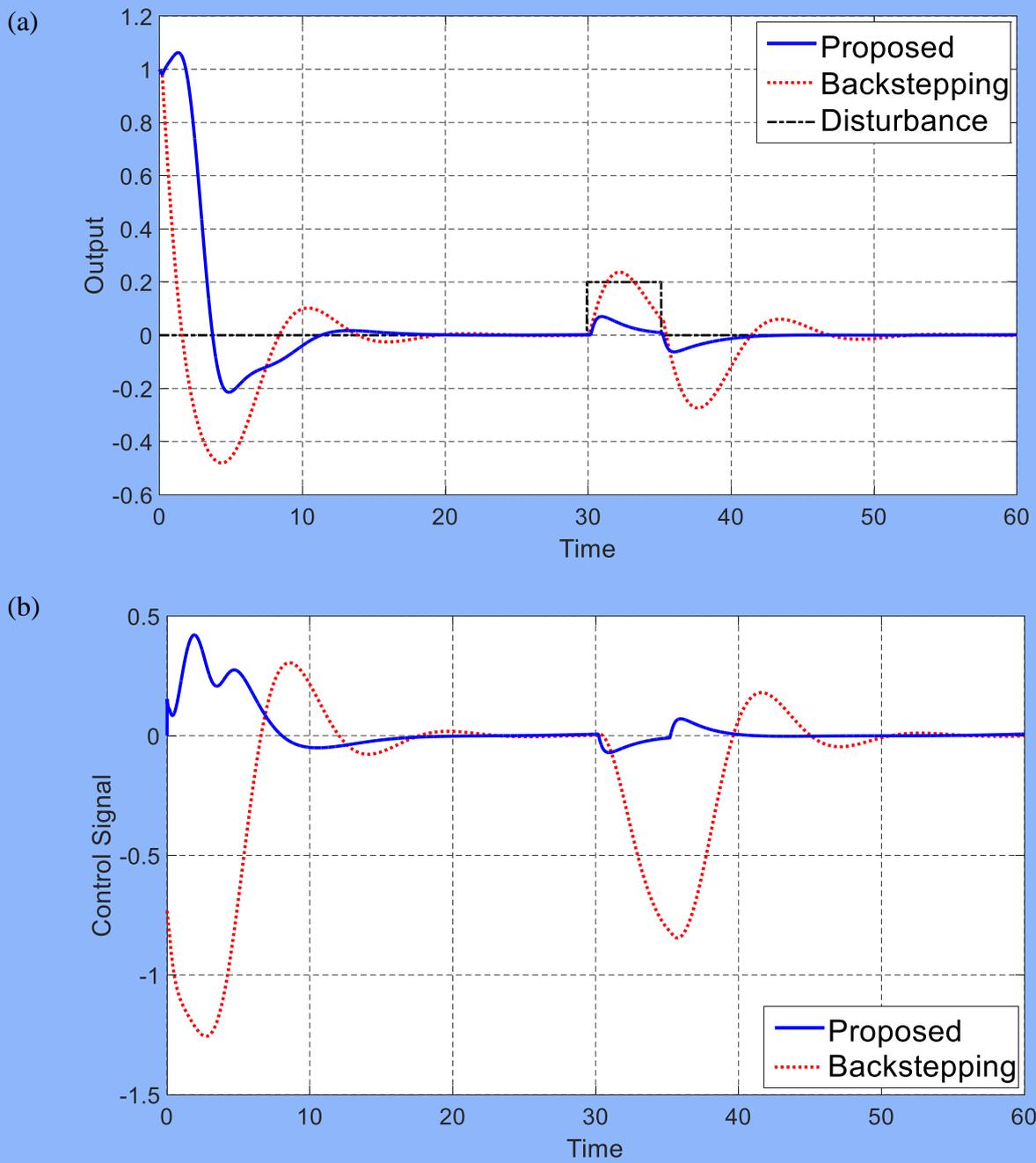


Figure 8. Perturbed system responses of Example 2: (a) output response, (b) control signal

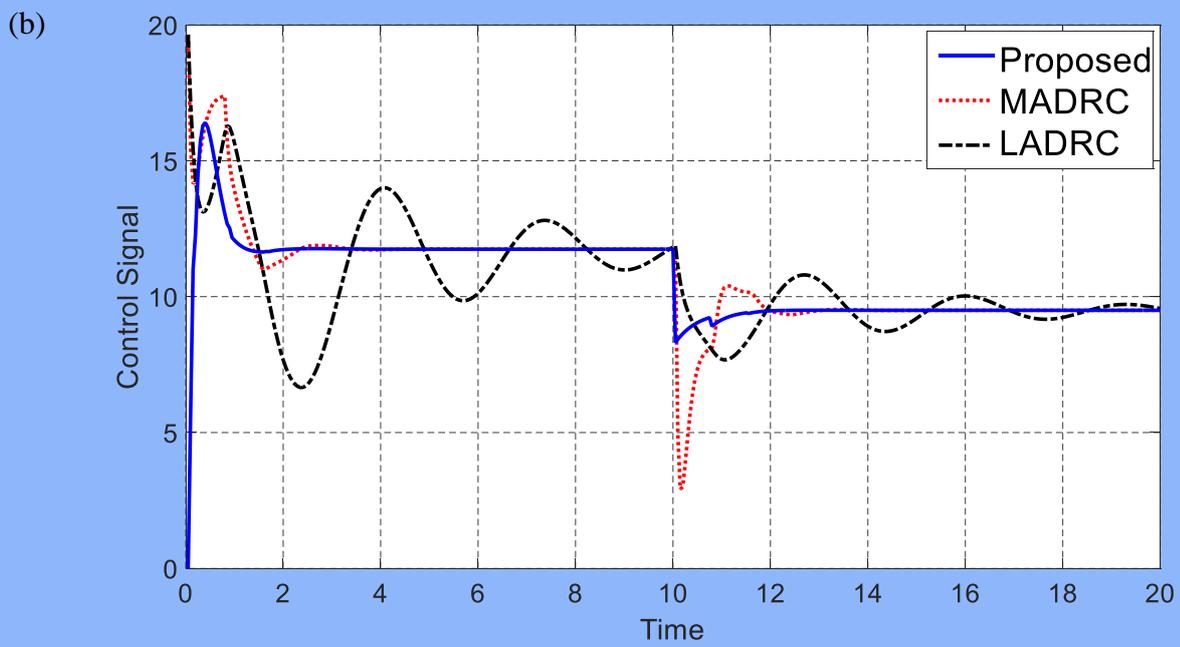
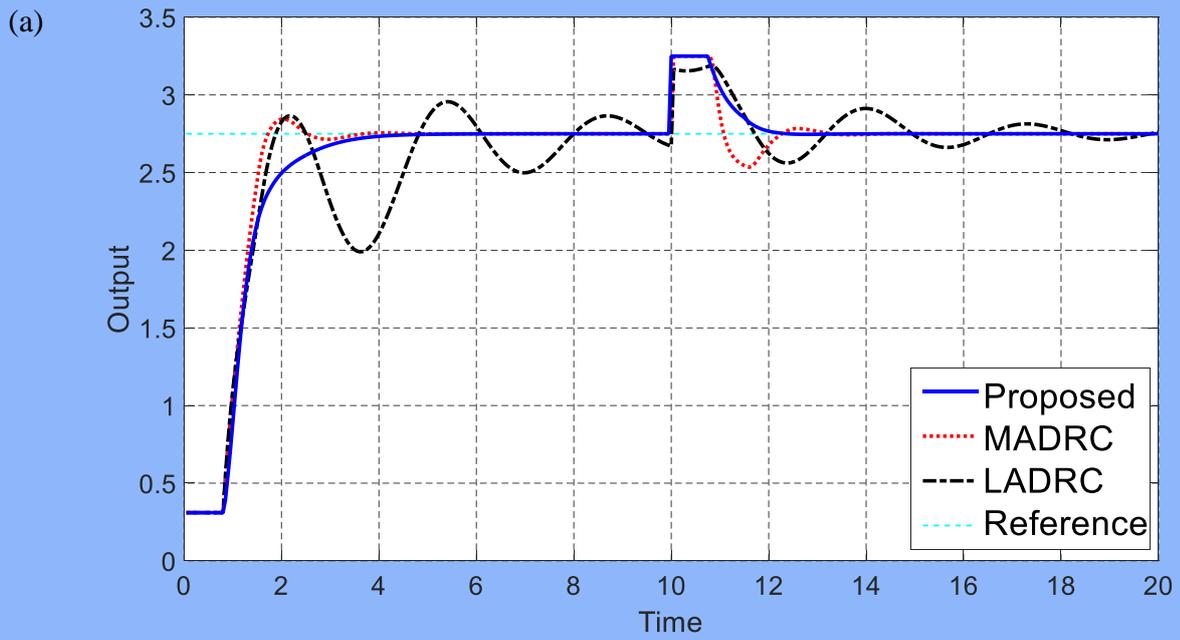


Figure 9. Control results for Example 3: (a) output response, (b) control signal

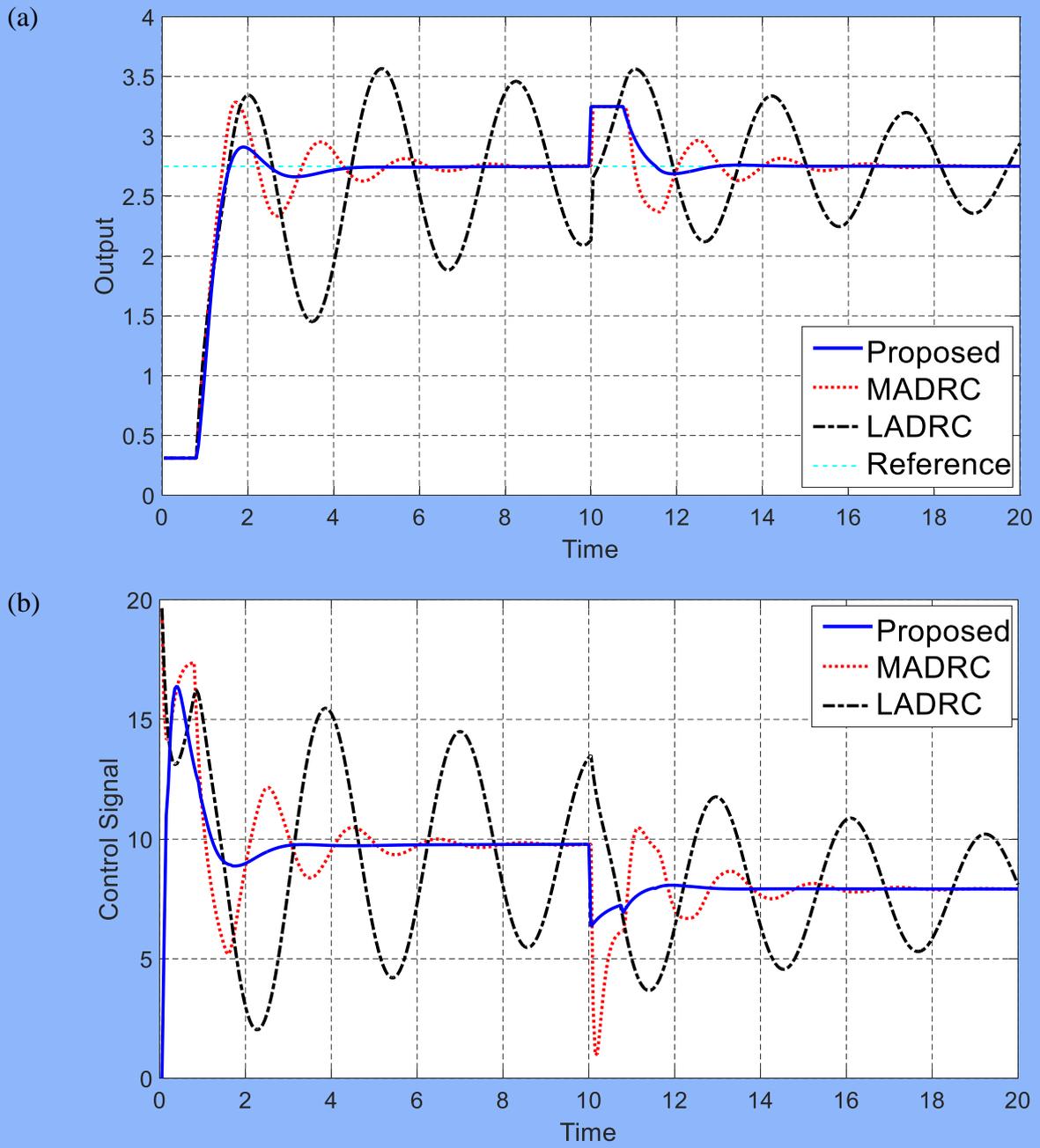


Figure 10. Perturbed system responses of Example 3: (a) output response, (b) control signal

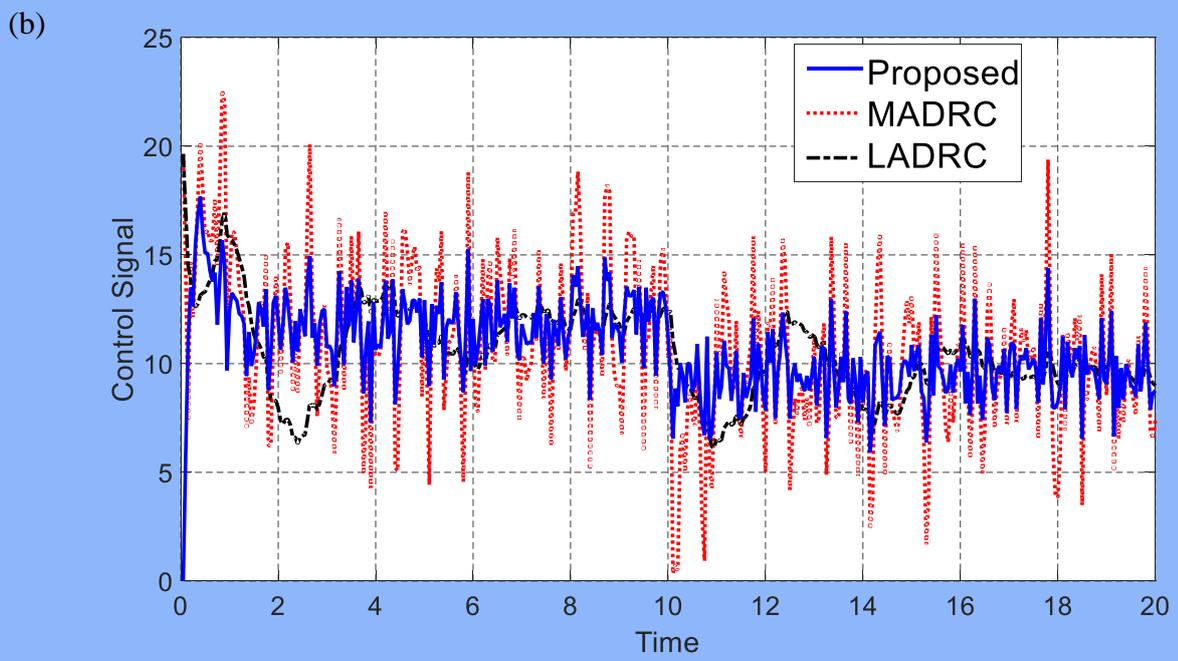
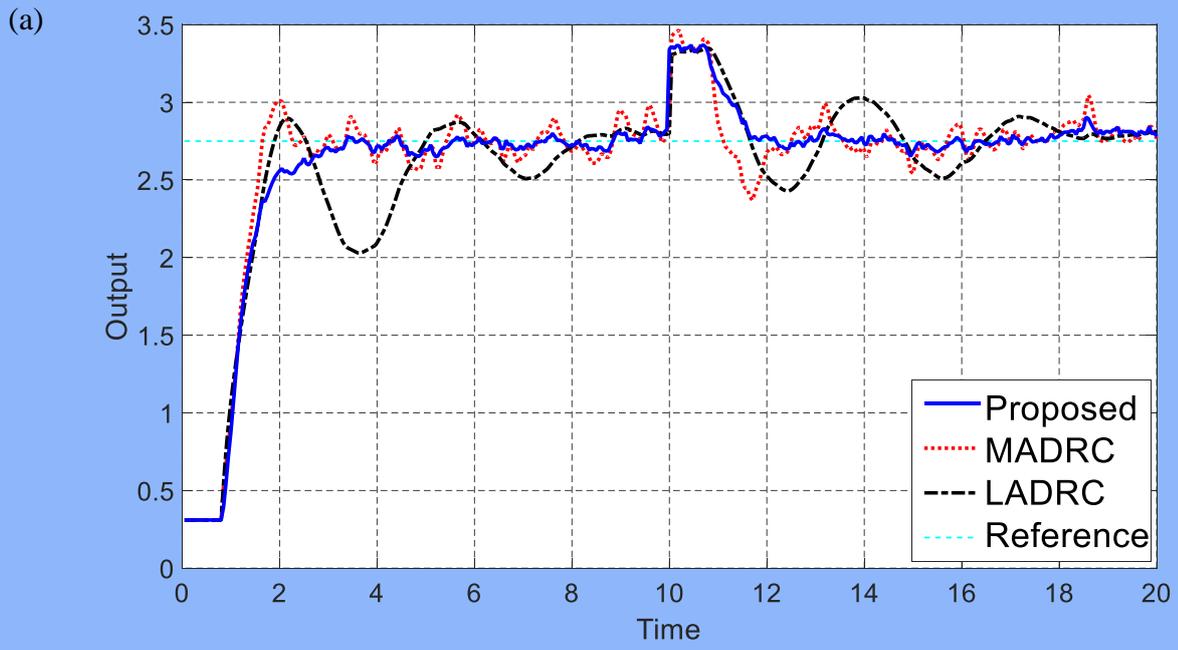


Figure 11. Control results under measurement noise of Example 3: (a) output response, (b) control signal