

Research Article

Stability and Convergence Analysis of Direct Adaptive Inverse Control

Muhammad Shafiq,¹ Muhammad A. Shafiq,² and Hassan A. Yousef¹

¹ECE, SQU, Muscat, Oman

²ECE, GIT, Atlanta, GA, USA

Correspondence should be addressed to Muhammad Shafiq; shafiqsaeeda@yahoo.com

Received 14 June 2017; Accepted 10 October 2017; Published 14 November 2017

Academic Editor: Danilo Comminiello

Copyright © 2017 Muhammad Shafiq et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In adaptive inverse control (AIC), adaptive inverse of the plant is used as a feed-forward controller. Majority of AIC schemes estimate controller parameters using the indirect method. Direct adaptive inverse control (DAIC) alleviates the adhocism in adaptive loop. In this paper, we discuss the stability and convergence of DAIC algorithm. The computer simulation results are presented to demonstrate the performance of the DAIC. Laboratory scale experimental results are included in the paper to study the efficiency of DAIC for physical plants.

1. Introduction

Adaptive inverse control (AIC) is a well-established adaptive tracking methodology [1–4]. Robust tracking and computationally less expensive characteristics of AIC have attracted the interest of many researchers for several decades [1–15]. AIC schemes are applicable to stable or stabilized plants [2]. AIC has been applied successfully in several practical applications such as real-time blood pressure control, shock testing, control of the kiln, real-time control of temperature of a heating process, real-time speed control of a brush DC motor, nonlinear ship maneuvering, echo cancelation, and noise cancelation [1, 15–20]. Recently, AIC is used to control the position of piezoelectric inchworm actuator [21] and the acceleration of six-degree-of-freedom electrohydraulic shaking table [22].

Discrete time plants for which one or more zeros lie outside the unit circle are called non-minimum phase plants. Similarly continuous time plants in which one or more zeros lie on the right hand side of the S -plane are known as non-minimum phase plants [23]. Discretization of continuous time plants most often gives non-minimum phase discrete plants [24, 25]. Numerous techniques have been developed for the control of non-minimum phase plants. AIC based on

linear and nonlinear filtering, AIC of linear and nonlinear systems using dynamic neural networks, Normalized Least Means Square (NLMS) based adaptive controller, nonlinear adaptive inverse control systems based on filtered- ϵ Least Mean Square (LMS) Algorithm, internal model control structure using adaptive inverse control strategy, k -step delay controller for robust tracking, and causal inversion solution are few of them [5–14, 17, 26, 27]. Majority of control schemes for non-minimum phase plants are indirect [5–8, 11]. In some AIC schemes inverse is designed based on identified plant [9, 10]. Most of the AIC schemes estimate right inverse $\widehat{Q}_R(q^{-1})$ and then it is used as left inverse $\widehat{Q}_L(q^{-1})$ by considering left and right inverse are equal. But they are not equal, because practical plants most often have some kind of nonlinearities. Therefore, directly estimated left inverse in principle accomplishes better tracking than indirect algorithms. Since the plant and its inverse are in cascade, they collectively form a unity gain transfer function. Similarly the left inverse precedes plant. Right and left inverse are shown in Figures 1(a) and 1(b), respectively.

A direct adaptive inverse technique based on NLMS for control of discrete time linear plants to alleviate the adhocism in adaptive loop is proposed in [1]. However, the stability and convergence analysis is missing. A neural network based

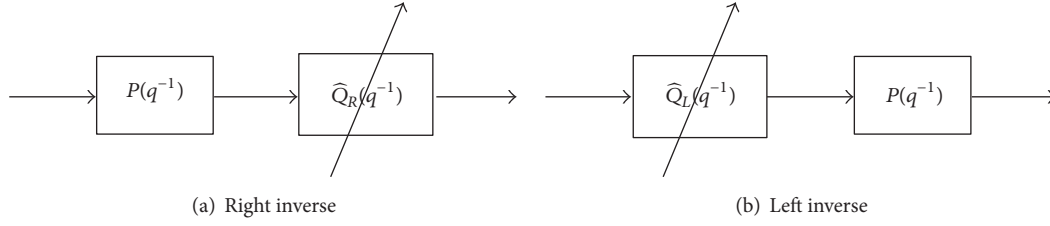


FIGURE 1

DAIC was then proposed in [28]. Reference [29] proposed a prefilter inversion system on the similar lines of DAIC. This DAIC structure was used for the adaptive control of electrohydraulic servo systems. The modified DIAC scheme was proposed for the prediction of the compression strength of concrete using neural network based on kernel ridge regression [30]. Another extension of the DAIC is proposed in [31]. Based on DAIC proposed in [1], a closed loop direct adaptive inverse control scheme is introduced in [32] that improves tracking, error convergence, and disturbance rejection properties of DAIC. These extensions and uses of DAIC motivate us to provide stability and convergence analysis of DAIC along with some applications to experimental systems. In this paper, parameter estimation for plant model and adaptive inverse controller, stability analysis, and error convergence for DAIC is discussed thoroughly. Further, simulation results in presence of disturbance are given in the paper. Laboratory scale experiments are presented to elaborate the performance of DAIC on physical plants. DAIC can be used for tracking of stable or stabilized, minimum or non-minimum phase discrete time linear plants. Little modification can also establish model reference adaptive tracking as well.

The rest of the paper is organized as follows. Section 2 presents problem statement. Section 3 discusses existing indirect adaptive inverse control (IAIC) schemes. Design of DAIC scheme is given in Section 4. Parameter estimation algorithms and stability analysis for DAIC are given in Section 5. Simulation results are presented in Section 6. Experimental results are described in Section 7. Conclusions are drawn in Section 8.

2. Problem Statement

Let us consider $P(q^{-1})$ as a discrete time stable or stabilized linear plant, which is given by

$$\begin{aligned} P(q^{-1}) &= q^{-d} \frac{B(q^{-1})}{A(q^{-1})}, \\ A(q^{-1}) &= 1 + a_1 q^{-1} + a_2 q^{-2} + \dots + a_n q^{-n}, \\ B(q^{-1}) &= b_0 + b_1 q^{-1} + b_2 q^{-2} + \dots + b_m q^{-m}, \end{aligned} \quad (1)$$

where q^{-1} is a back shift operator defined as $q^{-1}y(k) = y(k-1)$, k is a positive integer that represents discrete time instant, d is a positive integer that represents delay of the plant. n and m are positive integers, and $n \geq m$. $A(q^{-1})$ and

$B(q^{-1})$ are relatively coprime polynomials. We also assume that the plant may be non-minimum phase; that is, inverse of plant is unstable. Let $r(k)$, $y_d(k)$, and $y(k)$ be the reference input, desired output, and plant output, respectively. Further, it is assumed that parameters of the plant are unknown or slowly time varying compared to the adaptation algorithm. The objective is to design a controller such that $y(k)$ tracks $y_d(k)$; that is,

$$\lim_{k \rightarrow \infty} (e_{\text{ref}}(k))^2 = \lim_{k \rightarrow \infty} (y_d(k) - y(k))^2 \rightarrow 0, \quad (2)$$

where $y_d(k) = r(k-L)$, L is a positive integer that represents known delay, and $e_{\text{ref}}(k)$ is error at instant k .

3. Overview of Existing IAIC Schemes

Control scheme for linear Single Input Single Output (SISO) plants that uses IAIC proposed in [4] is shown in Figure 2.

Right inverse $\widehat{Q}_R(q^{-1})$ is estimated using inverse model identification. $\widehat{Q}_R(q^{-1})$ is then copied into feed-forward path of plant, that is, $\widehat{Q}_{\text{Rcopy}}(q^{-1})$. L is considered zero in Figure 2 for controlling minimum phase plants [4]. $e_r(k)$ is used to adapt the weights of adaptive filter, where $e_r(k) = y_i(k) - u(k-L)$. $y_i(k)$ is output of $\widehat{Q}_R(q^{-1})$. When $e_r(k) \rightarrow 0$ then $e_{\text{ref}}(k)$ will approach zero as well [2]. In this case, $e_{\text{ref}}(k)$ is given by

$$e_{\text{ref}}(k) = [q^{-L} - \widehat{Q}_{\text{Rcopy}}(q^{-1})P(q^{-1})]r(k). \quad (3)$$

Due to commutability of linear filters

$$\widehat{Q}_{\text{Rcopy}}(q^{-1})P(q^{-1}) \cong P(q^{-1})\widehat{Q}_R(q^{-1}). \quad (4)$$

AIC based on linear and nonlinear adaptive filtering discussed in [5] is shown in Figure 3. $M(q^{-1})$ is filter with desired response. For structure in Figure 3, (3) can be rewritten as

$$e_{\text{ref}}(k) = [M(q^{-1}) - \widehat{Q}_{\text{Rcopy}}(q^{-1})P(q^{-1})]r(k). \quad (5)$$

Indirect adaptive tracking schemes discussed above prove good for stable or stabilized plant. IAIC schemes estimate $\widehat{Q}_R(q^{-1})$ and then it is copied in feed-forward path as left inverse $\widehat{Q}_L(q^{-1})$. There are situations in which $\widehat{Q}_{\text{Rcopy}}(q^{-1})$ may not be equal to $\widehat{Q}_L(q^{-1})$ because of nonlinearities in the plant. So, the use of $\widehat{Q}_{\text{Rcopy}}(q^{-1})$ instead of $\widehat{Q}_L(q^{-1})$ in such situations will not accomplish tracking [2].

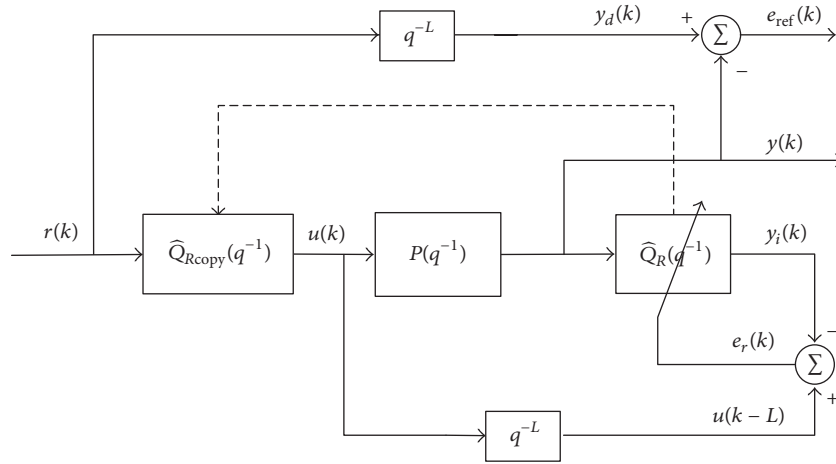


FIGURE 2: Indirect control scheme for non-minimum phase plants [4].

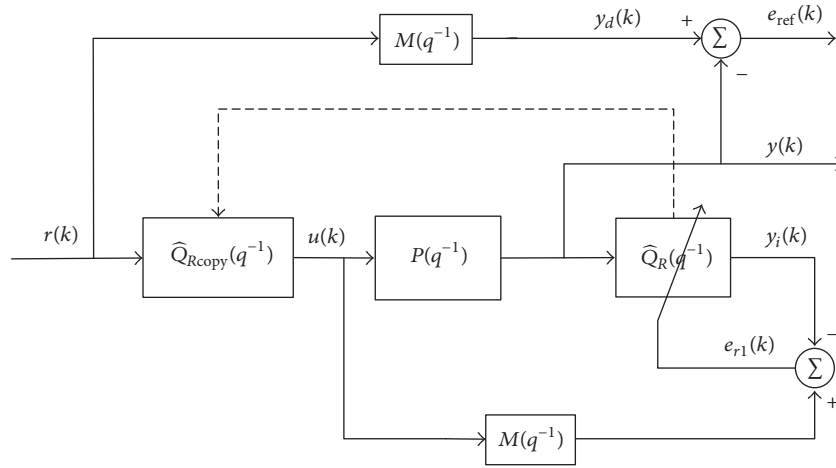


FIGURE 3: Indirect AIC structure for linear SISO plants [5].

4. Design of DAIC

DAIC structure for controlling stable or stabilized minimum/non-minimum phase linear SISO plants [1] is shown in Figure 3. In this structure approximate inverse system $\hat{Q}_L(q^{-1})$ is directly estimated. Control input to plant $u(k)$ is synthesized by

$$u(k) = \hat{Q}_L(q^{-1}) r(k). \quad (6)$$

The online estimation of $\hat{Q}_L(q^{-1})$ is accomplished using three steps given below:

- (1) Adaptive plant model $\hat{P}(q^{-1})$ is obtained using NLMS adaptive filter.
- (2) The mismatch error $e_{ref}(k)$ between desired response $y_d(k)$ and plant output $y(k)$ is propagated through plant model $\hat{P}(q^{-1})$.
- (3) Output obtained from the second step $e_f(k)$ is used to adapt the weights of controller which is also an NLMS adaptive filter.

In this algorithm, the parameters of the controller $\hat{Q}_L(q^{-1})$ are estimated directly. This means $\hat{Q}_{Rcopy}(q^{-1})$ is not used. Plant is preceded by the controller. There is no direct feedback from the plant output. Control scheme is not strictly feed-forward because controller weights are updated such that it contains information about the plant output and reference input. As shown in Figure 4, we identify the plant as a moving average system (i.e., the plant is approximated by an adaptive Finite Impulse Response (FIR) filter). Then for estimation of the adaptive inverse controller parameters, $e_f(k)$ is used as an error signal, where

$$e_f(k) = \hat{P}(q^{-1}) e_{ref}(k). \quad (7)$$

DAIC is much simpler as compared to methods presented in [9, 10]. We use NLMS algorithm to estimate the plant and the adaptive inverse controller parameters, whereas Jacobian matrices of network are calculated using dual subroutine and Back Propagation Through Model (BPTM) algorithm is used to adapt plant model and constrained controller in [9, 10]. The

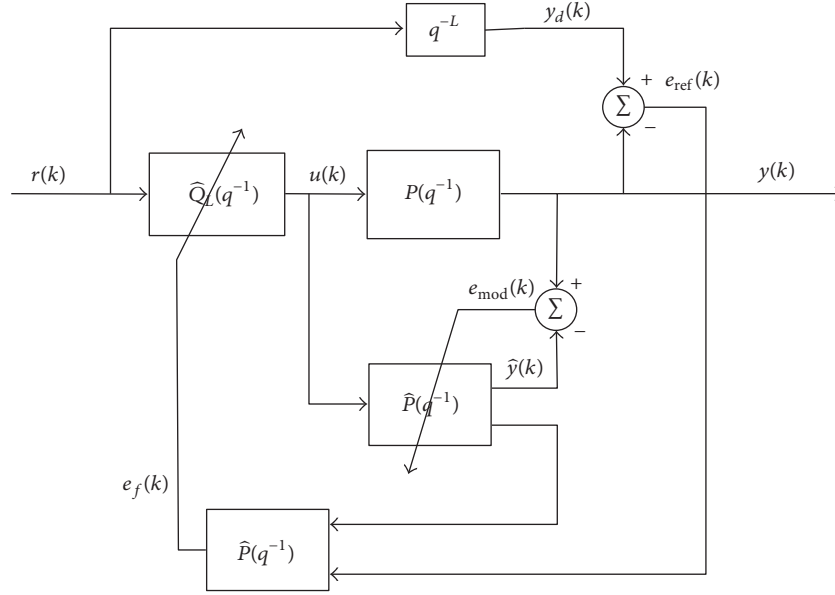


FIGURE 4: Direct adaptive inverse control scheme.

details of parameter estimation and stability analysis of the proposed DAIC are given in Section 5.

Mean square error (MSE) between desired output and plant output for non-minimum phase plants can be made small by incorporating the delay q^{-L} . Since $\widehat{Q}_L(q^{-1})$ is used as feed-forward controller for $P(q^{-1})$, this gives

$$\widehat{Q}_L(q^{-1})P(q^{-1}) \cong q^{-L}. \quad (8)$$

The parameter L is generally kept small for minimum phase and large for non-minimum phase plants. In simulations, we have observed that choosing $L \cong (\nu + d + m)/2$ gives good tracking in non-minimum phase systems, where ν is the order of $\widehat{Q}_L(q^{-1})$.

Using $\widehat{Q}_R(q^{-1})$ for $\widehat{Q}_L(q^{-1})$ in IAIC introduces at least one step delay in the controller parameters. DAIC dwindles the adhocism of adaptive loop by directly incorporating an adaptive controller $\widehat{Q}_L(q^{-1})$ in feed-forward loop. Since plant model is identified first, DAIC is less sensitive to plant uncertainties and variations. Further, mild nonlinearities at the output of plant may be learnt by $\widehat{Q}_R(q^{-1})$ in IAIC causing deviation from desired signal. Using $\widehat{Q}_{Rcopy}(q^{-1})$ as left inverse may not then accomplish tracking as commutability is lost. DAIC rectifies this deficiency. In DAIC

$$\lim_{k \rightarrow \infty} (e_{\text{ref}}(k))^2 \rightarrow 0 \quad (9)$$

provided

$$\lim_{k \rightarrow \infty} (e_{\text{mod}}(k))^2 \rightarrow 0, \quad (10)$$

where $e_{\text{mod}}(k) = y(k) - \widehat{y}(k)$ and $\widehat{y}(k)$ is output of estimated plant $\widehat{P}(q^{-1})$ given by

$$\widehat{y}(k) = \theta(k) \psi^T(k), \quad (11)$$

where $\theta(k)$ is a parameter vector for $\widehat{P}(q^{-1})$ defined as $\theta(k) = [\beta_0, \beta_1, \dots, \beta_M]$ and $\psi(k)$ is regression vector defined as $\psi(k) = [u(k), u(k-1), \dots, u(k-M)]$.

5. Development of Estimation Algorithm for SISO Systems

In this section, estimation algorithms for linear SISO systems are developed. Parameter estimation is developed based on NLMS algorithm.

5.1. Parameter Updating for Plant Model. The parameters of the plant model $\widehat{P}(q^{-1})$ are obtained by minimizing the performance index ϕ defined by

$$\phi = \frac{1}{2} e_{\text{mod}}^2(k). \quad (12)$$

Parameters of the plant model should be updated in the direction of negative gradient as

$$\theta(k+1) = \theta(k) - \mu_1 \frac{\partial \phi}{\partial \theta}, \quad (13)$$

where μ_1 is the learning rate.

NLMS is self-normalized version of LMS. Convergence of NLMS is faster than LMS [33]. Due to normalization of input, NLMS is less sensitive to colored input signal and has more stable behavior than LMS [34]. The parameter update equation for the plant model based on NLMS is given below.

$$\theta(k+1) = \begin{cases} \theta(k) & \text{if } \psi(k) \psi^T(k) = 0, \\ \theta(k) + \mu_1 e_{\text{mod}}(k) \frac{\psi(k)}{\psi(k) \psi^T(k)} & \text{if } \psi(k) \psi^T(k) \neq 0. \end{cases} \quad (14)$$

Stability Analysis and Error Convergence. For $\psi(k)\psi^T(k) \neq 0$, (14) gives

$$\theta(k+1) = \theta(k) + \mu_1 \frac{\psi(k) e_{\text{mod}}(k)}{\psi(k)\psi^T(k)}, \quad (15)$$

$$\theta(k+1) = \theta(k) + \mu_1 \frac{\psi(k) [y(k) - \theta(k)\psi^T(k)]}{\psi(k)\psi^T(k)}, \quad (16)$$

$$\theta(k) = \frac{q^{-1}\mu_1}{1 - (1 - \mu_1)q^{-1}} \frac{\psi(k) y(k)}{\psi(k)\psi^T(k)}. \quad (17)$$

$[1 - (1 - \mu_1)q^{-1}]$ will be a Schur polynomial if $0 < \mu_1 < 1$. This means that (17) is stable; that is, $\theta(k)$ will be bounded if $\mu_1(\psi(k)y(k)/\psi(k)\psi^T(k))$ is bounded. The term $\mu_1(\psi(k)y(k)/\psi(k)\psi^T(k))$ will remain bounded if $\psi(k)\psi^T(k) \neq 0$. Now convergence of error will be proved. Let

$$e_{\text{mod}}(k) = y(k) - \hat{y}(k), \quad (18)$$

$$e_{\text{mod}}(k) = y(k) - \theta(k)\psi^T(k), \quad (19)$$

$$e_{\text{mod}}(k+1) = y(k) - \theta(k+1)\psi^T(k). \quad (20)$$

Subtracting (19) from (20)

$$e_{\text{mod}}(k+1) - e_{\text{mod}}(k) = -\Delta\theta(k)\psi^T(k), \quad (21)$$

where $\Delta\theta(k) = \theta(k+1) - \theta(k)$. Substituting value of $\Delta\theta(k)$ from (15) in (20), we get

$$e_{\text{mod}}(k+1) - e_{\text{mod}}(k) = -\mu_1 \frac{\psi(k) e_{\text{mod}}(k)}{\psi(k)\psi^T(k)} \psi^T(k), \quad (22)$$

$$e_{\text{mod}}(k+1) = (1 - \mu_1) e_{\text{mod}}(k) \quad (23)$$

$$= (1 - \mu_1)^2 e_{\text{mod}}(k-1) \quad (24)$$

$$= (1 - \mu_1)^3 e_{\text{mod}}(k-2) \quad (25)$$

$$\vdots \quad (26)$$

$$e_{\text{mod}}(k+1) = (1 - \mu_1)^{k+1} e_{\text{mod}}(0). \quad (27)$$

Taking limits on both sides of (27)

$$\lim_{k \rightarrow \infty} e_{\text{mod}}(k+1) = \lim_{k \rightarrow \infty} (1 - \mu_1)^{k+1} e_{\text{mod}}(0) = 0. \quad (28)$$

Convergence of $e_{\text{mod}}(k+1)$ will be satisfied if $0 < \mu_1 < 1$.

5.2. Parameter Updating for Controller Parameters. Parameters of controller are obtained by minimizing the performance index J given by

$$J = \frac{1}{2} e_{\text{ref}}^2(k), \quad (29)$$

$$e_{\text{ref}}(k) = y_d(k) - y(k),$$

$$e_{\text{ref}}(k) = r(k-L) - y(k).$$

Weights of the adaptive controller should be updated in the direction of negative gradient as

$$\omega(k+1) = \omega(k) - \mu_2 \frac{\partial J}{\partial \omega}, \quad (30)$$

where μ_2 is learning rate and $\omega(k)$ is the parameter vector for controller $\widehat{Q}_L(q^{-1})$ defined as $\omega(k) = [\alpha_0, \alpha_1, \dots, \alpha_N]$. Now finding partial derivative

$$\frac{\partial J}{\partial \omega} = \frac{\partial}{\partial \omega} \left(\frac{1}{2} e_{\text{ref}}^2(k) \right), \quad (31)$$

$$\frac{\partial J}{\partial \omega} = -e_{\text{ref}}(k) \frac{\partial}{\partial \omega} \left(P(q^{-1}) u(k) \right), \quad (32)$$

$$\frac{\partial J}{\partial \omega} = -e_{\text{ref}}(k) P(q^{-1}) \varphi(k), \quad (33)$$

where $\varphi(k)$ is regression vector defined as $\varphi(k) = [r(k), r(k-1), \dots, r(k-N)]$. Now final parameter update equation for controller can be obtained by substituting (33) in (30)

$$\omega(k+1) = \omega(k) + \mu_2 P(q^{-1}) e_{\text{ref}}(k) \varphi(k). \quad (34)$$

We can replace $P(q^{-1})$ in (34) by its adaptive model $\widehat{P}(q^{-1})$ because it is shown in Section 5.1 that as $k \rightarrow \infty$ then $e_{\text{mod}}(k+1) \rightarrow 0$. Therefore, (34) becomes

$$\omega(k+1) = \omega(k) + \mu_2 e_f(k) \varphi(k), \quad (35)$$

where

$$e_f(k) = \widehat{P}(q^{-1}) e_{\text{ref}}(k). \quad (36)$$

Since NLMS is used, weight updating for controller is given by

$$\omega(k+1) = \begin{cases} \omega(k) & \text{if } \varphi(k)\varphi^T(k) = 0, \\ \omega(k) + \mu_2 e_f(k) \frac{\varphi(k)}{\varphi(k)\varphi^T(k)} & \text{if } \varphi(k)\varphi^T(k) \neq 0. \end{cases} \quad (37)$$

Stability Analysis and Error Convergence for Controller Parameters. Now sufficient conditions on μ_2 are obtained to ensure stability of DAIC. $e_f(k)$ can be written as

$$e_f(k) = \theta(k) E_{\text{ref}}^T(k), \quad (38)$$

$$e_f(k) = \theta(k) [e_{\text{ref}}(k), e_{\text{ref}}(k-1), \dots, e_{\text{ref}}(k-M)]^T, \quad (39)$$

$$e_f(k) = -\beta_0 y(k) - \beta_1 y(k-1) - \dots - \beta_M y(k-M) + \beta_0 r(k-L) + \beta_1 r(k-L-1) + \dots + \beta_M r(k-L-M). \quad (40)$$

Substituting (40) in (37)

$$\begin{aligned} \omega(k+1) = & \omega(k) - \mu_2 \frac{\varphi(k)}{\varphi(k)\varphi^T(k)} [\beta_0 y(k) \\ & + \beta_1 y(k-1) + \dots + \beta_M y(k-M)] + \rho(k), \end{aligned} \quad (41)$$

where

$$\begin{aligned} \rho(k) = & \mu_2 \frac{\varphi(k)}{\varphi(k)\varphi^T(k)} [\beta_0 r(k-L) + \beta_1 r(k-L-1) \\ & + \dots + \beta_M r(k-L-M)]. \end{aligned} \quad (42)$$

Now, (41) can be written as

$$\begin{aligned} \omega(k+1) = & \omega(k) + \rho(k) - \mu_2 \\ & \cdot \frac{\varphi(k)}{\varphi(k)\varphi^T(k)} [\beta_0 P(q^{-1})\omega(k)\varphi^T(k) \\ & + \beta_1 P(q^{-1})\omega(k-1)\varphi^T(k) + \dots \\ & + \beta_M P(q^{-1})\omega(k-M)\varphi^T(k)]. \end{aligned} \quad (43)$$

Grouping and rearranging the terms of (43), we get

$$\begin{aligned} \omega(k+1) = & \rho(k) + (1 - \mu_2 \beta_0 P(q^{-1}))\omega(k) \\ & - \mu_2 P(q^{-1})[\beta_1 \omega(k-1) + \dots + \beta_M \omega(k-M)]. \end{aligned} \quad (44)$$

Using (44), the controller parameter vector can be written as

$$\omega(k) = \frac{\rho(k-1)}{D(q^{-1})}, \quad (45)$$

where

$$\begin{aligned} D(q^{-1}) = & S(q^{-1}) + S_1(q^{-1}), \\ S(q^{-1}) = & 1 - (1 - \mu_2 \beta_0 P(q^{-1}))q^{-1}, \\ S_1(q^{-1}) = & \mu_2 P(q^{-1})[\beta_1 q^{-2} + \dots + \beta_M q^{-M-1}]. \end{aligned} \quad (46)$$

$S(q^{-1})$ will be Schur polynomial if

$$\begin{aligned} |1 - \mu_2 \beta_0 P(q^{-1})| & < 1, \\ -1 & < 1 - \mu_2 \beta_0 P(q^{-1}) < 1, \\ 0 & < \mu_2 \beta_0 P(q^{-1}) < 2. \end{aligned} \quad (47)$$

To avoid overcorrection, range is given by

$$0 < \mu_2 \beta_0 P(q^{-1}) < 1. \quad (48)$$

Further, $D(q^{-1})$ is Schur and will remain stable if it is shown that

$$\mu_2 |P(q^{-1})| \sum_{i=1}^M |\beta_i| < |1 - \mu_2 \beta_0 P(q^{-1})|. \quad (49)$$

Using triangle difference inequality [35] and simplifying, we get

$$\begin{aligned} \mu_2 & < \frac{1}{|P(q^{-1})| \sum_{i=0}^M |\beta_i|}, \\ \mu_2 & < \frac{1}{|P(q^{-1})| |\theta(k)|}. \end{aligned} \quad (50)$$

Controller output will remain bounded if μ_2 is chosen such that $0 < \mu_2 < 1/|P(q^{-1})||\theta(k)|$. $|P(q^{-1})|$ and $|\theta(k)|$ can be found online and incorporated as learning rate, but we choose small learning rate for controller in order to avoid instability. To be more conservative if $\mu_2 > 1$, then we use $0 < \mu_2 < 1$. Now convergence of error will be proved. Let

$$e_{\text{ref}}(k) = y_d(k) - y(k), \quad (51)$$

$$e_{\text{ref}}(k) = r(k-L) - P(q^{-1})\omega(k)\varphi^T(k), \quad (52)$$

$$e_{\text{ref}}(k+1) = r(k-L) - P(q^{-1})\omega(k+1)\varphi^T(k). \quad (53)$$

Subtracting (52) from (53), we obtain

$$\begin{aligned} e_{\text{ref}}(k+1) - e_{\text{ref}}(k) = & -P(q^{-1})(\omega(k+1) - \omega(k))\varphi^T(k) \\ = & -P(q^{-1})\Delta\omega(k)\varphi^T(k). \end{aligned} \quad (54)$$

$$= -P(q^{-1})\Delta\omega(k)\varphi^T(k). \quad (55)$$

Using (35) and (55), the following relationship is obtained

$$\begin{aligned} e_{\text{ref}}(k+1) = & (1 - \mu_2 P(q^{-1})\hat{P}(q^{-1}))e_{\text{ref}}(k)\varphi(k)\varphi^T(k). \end{aligned} \quad (56)$$

Equation (56) will be asymptotically stable; that is, $\lim_{k \rightarrow \infty} e_{\text{ref}}(k+1) = 0$, if μ_2 is chosen such that $0 < \mu_2 < 1/|P(q^{-1})||\theta(k)|$. NLMS adaptive filters are inherently stable and are used for plant estimating the parameters of the plant and controller. Output of controller remains bounded as long as μ_2 is kept small. Since a bounded input is applied to the plant and stable adaptive filters are used in conjunction with the stabilized plant, the controller output remains bounded.

6. Simulation Results

Computer simulations of DAIC and IAIC schemes are presented to show effectiveness of DAIC. Two linear non-minimum phase systems are chosen, one without disturbance and other with disturbance.

6.1. Example 1. A disturbance free discrete time non-minimum phase linear plant is chosen having

$$y(k) = q^{-1} \frac{1 + 1.2q^{-1}}{1 + 0.5q^{-1} + 0.1q^{-2}} u(k). \quad (57)$$

This is a stable non-minimum phase plant having zero at -1.2000 and poles at $-0.2500 \pm 0.1936i$. In this example, we

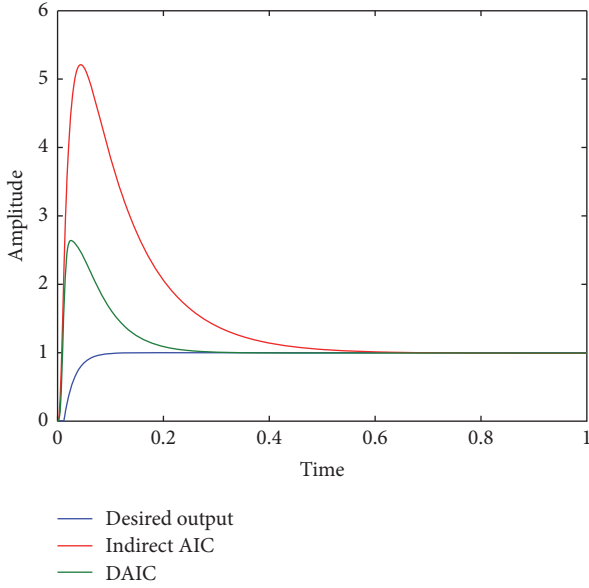


FIGURE 5: Tracking desired output: first 1 sec.

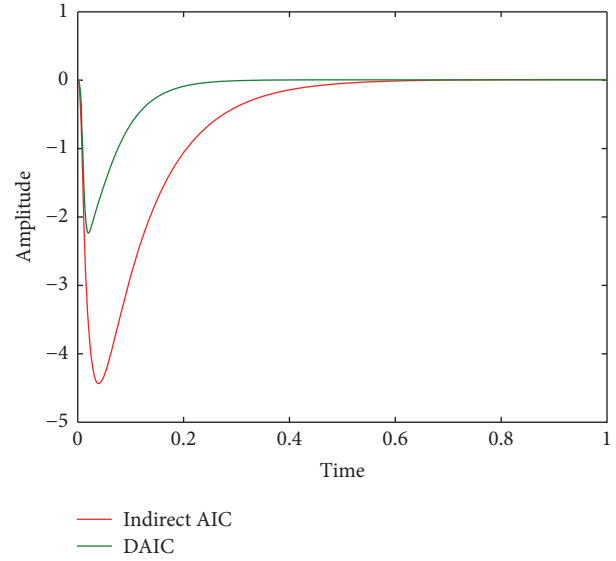


FIGURE 7: Tracking error: first 1 sec.

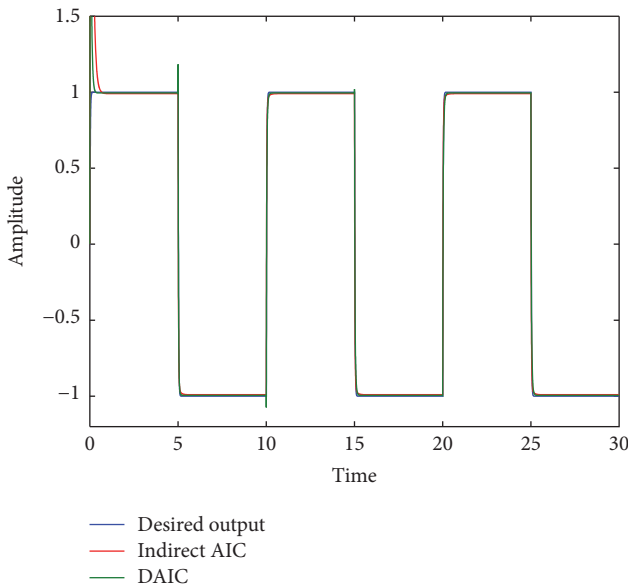


FIGURE 6: Tracking desired output: amplitude $-1.2 \sim 1.5$.

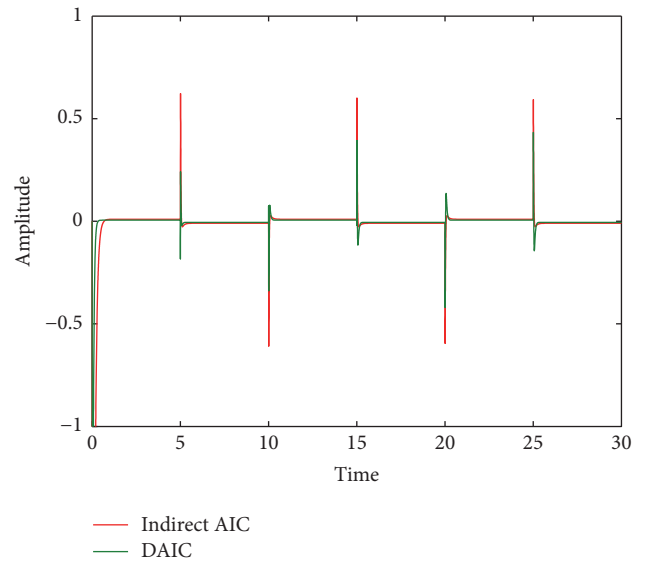


FIGURE 8: Tracking error: amplitude $-1 \sim 1$.

choose $\mu_1 = 0.1$ and $\mu_2 = 0.01$. Similarly learning rate for IAIC is chosen as 0.01. Orders of $\widehat{Q}_R(q^{-1})$ and $\widehat{Q}_L(q^{-1})$ are chosen as 10. Sampling time is chosen as 0.001 sec. Simulation results are depicted in Figures 5–12. Zoomed preview for desired output tracking is shown in Figures 5 and 6. Plant output in DAIC has less overshoot and converges to desired output quickly compared to IAIC. Tracking error is shown in Figures 7 and 8. Tracking error has less amplitude and converges to zero faster in DAIC compared to IAIC. MSE for IAIC and DAIC is shown in Figures 9 and 10. MSE is less for DAIC compared to IAIC. Control input is shown in Figure 11.

Control input for DAIC converges quickly compared to IAIC. Model identification error $e_{\text{mod}}(k)$ in DAIC converges to zero very quickly and is shown in Figure 12.

6.2. Example 2. A disturbance $n(k)$ is added to discrete time non-minimum phase linear plant. Now, plant output can be written as

$$y(k) = q^{-1} \cdot \frac{1 - 3q^{-1} + 3.5q^{-2}}{1 + 0.05q^{-2} + 0.05q^{-3} + 0.02q^{-4}} (u(k) + n(k)). \quad (58)$$

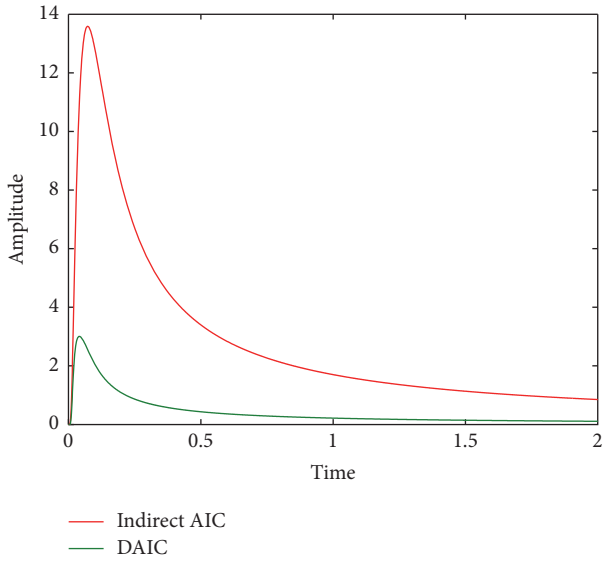


FIGURE 9: Mean square error: first 2 sec.

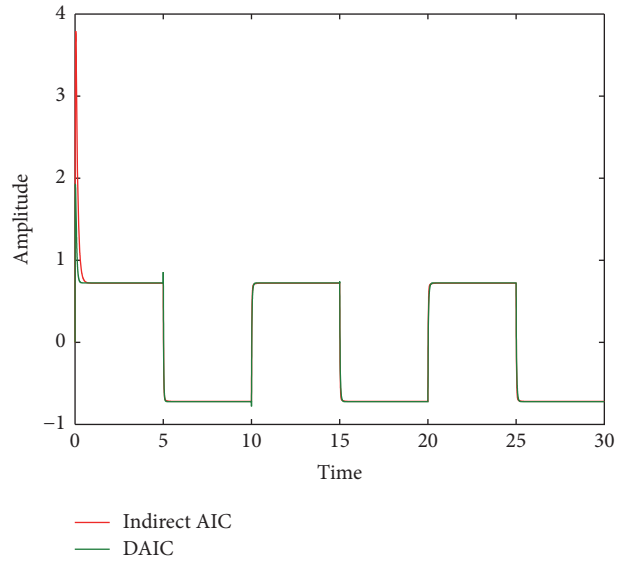


FIGURE 11: Control input.

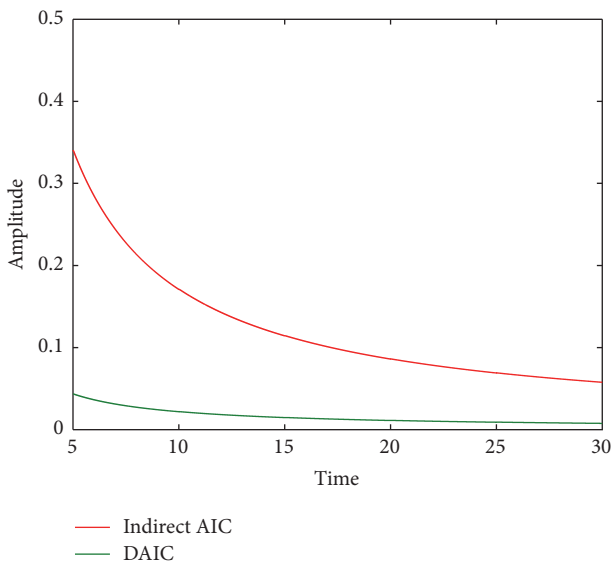


FIGURE 10: Mean square error: amplitude 0~0.5.

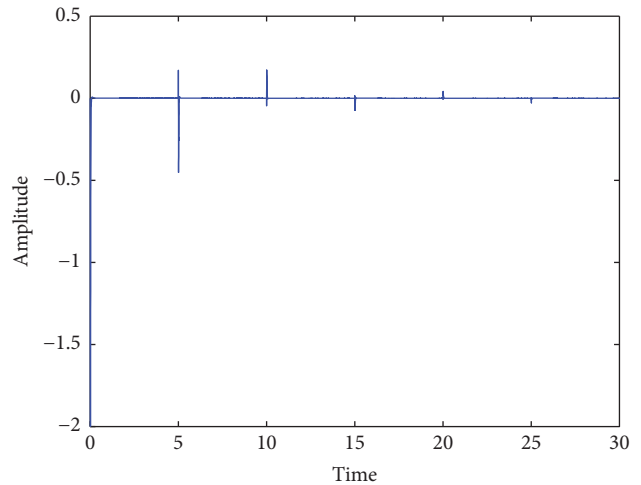


FIGURE 12: Model identification error in DAIC.

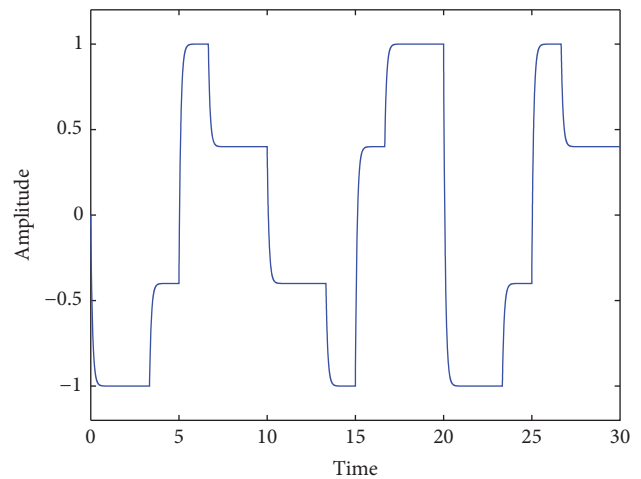


FIGURE 13: Disturbance.

This is a stable non-minimum phase plant having zeros at $1.5000 \pm 1.1180i$ and poles at $0.2500 \pm 0.3708i$ and $-0.2500 \pm 0.1936i$. Here, $n(k)$ is disturbance added to the plant and is shown in Figure 13. In this example, we choose $\mu_1 = 0.01$ and $\mu_2 = 0.01$. Similarly learning rate for IAIC is chosen 0.01. Sampling time is chosen 0.001 sec. Simulation results are shown in Figures 14–16.

Control input is depicted in Figure 14. Control input for DAIC is synthesized such that plant tracks desired output even in the presence of disturbance. Desired output tracking for DAIC is shown in Figure 15. Plant output not only converges to desired output but good disturbance rejection is

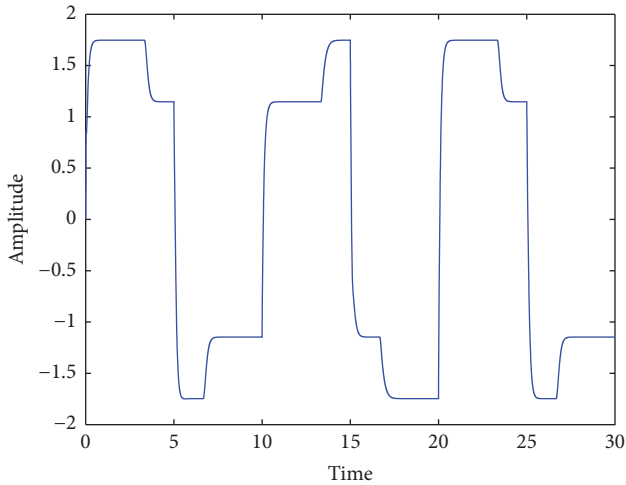
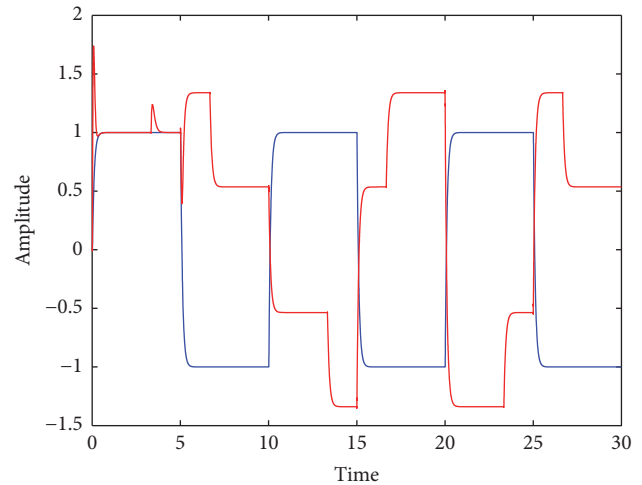
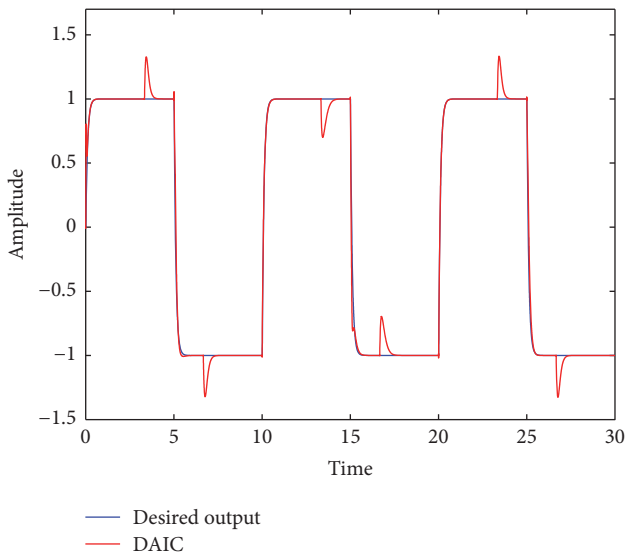


FIGURE 14: Control input in DAIC.



— Desired output
— DAIC

FIGURE 16: Desired output tracking in IAIC.



— Desired output
— DAIC

FIGURE 15: Desired output tracking in DAIC.

also achieved in DAIC. Plant output in IAIC does not follow desired output and is depicted in Figure 16.

7. Experimental Results

The proposed scheme is implemented on laboratory scale temperature control of a heating process, speed, and position tracking of direct current motor. The temperature control of a process is a non-minimum phase system while the speed and position control of a DC motor is a minimum phase system. To accomplish the adaptive tracking, the proposed DAIC does not require a prior information of the system phase. In these experiments, a standard IBM PC-type Pentium IV is used for the computation in real time. Data acquisition is accomplished by National Instrument card NI-6024E and the

controller is implemented in SIMULINK real-time windows target environment. The computations are performed in floating-point format and the sampling interval is selected as 0.001 sec.

7.1. Temperature Control of Heating Process. In this experiment, we use process trainer PT326 manufactured by Feedback Ltd., UK. This process is composed of a blower, a heating grid, tube, and temperature sensor (bead thermistor). A variable power supply provides power to the heater. This power can be controlled by initiating an appropriate controlling signal from the computer. The process can be considered as a second-order time delay system. This is a non-minimum phase system. Input of the process is power and output is the temperature of air at some desired location in the process tube. This is a time delay system. In this experiment 10 parameters are selected for the plant estimation and 30 for inverse of the plant. The proposed DAIC does not provide a procedure for the selection of optimal number of the plant and the controller parameters. However, the control input, plant output, and the tracking error remain bounded for any selected number of these parameters. Experimental and simulations studies show that large number of parameters achieves better tracking at the cost of computational burden. Figure 17 shows that output (temperature) of the process converges to the desired temperature quickly. Control input to the plant is smooth and bounded. Control signal is depicted in Figure 18.

7.2. Speed and Position Control of DC Motor. In this experiment, we use modular servo system (MSS) manufactured by Feedback Ltd., UK. All the modules used in this experiment are parts of MSS. MT150F is a module containing DC motor and tacho-generator. PS150E is the power supply and SA-150D is a servo amplifier. Control input to the servo amplifier

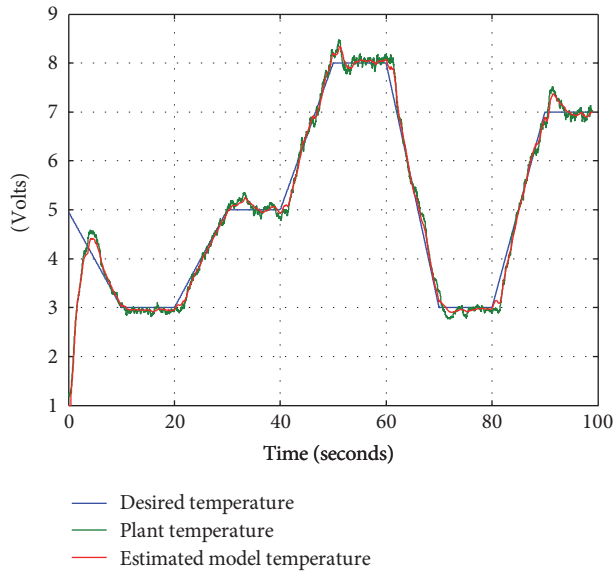


FIGURE 17: Temperature control of heating process.

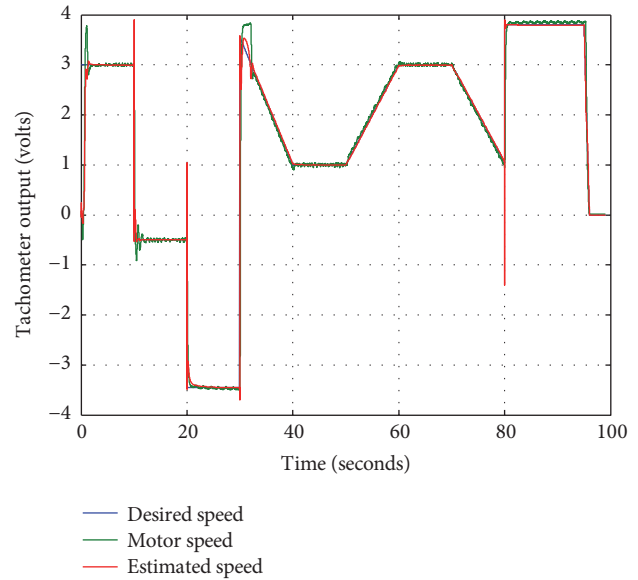


FIGURE 19: Speed control of DC motor.

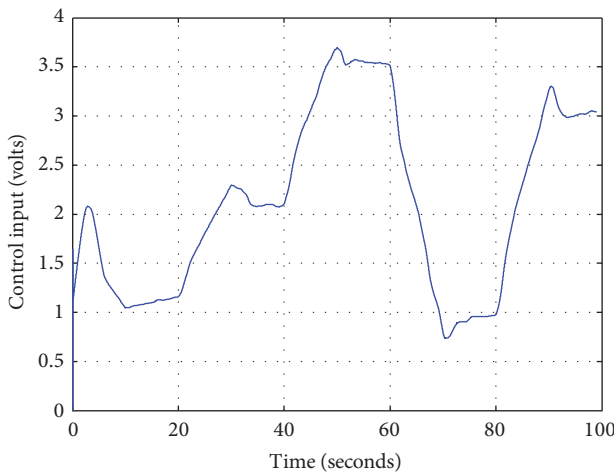


FIGURE 18: Control input to heating process.

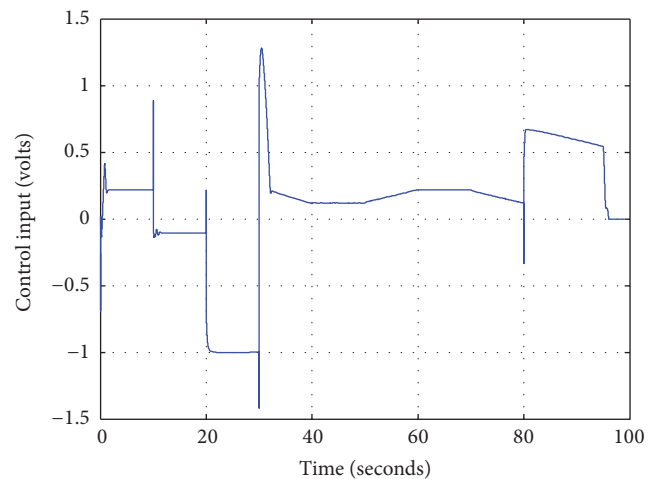


FIGURE 20: Control input to speed control system.

is through a preamplifier PA150C. In the speed control experiment the speed signal is measured by the tacho-generator. This generator measures the speed in ± 10 volts corresponding to ± 1800 rpm. Figure 19 show that the motor speed converges to the desired speed. There are some large spikes in the estimated speed but the actual speed of the motor is quite smooth. It can be observed that the speed control in the vicinity of the dead zone is also accomplished. Speed control of such systems at low speed is known be a difficult problem. The control input is shown in Figure 20, which is bounded and smooth. Module OPI50K is used to measure the position of the motor shaft. This module measures the shaft position in the range 0–10 volts corresponding to 0–360 degrees. Figure 21 shows that the motor shaft follows the desired position smoothly and quickly. Figure 22 shows that control input is smooth and bounded.

8. Conclusion

Stability and convergence of DAIC are discussed in detail. Simulation results show that DAIC performs better than IAIC in terms of mean square tracking error and disturbance rejection. The stability of the closed loop is discussed in detail. The convergence of the error to zero and the boundedness of the controller parameters are proved. However, an algorithm to determine the optimal number of the estimated plant and the controller parameters is needed. Laboratory scale experiments show that DAIC accomplishes tracking of the plant output to the desired smooth trajectory. The synthesized control input in simulations and experiments remains smooth and bounded.

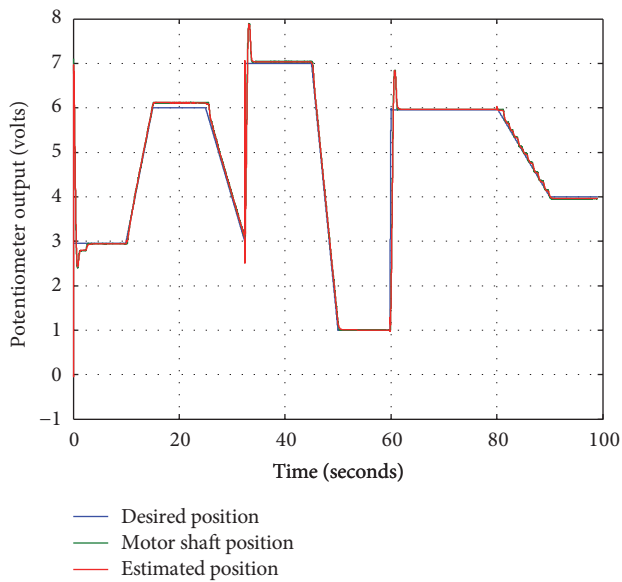


FIGURE 21: Position control of Dc motor.

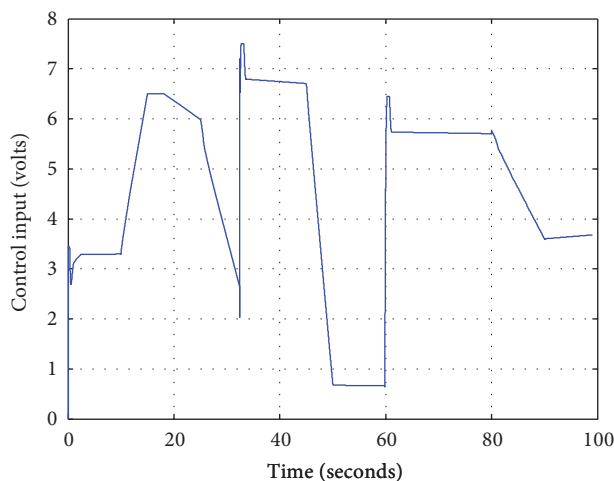


FIGURE 22: Control input to position control system.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] M. Shafiq and M. A. Shafiq, "Direct adaptive inverse control," *IEICE Electronics Express*, vol. 6, no. 5, pp. 223–229, 2009.
- [2] B. Widrow and E. Walach Prentice-Hall, Englewood Cliffs, NJ, USA, 1st edition, 1995, Adaptive inverse control: a signal processing approach.
- [3] K. J. Astrom and B. Wittenmark, *Adaptive Control*, Addison-Wesley Longman publishing Co. Inc, Boston, MA, USA, 2nd edition, 1994.
- [4] Y. D. Landau, *Adaptive Control: The Model Reference Approach*, Marcel Dekker Inc., New — York, NY, USA, 1st edition, 1979.
- [5] B. Widrow and M. Bilello, "Adaptive Inverse Control," in *Proceedings of the International symposium on Intelligent Control*, pp. 1–6, IEEE, Chicago, IL, USA, 1993.
- [6] B. Widrow and G. L. Plett, "Adaptive inverse control based on linear and nonlinear adaptive filtering," in *Proceedings of the International workshop on neural networks for Identification, Control, Robotics, and Signal Image processing*, pp. 30–38, IEEE, Venice, Italy, 1996.
- [7] B. Widrow and G. L. Plett, "Nonlinear adaptive inverse control," in *Proceedings of the Decision and Control*, pp. 1032–1037, IEEE, San Diego, CA, USA.
- [8] N. A. Hizal, "Improved adaptive model control," in *ARI—An International Journal for Physical and Engineering Sciences*, vol. 51, pp. 181–190, 3 edition, 1999.
- [9] G. L. Plett, "Adaptive inverse control of linear and nonlinear systems using dynamic neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 14, no. 2, pp. 360–376, 2003.
- [10] G. L. Plett, "Adaptive inverse control of unmodeled stable SISO and MIMO linear systems," *International Journal of Adaptive Control and Signal Processing*, vol. 16, no. 4, pp. 243–272, 2002.
- [11] M. Shafiq, F. M. AL-Sunni, and S. O. Farooq, "Adaptive control of nonlinear hammerstein model using NLMS filter," in *Proceedings of the International Conference on Electronics, Circuits and Systems*, vol. 2, pp. 439–442, IEEE, Sharjah, UAE, 2003.
- [12] R. Salman, "Neural networks of adaptive inverse control systems," *Applied Mathematics and Computation*, vol. 163, no. 2, pp. 931–939, 2005.
- [13] L. Ming, Y. Cheng, and S. Yu, "An improved nonlinear adaptive inverse control systems based on filtered- ϵ LMS algorithm," in *Proceedings of the Chinese control conference*, pp. 101–105, IEEE, Kunming, China, 2008.
- [14] B. D. O. Anderson and A. Dehghani, "Challenges of adaptive control-past, permanent and future," *Annual Reviews in Control*, vol. 32, no. 2, pp. 123–135, 2008.
- [15] A. M. Karshenas, M. W. Dunnigan, and B. W. Williams, "Adaptive inverse control algorithm for shock testing," *IEE Proceedings Control Theory and Applications*, vol. 147, no. 3, pp. 267–276, 2000.
- [16] F. M. Dias and AM. Mota, "Direct inverse control of a kiln," in *Proceedings of the Direct inverse control of a kiln*, pp. 336–341, 2000.
- [17] M. Shafiq, "Internal model control structure using adaptive inverse control strategy," *ISA Transactions*, vol. 44, no. 3, pp. 353–362, 2005.
- [18] G. Du, X. Zhan, W. Zhang, and S. Zhong, "Improved filtered- ϵ adaptive inverse control and its application on nonlinear ship maneuvering," *Journal of Systems Engineering and Electronics*, vol. 17, no. 4, pp. 788–792, 2006.
- [19] X. Wang, T. Shen, and W. Wang, "An approach for echo cancellation system based on improved," in *Proceedings of the International Conference on Wireless Communications, Networking and Mobile Computing*, pp. 2853–2856, IEEE, Shanghai, China, 2007.
- [20] B.-S. Ryu, J.-K. Lee, J. Kim, and C.-W. Lee, "The performance of an adaptive noise canceller with DSP processor," in *Proceedings of the 40th Southeastern Symposium on System Theory*, pp. 42–45, IEEE, New Orleans, LA, USA, March 2008.
- [21] S. Shubao, S. Siyang, C. Nan, and X. Minglong, "Structure and control strategy for a piezoelectric inchworm actuator equipped with MEMS ridges," *Sensors and Actuators A: Physical*, vol. 264, no. 9, pp. 40–50, 2017.
- [22] S. Gang, L. Xiang, Z. Zhencai, T. Yu, Z. Weidong, and L. Shanzeng, "Acceleration tracking control combining adaptive

- control and off-line compensators for six-degree-of-freedom electro-hydraulic shaking tables,” *ISA Transactions*, vol. 70, no. 9, pp. 322–337, 2017.
- [23] K. J. Astrom and B. Wittenmark, *Computer controlled systems: theory and design*, Prentice-Hall, 3rd edition, 1996.
- [24] K. J. Astrom, P. Hagander, and J. Sternby, “Zeros of sampled systems,” in *Proceedings of the IEEE Conference on Decision and Control including the Symposium on Adaptive Processes*, vol. 19, pp. 1077–1081, IEEE, Albuquerque, NM, USA, 1980.
- [25] M. Ishitobi, “Properties of zeros of a discrete-time system with fractional order hold,” in *Proceedings of the IEEE Conference on Decision and Control*, vol. 4, pp. 4339–4344, IEEE, Kobe, Japan, 1996.
- [26] E. W. Bai and S. Dasgupta, “A minimal k-step delay controller for robust tracking of non-minimum phase systems,” in *Proceedings of the IEEE Conference on Decision and Control*, vol. 1, pp. 12–17, IEEE, Lake Buena Vista, FL, USA, 1994.
- [27] X. Wang and D. Chen, “Causal inversion of non-minimum phase systems,” in *Proceedings of the IEEE Conference on Decision and Control*, pp. 73–78, IEEE, Orlando, FL, USA, 2001.
- [28] J. Yao, X. Wang, S. Hu, and W. Fu, “Adaline neural network-based adaptive inverse control for an electro-hydraulic servo system,” *Journal of Vibration and Control*, vol. 17, no. 13, pp. 2007–2014, 2011.
- [29] M. Ahmed, N. Lachhab, and F. Svaricek, “Non-model based adaptive control of electro-hydraulic servo systems using pre-filter inversion,” in *Proceedings of the 9th International Multi-Conference on Systems, Signals and Devices (SSD)*, IEEE, Chemnitz, Germany, March 2012.
- [30] M. A. Shafiq, “Predicting the compressive strength of concrete using neural network and kernel ridge regression,” in *Proceedings of the Future Technologies Conference (FTC)*, pp. 821–826, IEEE, San Francisco, CA, USA, December 2016.
- [31] M. A. Shafiq, “Direct adaptive inverse control of nonlinear plants using neural networks,” in *Proceedings of the Future Technologies Conference (FTC)*, pp. 827–830, IEEE, San Francisco, CA, USA, December 2016.
- [32] M. Shafiq, M. A. Shafiq, and N. Ahmed, “Closed loop direct adaptive inverse control for linear plants,” *The Scientific World Journal*, vol. 2014, Article ID 658497, 2014.
- [33] D. T. M. Slock, “On the Convergence Behavior of the LMS and the Normalized LMS Algorithms,” *IEEE Transactions on Signal Processing*, vol. 41, no. 9, pp. 2811–2825, 1993.
- [34] N. J. Bershad, “Analysis of the normalized LMS algorithm with gaussian inputs,” *IEEE Transactions on Signal Processing*, vol. 34, no. 4, pp. 793–806, 1986.
- [35] O. Smith Julius W3K Publishing, 2nd edition, 2007.

Reproduced with permission of copyright owner. Further reproduction prohibited without permission.