



Online learning of stable robust adaptive controllers design based on data-dependent feedback linearization with application to rotary inverted pendulum

Mehmet Uğur Soydemir¹ · Savaş Şahin¹ · Aykut Kocaoğlu² · Parvin Bulucu³ · Cüneyt Güzeliş⁴

Received: 20 October 2022 / Accepted: 21 February 2024
© The Author(s) 2024

Abstract

This study introduces an online (supervised) learning method to design nonlinear auto-regressive moving average (NARMA) controllers for feedback-linearized nonlinear single-input single-output (SISO) systems. The algorithm ensures Schur stability of the overall closed-loop system and provides adaptiveness and robustness for the NARMA controllers. The first stage of the method derives, in a data-dependent way, a feedback-linearized model of the nonlinear plant by using its input and output sample pairs. The method's second stage, which constitutes the novel part of the presented study, builds up an online learning scheme for the linear auto-regressive moving average (ARMA) controller based on an already learned feedback-linearized model of the nonlinear plant. During online supervised learning, ARMA parameters of the feedback-linearized SISO plant model and the closed-loop ARMA model are computed by minimizing the plant identification and the closed-loop system tracking errors. Both errors are defined as $\ell_{1,\varepsilon}$, namely ε -insensitive loss functions that provide NARMA controller the robustness against noise and outliers. The proposed online learning control algorithm is applied to a rotary inverted pendulum model and to a real rotary inverted pendulum setup. The tracking performance of the developed controller is compared with those of the linear quadratic regulator and coupled sliding mode controller in terms of mean square error.

Keywords Stable robust adaptive control · Feedback linearization · Learning controller · Coupled sliding mode control · Rotary inverted pendulum

1 Introduction

Feedback linearization of nonlinear control systems is a widely known control method applied to continuous-time nonlinear plants that satisfy certain conditions such as

involutivity and being transformable into special canonical forms like Brunovsky or normal form [1–4]. The feedback linearization method is developed for discrete-time nonlinear systems too, both for affine [5] and non-affine systems [6]. By means of applying the artificial neural

✉ Savaş Şahin
savas.sahin@ikc.edu.tr; sahin.savas@yahoo.com

Mehmet Uğur Soydemir
soydemirmehmetugur@gmail.com

Aykut Kocaoğlu
aykut.kocaoğlu@deu.edu.tr

Parvin Bulucu
pervinhuseyinova@gmail.com

Cüneyt Güzeliş
cuneyt.guzelis@yasar.edu.tr

¹ Department of Electrical and Electronics Engineering, İzmir Katip Çelebi University, İzmir, Turkey

² Department of Electrical and Energy, İzmir Vocational School, Dokuz Eylül University, İzmir, Turkey

³ Department of Electrical and Electronics Engineering, Graduate School, Yaşar University, İzmir, Turkey

⁴ Department of Electrical and Electronics Engineering, Yaşar University, İzmir, Turkey

network (ANN) approach to the feedback linearization domain, ANN-based feedback linearization control methods have been proposed as powerful strategies for the control of the nonlinear systems in order to provide the ability of learning and generalization from the real data [6–14]. An approximate feedback linearization was developed by means of multilayer perceptron (MLP) based on a nonlinear state transformation providing local diffeomorphism conditions [8]. ANN-based control methods generally construct nonlinear auto-regressive moving-average (NARMA) plant models learned from the plant's input and output and design NARMA-based neuro controllers [6–10]. ANN-based feedback linearization methods usually implement two separate ANN blocks, each of which defines a nonlinear function to make the overall feedback system with the newly defined input as linear [6, 8, 9, 12]. The direct adaptive neural control scheme of [11] can also be considered as a kind of implementation of feedback linearization but with a radial basis function neural network for an affine nonlinear multi-input–multi-output system. The so-called Learning Feedback Linearization (LFL) method proposed in [12] employs an ANN-based NARMA model implementing input-to-state feedback linearization blocks and then uses the conventional PID controller for the overall feedback-linearized control system.

On the other hand, several adaptive control methods have been used in the literature [15–21] for controlling nonlinear systems. The adaptive control method in [15] uses ANN-based prediction of a reference trajectory and the model-free adaptive controller simultaneously via the feedback-linearized model of the nonlinear dynamics [15]. An inverse system-based adaptive nonlinear controller employing ANN is developed in [16]. A model reference adaptive control and its extension version are used together for guaranteeing adaptive control parameters converge [17]. The work in [18] introduces an adaptive ANN-based control method having adaptation laws obtained from input–output data for the control of inverted pendulum. An output feedback-based adaptive ANN controller scheme is developed in [19] for tracking control of switched non-strict-feedback nonlinear systems. The proposed adaptive control scheme is applied to the interconnected nonlinear system with the global stability condition [20]. A kind of inverse system approach is applied for an adaptive nonlinear controller design [22]. This study employs a Wiener-type nonlinear controller model and a Hammerstein-type nonlinear plant model such that the overall closed-loop system becomes a linear system to which the online learning ARMA controller design method of [23] is applied.

The study presented in this paper proposes a new LFL method that applies online learning NARMA controller

design methodology, originally developed in [23] for linear SISO plants, to the input–output feedback-linearized nonlinear affine SISO systems. The online learning NARMA controller design provides a Schur stable robust adaptive control scheme that is formulated in this work as a data-dependent robust solution to a system identification problem of a partially known system, i.e., the closed system. Herein, the known part is the data-dependent LFL-based identified plant model, whereas the unknown part is the controller to be adjusted online. The scheme firstly achieves the input–output feedback linearization of the nonlinear plant via LFL, and then, the plant model is identified as the combination of feedback linearization block and nonlinear plant. The overall linearized plant model parameters are computed by minimizing the plant identification error. The determination of the ARMA controller parameters is then performed by minimizing the tracking error, which is considered as the closed-loop system identification error, under Schur stability conditions. Both error terms are defined as regularized ε -insensitive loss functions that provide robustness property against to noise and disturbances, where the regularization is used for the penalization of generalization error [23–25].

The main contributions of the developed control scheme by which it differs from the related works existing in the literature are listed as follows.

- (i) The proposed control scheme relies on formulating both the plant identification and adaptive controller design as supervised learning problems. So, the proposed controller design gets benefit of the data-dependent online learning by which the closed-loop stability, adaptiveness, robustness, and generalization ability can be embedded in controller design process.
- (ii) As one of the most distinguished features of the proposed method, the controller design for tracking control purposes is described as an identification problem of a partially known system where the controller is the unknown part and the already identified plant model is the known part. From this point of view, the introduced supervised learning scheme for the controller design is applied in a direct way with no need to apply any preprocessing such as finding an inverse system for the plant. By means of this closed-loop system identification framework, the closed-loop system stability is ensured by imposing Schur conditions as linear inequality constraints in tracking error minimization. In the tracking error minimization procedure, which is implemented by pattern/batch mode minimization algorithms, the error is chosen as regularized ε -insensitive loss functions $\ell_{1,\varepsilon}(\cdot, \cdot)$.

Herein, being absolute for the error measures, i.e., ℓ_1 'ness, provides robustness against disturbances and other outliers. The ε -insensitiveness provides robustness against noise, and immunization for the model errors originated from the ANN-based LFL approximation to the nonlinear plant. The regularization term added to the loss functions makes the controller model possess higher generalization abilities. The regularize $\ell_{1,\varepsilon}(\cdot, \cdot)$ norm is used also in the plant identification phase as the identification error measure for building up a robust neural network model of the plant with high generalization ability.

- (iii) The proposed method provides the possibility of exploiting a linear ARMA controller for nonlinear plants by means of deriving a linear (augmented) plant model as a cascade of the LFL block and the nonlinear plant via input–output feedback linearization. The obtained linear closed-loop system model allows for exploitation of the online ARMA controller design method of [23] that is developed for linear plants. In the method of this paper, both the LFL block and linear ARMA controller are determined in a data-dependent way by using input–output data pairs measured from the nonlinear plant and the closed-loop system. It should be noted that the data-dependent LFL scheme of the proposed controller design method, which yields an ANN-based LFL approximation, does not require a known nonlinear plant model satisfying certain technical conditions such as involutivity to which the standard procedure of feedback linearization method valid for continuous-time systems [1, 2, 26] is applied. With the underlying learning abilities of ANN-based LFL approximation, the proposed data-dependent feedback linearization learns an approximate but acceptable linearized closed-loop system model. Herein, the approximation error is indeed defined in terms of the tracking and plant identification errors to be minimized. The learning feedback linearization scheme applied here is similar to the linearization method of [12]. Both are data-dependent feedback linearization. It means that the feedback-linearized (augmented) plant model is designed by learning in a supervised way. The main differences between them are the following. The method proposed in this paper is based on input–output linearization, while the study in [12] employs input-state linearization. On the other hand, the method in [12] employs a conventional PID whose parameters are learned by learning, so it does not have the adaptiveness, robustness,

generalization ability and the guarantee for the closed-loop stability described in (i)–(ii). In contrast, the method proposed in this paper has all these features that are important, especially for nonlinear control systems.

The most decisive aspect of this article's main contributions, which are described above, in terms of system identification and control applications of artificial neural networks is the definition of model identification and controller design problems as supervised learning that runs online. With the “learning of feedback linearization” proposed in this article, the study given in [23], which is based on linear (ARMA) system identification and online learning of ARMA controller parameters, has been extended to the method of supervised learning of linear ARMA-based closed-loop system identification. This is achieved by employing a learned feedback-linearized model of the nonlinear plant. Thus, linear ARMA-based system identification, which is a widely studied method not only in control area but also in time-series analysis, has been extended to the identification and control of nonlinear systems, where both system identification and controller design have been defined as supervised learning schemes that run online. In this sense, the paper has contributed to the use of artificial neural networks and machine learning in the field of system identification and control.

In short, the proposed controller design scheme is the extension of the method of online learning ARMA controllers developed in [23] for linear plants to nonlinear affine plant cases with exploiting the method of learning feedback linearization introduced in [12]. Preliminary results of this study are partially published in the conference paper [27]. The method can be called in two ways: stable robust adaptive ARMA controller for feedback-linearized nonlinear plants or stable robust adaptive NARMA controller for nonlinear plants. In the former case, LFL block is considered as the front side block of the linearized (augmented) plant model. In the latter, LFL block is considered as the backside block coming after the linear ARMA controller to constitute a NARMA controller. Since the LFL block is not a part of the plant such that it is just used to transform the plant to a linear one by a linearization feedback control, the LFL block is appropriate to be considered as a part of the controller so the developed controller in this paper is preferred to be named as the NARMA controller that consists of a linear ARMA block cascaded by the LFL block. In the application part of this study, to show its superiority, the developed LFL-based stable robust adaptive NARMA controller is compared to the linear quadratic regulator (LQR) of [28] and the coupled sliding mode controller (SMC) [29, 30] in terms of their tracking performances. As the nonlinear plant, the

ROtary inverted PENdulum (ROTPEN) is chosen. The experiments are conducted in two different modes: ROTPEN model is implemented in the software-in-the-loop mode first, that is, the computer simulation. Then, the developed method is applied for controlling a real ROTPEN plant in prototyping real-time simulation mode [31].

This paper is organized as follows. In Sect. 2, the developed LFL-based stable robust adaptive control scheme is explained. In Sect. 3, the performances of the proposed adaptive controller and the other controllers considered are compared to each other for the ROTPEN simulation model and a real ROTPEN plant. The conclusion and the possible future directions are provided in Sect. 4.

2 Proposed LFL-based stable robust adaptive control scheme

The proposed stable robust adaptive controller scheme, which is shown in Fig. 1, is implemented in three stages as follows. (1) A feedback-linearized plant model is constructed by an LFL block implemented with ANN that is cascaded by the nonlinear plant. The cascade combination of LFL block and the plant is identified as a linear ARMA plant model by minimizing the augmented plant identification error in (9). (2) The overall linear closed-loop system model that provides Schur stability is identified as an ARMA system model by minimizing the closed-loop system identification error in (12). (3) The linear ARMA controller parameters are calculated from the ARMA parameters of the closed-loop system found in 2) by using the augmented plant’s ARMA model parameters found in 1). These three stages might be followed from Fig. 1 in identifying the operation in the 1st stage that yields $\{a_n\}_{n=1}^N$ and $\{b_n\}_{n=1}^M$ as the plant ARMA model parameters, the operation in the 2nd stage that yields $\{\alpha_n\}_{n=1}^{\hat{N}}$ and $\{\beta_n\}_{n=1}^{\hat{M}}$ as the closed-loop system ARMA model

parameters, and the operation in the 3rd stage that yields $\{f_m\}_{m=1}^P$, $\{c_m\}_{m=1}^R$ and $\{d_m\}_{m=1}^Q$ as the controller ARMA model parameters.

2.1 LFL for the nonlinear plant

This subsection presents the rationale behind the augmented plant model, which is the cascade of the LFL nonlinear block and the SISO plant together with a state-feedback architecture, by a linear ARMA model. For this purpose, let us give a brief explanation of the input–output feedback linearization method for time-invariant discrete-time Single-Input Single-Output (SISO) affine plants [5]. Assume that the plant is defined in the following input–output form.

$$y^{k+1} = f_n(y^k, \dots, y^{k-n+1}; u^{k-d}, \dots, u^{k-d-m+1}) + g_n(y^k, \dots, y^{k-n+1}; u^{k-d}, \dots, u^{k-d-m+1})u^{k-d+1} \tag{1}$$

Herein $f_n(\bullet) : \mathbf{R}^n \times \mathbf{R}^m \rightarrow \mathbf{R}$ and $g_n(\bullet) : \mathbf{R}^n \times \mathbf{R}^m \rightarrow \mathbf{R}$ are infinitely many differentiable functions such that $g_n(\bullet)$ has a multiplicative inverse. $u^k \in \mathbf{R}$ is the control input, and $y^k \in \mathbf{R}$ is the output of the SISO system. $m \leq n$ and d stand for the relative degree of the system.

Defining a set of state variables as $x_i^k = y^{k-i+1}$ with $i \in \{1, 2, \dots, n\}$ and $x_{n+i}^k = y^{k-m-d+i}$ with $i \in \{1, 2, \dots, m+d-1\}$, the following state form is obtained for the system defined in (1).

$$\begin{aligned} x_i^{k+1} &= x_{i+1}^k \text{ with } i \in \{1, 2, \dots, n\} \\ x_n^{k+1} &= f_n(x^k) + g_n(x^k)x_{n+m}^k \\ x_{n+i}^{k+1} &= x_{n+i+1}^k \text{ with } i \in \{1, 2, \dots, m+d-2\} \\ x_{n+m+d-1}^{k+1} &= u^k \\ y^k &= x_n^k \end{aligned} \tag{2}$$

For a system with relative degree one, i.e., $d = 1$, we get $x_n^{k+1} = f_n(x^k) + g_n(x^k)u^k$.

So, the linearizing control input is obtained as

$$u^k = \frac{-f_n(x^k) + v^k}{g_n(x^k)} \tag{4}$$

where v^k represents the reference, namely the desired output at time $k + 1$.

For a system with the relative degree $d > 1$, the linearizing control can be derived [5] as follows.

$$u^k = \frac{-f_n(\mathbf{T}^{-1}[\mathbf{z}^k]) + v^k}{g_n(\mathbf{T}^{-1}[\mathbf{z}^k])} \tag{5}$$

where the invertible transformation $\mathbf{z}^k = \mathbf{T}[x^k]$ is defined below.

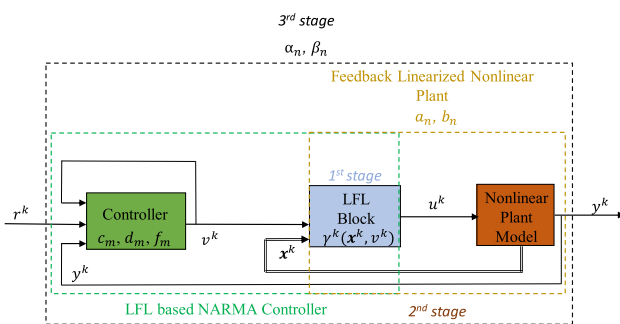


Fig. 1 The proposed LFL-based adaptive control scheme (Color figure online)

$$z^k = \begin{bmatrix} z_{11}^k \cdots z_{1n}^k z_{1,n+1}^k \cdots z_{1,n+d-1}^k z_{21}^k \cdots z_{2m}^k \\ \vdots \\ x_1^k \cdots x_n^k x_{n+1}^k \cdots x_n^k x_{n+1}^k \cdots x_{n+m}^k \end{bmatrix}^T \tag{6}$$

So, for an arbitrary relative degree d , the linearizing feedback control is defined in terms of \mathbf{x} and v as:

$$u = \gamma(\mathbf{x}, v) \tag{7}$$

In this paper, the LFL block that is inspired by the well-known input–output linearization methodology described above. Since the proposed method in the paper learns LFL block based on input–output data, an LFL block that provides an exact linearization from the newly defined control input v to the system output y by means of a feedback control $u = \gamma(\mathbf{x}, v)$ may not exist. In this sense, the LFL block depicted in Fig. 1 does just define an approximate linearization rather than an exact one. The LFL block used for an approximate linearization is considered in this paper as having the following (discrete time) linear ARMA representation.

$$y^k = \sum_{n=1}^N a_n y^{k-n} + \sum_{n=0}^M b_n v^{k-n} \tag{8}$$

The LFL block of this paper is implemented by a specific algebraic ANN, which has the universal approximation property, namely the feedforward MLP with a single hidden layer shown in Fig. 2. ANN is trained with the training sample set $\left\{ \left[(\mathbf{x}^s)^T v^s \right]^T, u^s \right\}_{s=0}^S$. Herein, \mathbf{x}^s whose entries are defined in (2) in terms of the plant output is the measured output at different time instants k . The LFL block is identified by training the MLP to learn the above given input–output samples in an offline manner with a batch mode (Levenberg–Marquardt) backpropagation algorithm.

Now, as expressed in SubSect. 2.2, the identification of the augmented plant with an already learned LFL block can be made based on the ARMA model of (8).

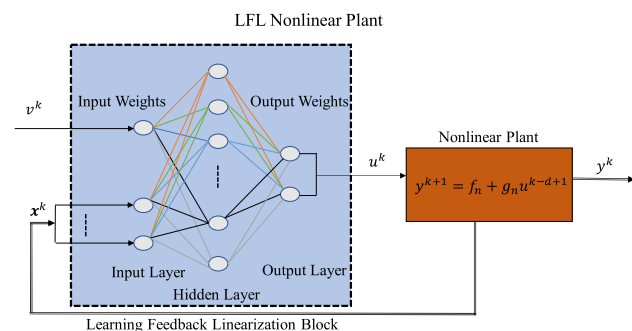


Fig. 2 The LFL is designed for the nonlinear plant during learning phase

2.2 ARMA-based identification of augmented plant

The ARMA model identification of the LFL-based augmented plant is depicted in Fig. 3. The augmented (feedback-linearized) plant identification is formulated as the minimization of the identification error in (9) that is defined by ε -insensitive loss function $\ell_{1,\varepsilon}(\cdot, \cdot)$ given in (10). In contrast to the offline nature of the identification of LFL block, the augmented plant is identified in an online (mini-batch) mode. In this mode, the ARMA parameters of the augmented plant are updated in time in accordance with the minimization of the regularized identification error in (9) for each time interval of $[k, k - K + 1]$ where K is the sliding window length.

$$\frac{1}{K} \sum_{s=0}^{K-1} \ell_{1,\varepsilon} \left(y_a^{k-s}, \sum_{n=1}^N a_n y^{k-s-n} + \sum_{n=0}^M b_n v^{k-s-n} \right) + \lambda \left\| \begin{bmatrix} a \\ b \end{bmatrix} \right\|_2^2 \tag{9}$$

Herein, ε -insensitive loss function $\ell_{1,\varepsilon}(\cdot, \cdot)$ is a measurement of the distance between the $(k - s)^{\text{th}}$ actual output sample y_a^s of the plant and the $(k - s)^{\text{th}}$ output sample of the linear ARMA model $y^{k-s} = \sum_{n=1}^N a_n y^{k-s-n} + \sum_{n=0}^M b_n v^{k-s-n}$ of the augmented plant model where N and M standing for the degrees of the ARMA model. As given in [32], the ε -insensitive (absolute value-based) loss function and the regularizing term are defined below.

$$\ell_{1,\varepsilon}(y_a^s, y^s) = \begin{cases} |y_a^s - y^s| & \text{if } |y_a^s - y^s| \geq \varepsilon \\ 0 & \text{if } |y_a^s - y^s| < \varepsilon \end{cases} \tag{10}$$

$$\lambda \left\| \begin{bmatrix} a \\ b \end{bmatrix} \right\|_2^2 = \lambda \left(\sum_{n=1}^N a_n^2 + \sum_{n=0}^M b_n^2 \right) \tag{11}$$

In the above given identification error, ε -insensitivity is used for providing robustness against noise,

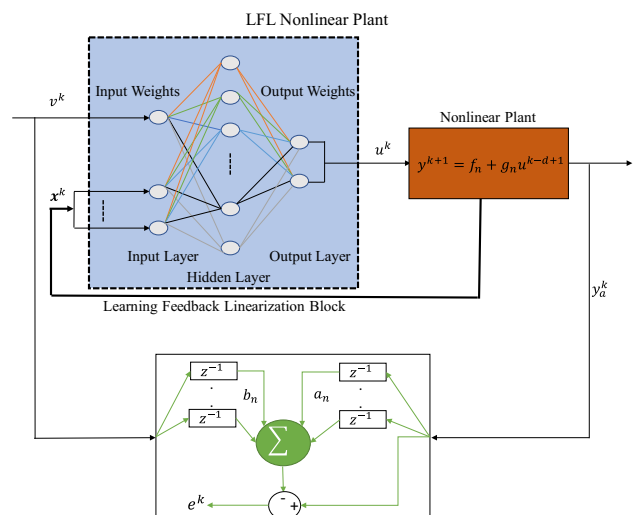


Fig. 3 ARMA plant identification of the augmented plant

disturbances, and small modeling/approximation errors Δy 's to the ARMA model. Herein, the modeling/approximation error Δy is considered as being additive to the output y , which is inaccuracy in the assumption of that the augmented plant model is exactly feedback linearizable by means of the LFL block realized by MLP. The regularization term in (11) is used to provide a smooth augmented plant model by avoiding over-fitting, and so increasing the generalization ability to respond well to the data unseen before [23].

2.3 Learning ARMA controller as a closed-loop system identification

The study presented in this paper considers the system identification problem of the closed-loop system as a controller design which is the only unknown part of the system (Fig. 1). Once the ARMA parameters $\{\{a_n\}_{n=1}^N, \{b_n\}_{n=1}^M\}$ of the augmented (linear) plant model are identified, the linear closed-loop system's ARMA parameters $\{\{\alpha_n\}_{n=1}^{\hat{N}}, \{\beta_n\}_{n=1}^{\hat{M}}\}$ are learned in an online manner, so are the ARMA controller parameters $\{\{f_m\}_{m=1}^P, \{c_m\}_{m=1}^R, \{d_m\}_{m=1}^Q\}$. The closed-loop system identification error to be minimized by the online (mini-batch mode) learning algorithm is indeed the regularized tracking error given with ε -insensitive loss function $\ell_{1,\varepsilon}(\cdot, \cdot)$ in (12) that is defined for the time interval of $[k, k - L + 1]$ where L is sliding window length.

$$\frac{1}{L} \sum_{s=0}^{L-1} \ell_{1,\varepsilon} \left(y_d^{k-s}, \sum_{n=1}^{\hat{N}} \alpha_n y^{k-s-n} + \sum_{n=0}^{\hat{M}} \beta_n r^{k-s-n} \right) + \lambda \left\| \begin{matrix} \alpha \\ \beta \end{matrix} \right\|_2^2 \quad (12)$$

Herein, ε -insensitive loss function $\ell_{1,\varepsilon}(\cdot, \cdot)$ is a measure of the distance between the $(k - s)^{\text{th}}$ desired output sample, namely the reference, y_d^s of the plant and the $(k - s)^{\text{th}}$ actual output sample $y^{k-s} = \sum_{n=0}^{\hat{N}} \alpha_n y^{k-s-n} + \sum_{n=0}^{\hat{M}} \beta_n r^{k-s-n}$ of the closed-loop system. \hat{N} and \hat{M} stand for the degrees of the ARMA model. $\lambda \left\| \begin{matrix} \alpha \\ \beta \end{matrix} \right\|_2^2$ is the regularization term. The tracking error in (12) is minimized under the Schur stability constraints $\alpha_0 > \dots > \alpha_{2N-1} > \alpha_{2N} > 0$ to ensure the stability of the closed-loop system [23, 33].

As done in [9], the determination of the ARMA controller parameters $\{\{f_m\}_{m=1}^P, \{c_m\}_{m=1}^R, \{d_m\}_{m=1}^Q\}$ from already learned ARMA parameters $\{\{\alpha_n\}_{n=1}^{\hat{N}}, \{\beta_n\}_{n=1}^{\hat{M}}\}$ of the closed-loop system and the ARMA parameters

$\{\{a_n\}_{n=1}^N, \{b_n\}_{n=1}^M\}$ of the augmented (linear) plant model is performed by means of the following straightforward algebraic derivations.

By choosing $a_0 = -1$ without losing the generality, (8) can be written in the implicit form of (13).

$$\sum_{n=0}^N a_n y_a^{k-n} + \sum_{n=0}^M b_n v^{k-n} = 0 \quad (13)$$

Assume that the controller (colored as the green block in Fig. 1) admits the following ARMA representation.

$$v^k = \sum_{m=1}^P f_m v^{k-m} + \sum_{m=0}^R c_m r^{k-m} + \sum_{m=0}^Q d_m y^{k-m} \quad (14)$$

Herein, v^k is the output signal of the controller, r^k is the reference signal that is aimed to be tracked by the plant, and c_m, d_m, f_m denote the controller parameters. With the choice of $f_0 = -1$, then (14) is written in the implicit form of (15).

$$\sum_{m=0}^P f_m v^{k-m} + \sum_{m=0}^R c_m r^{k-m} + \sum_{m=0}^Q d_m y^{k-m} = 0 \quad (15)$$

(16) is obtained by taking the weighted sum of (13) with weights f_m ,

$$\sum_{n=0}^P f_m \sum_{n=0}^N a_n y_a^{k-m-n} + \sum_{n=0}^P f_m \sum_{n=0}^M b_n v^{k-m-n} = 0 \quad (16)$$

Substitution of the expression for v^k in (14) into (16), (17) is obtained.

$$0 = \sum_{n=0}^P f_m \sum_{n=0}^N a_n y_a^{k-m-n} + \sum_{n=0}^M b_n \sum_{n=0}^M b_n \left[\sum_{m=0}^Q d_m y^{k-m-n} - \sum_{m=0}^R c_m r^{k-m-n} \right] \quad (17)$$

(17) can be simplified by introducing new parameters α_n, β_n as below.

$$y^k = \sum_{n=0}^{\hat{N}} \alpha_n y^{k-n} + \sum_{n=0}^{\hat{M}} \beta_n r^{k-n} \quad (18)$$

Herein $\hat{N} =: \max \{P + N, M + Q\}$, and $\hat{M} =: M + R$ might be chosen for different values of P, N, R, M and Q . Without loss of generality, one can assume that $P = N = R = M = Q$ and $\hat{N} = \hat{M}$. Then, the controller parameters $\{\{f_m\}_{m=1}^P, \{c_m\}_{m=1}^R, \{d_m\}_{m=1}^Q\}$ can be calculated from the below given Diophantine.

$$\begin{aligned}
 a_o &= 1 + a_0f_0 - b_0d_0 \\
 a_i &= \sum_{j=0}^i a_jf_{i-j} - \sum_{j=0}^i b_jd_{i-j} \text{ for } i \in \{1, 2, \dots, N\} \\
 &\text{for } i \in \{N + 1, N + 2, \dots, 2N\} \\
 \beta_i &= - \sum_{j=0}^i b_jc_{i-j} \text{ for } i \in \{0, 1, 2, \dots, N\} \\
 \beta_i &= - \sum_{j=i-N}^N b_jc_{i-j} \text{ for } i \in \{N + 1, N + 2, \dots, 2N\}
 \end{aligned}
 \tag{19}$$

3 Simulations and experimental results

The proposed LFL-based stable robust adaptive controller is tested on the ROTPEN model and on a real ROTPEN setup. The performances of LQR, SMC, and the developed controller are compared as the balance controllers in terms of the settling time, overshoot percentage, and mean square error (MSE) of tracking errors. Each balance controller algorithm is used with an energy-based swing-up algorithm. Controller signals in all experiments are saturated $\pm 12 V$. The effect of ϵ -insensitive loss functions of the proposed controller is investigated for the tracking error minimization under with and without measurement noise and disturbance for ROTPEN.

3.1 Experiment 1: ROTPEN model

The ROTPEN system is one of the most popular plants used in the field of the nonlinear control. The ROTPEN system model has one input and two outputs. Therefore, the ROTPEN might be represented as a single-input multiple-output (SIMO) system. Hence, the control signal is

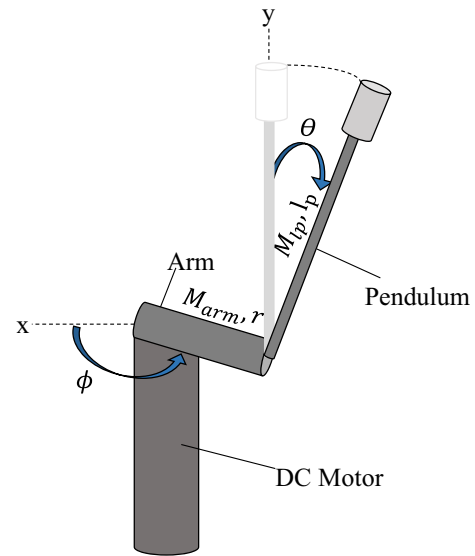


Fig. 4 ROTPEN 3-D simplified diagram

composed of summing of two controllers used for θ and ϕ borrowed from [34]. Either the pendulum angle or the angular position of the arm might be chosen as the output to be controlled. Some parameters of the dynamical system model are shown in a 3D diagram of ROTPEN in Fig. 4. The total force might be written in terms of Newton’s second law for ROTPEN system and dynamical model is given in (20) where θ and ϕ stands for the pendulum angle and the rotating arm angle, respectively, $\tau_{\text{output}} = K_t(V_m - K_m\dot{\phi})/R_m$ is used for torque control equation of the dc motor [35]. The ROTPEN system parameters are borrowed from [34] (See Table 1).

$$\begin{aligned}
 \ddot{\phi}(t) &= \frac{rl_pM_p \cos(\theta(t)) * \left[J_p \cos(\theta(t)) \sin(\theta(t)) \left(\dot{\phi}(t) \right)^2 + gl_pM_p \sin(\theta(t)) \right]}{J_p^2 \sin^2(\theta(t)) + J_pJ_{\text{eq}} - r^2l_p^2M_p^2 \cos^2(\theta(t))} \\
 &\quad - \frac{J_p \left[rl_pM_p \sin(\theta(t)) \left(\dot{\theta}(t) \right)^2 + 2J_p \left(\dot{\phi}(t) \right) \cos(\theta(t)) \sin(\theta(t)) \left(\dot{\theta}(t) \right) - \tau_{\text{output}} \right]}{J_p^2 \sin^2(\theta(t)) + J_pJ_{\text{eq}} - r^2l_p^2M_p^2 \cos^2(\theta(t))} \\
 \ddot{\theta}(t) &= \frac{\left[J_p \cos(\theta(t)) \sin(\theta(t)) \left(\dot{\phi}(t) \right)^2 + gl_pM_p \sin(\theta(t)) \right] * (J_p \sin^2(\theta(t)) + J_{\text{eq}})}{J_p^2 \sin^2(\theta(t)) + J_pJ_{\text{eq}} - r^2l_p^2M_p^2 \cos^2(\theta(t))} \\
 &\quad - \frac{J_p \left[rl_pM_p \sin(\theta(t)) \left(\dot{\theta}(t) \right)^2 + 2J_p \left(\dot{\phi}(t) \right) \cos(\theta(t)) \sin(\theta(t)) \left(\dot{\theta}(t) \right) - \tau_{\text{output}} \right]}{J_p^2 \sin^2(\theta(t)) + J_pJ_{\text{eq}} - r^2l_p^2M_p^2 \cos^2(\theta(t))}
 \end{aligned}
 \tag{20}$$

Table 1 Definitions of ROTPEN plant model parameters and their values

Symbol	Definitions (unit)	Value
K_m	Electromotive torque constant of the motor (V/(rad/s))	0.0333
g	Gravity acceleration (kg m ²)	9.81
B_{eq}	Arm viscous damping (Nms/rad)	0
B_p	Pendulum viscous damping (Nms/rad)	0
K_t	Motor torque constant (Nm)	0.0333
R_m	Armature resistance of the motor (Ω)	8.7
V_m	Motor input voltage (Volt)	0–24
M_{arm}	Mass of rotary arm (kg)	0.08
M_p	Mass of the whole pendulum “link and weight included” (kg)	0.027
r	Length of rotary arm (m)	0.0826
l_p	Length of inverted pendulum (m)	0.153
J_{eq}	Inertia rotary of rotary arm (kg m ²)	0.000368
J_p	Inertia rotary of inverted pendulum (kg m ²)	0.000698

By defining

$$x_1(t) = \phi(t), x_2(t) = \dot{\phi}(t), x_3(t) = \theta(t), x_4(t) = \dot{\theta}(t)$$

and then employing forward Euler approach, a discrete time approximation of a state-space form of the differential equation system in (20) can be obtained with $\Delta T = t_{k+1} - t_k$ in the following form

$$\begin{aligned} x_1^{k+1} &= x_1^k + \Delta T * x_2^k \\ x_2^{k+1} &= x_2^k + \Delta T * (\alpha - \beta) \\ x_3^{k+1} &= x_3^k + \Delta T * x_4^k \\ x_4^{k+1} &= x_4^k + \Delta T * (\rho - \sigma) \end{aligned} \tag{21}$$

Herein $\alpha, \beta, \rho,$ and σ are defined as:

$$\begin{aligned} \alpha &= \frac{r l_p M_p \cos(x_3^k) * [J_p \cos(x_3^k) \sin(x_3^k) (x_2^k)^2 + g l_p M_p \sin(x_3^k)]}{J_p^2 \sin^2(x_3^k) + J_p J_{eq} - r^2 l_p^2 M_p^2 \cos^2(x_3^k)} \\ \beta &= \frac{J_p [r l_p M_p \sin(x_3^k) (x_4^k)^2 + 2 J_p (x_2^k) \cos(x_3^k) \sin(x_3^k) (x_4^k) - \tau_{output}]}{J_p^2 \sin^2(x_3^k) + J_p J_{eq} - r^2 l_p^2 M_p^2 \cos^2(x_3^k)} \\ \rho &= \frac{[J_p \cos(x_3^k) \sin(x_3^k) (x_2^k)^2 + g l_p M_p \sin(x_3^k)] * ((J_p \sin^2(x_3^k)) + J_{eq})}{J_p^2 \sin^2(x_3^k) + J_p J_{eq} - r^2 l_p^2 M_p^2 \cos^2(x_3^k)} \\ \sigma &= \frac{r l_p M_p \cos(x_3^k) [r l_p M_p \sin(x_3^k) (x_4^k)^2 + 2 J_p (x_2^k) \cos(x_3^k) \sin(x_3^k) (x_4^k) - \tau_{output}]}{J_p^2 \sin^2(x_3^k) + J_p J_{eq} - r^2 l_p^2 M_p^2 \cos^2(x_3^k)} \end{aligned}$$

It should be noted that the obtained discrete-time plant model in (21) is linear in the input variable τ_{output} , so it constitutes an affine model with relative degree 1 as a special case of the input–output model in (1) that means

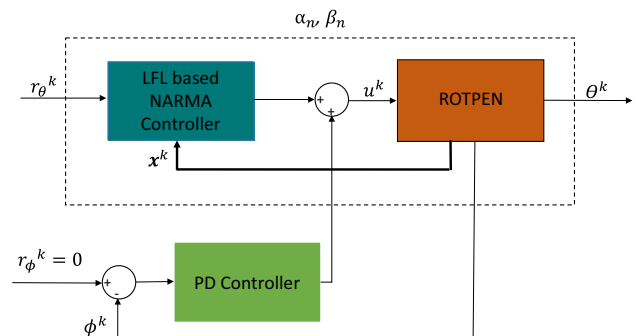


Fig. 5 LFL-based NARMA controller and ROTPEN block diagram

that the feedback linearization method presented in (2–8) for discrete-time nonlinear affine systems is applicable to control one of the outputs of the ROTPEN. In the application presented in the paper, the pendulum angle ($x_3 = \theta$) is chosen as the output to be controlled. For a realistic control, the rotating arm angle ($x_1 = \phi$), which is the second output of the ROTPEN, is to be controlled simultaneously. As for the balance controller part in Fig. 5, the

developed LFL-based stable robust adaptive controller is exerted to control the pendulum angle of the ROTPEN properly. Owing to the reference arm angle assumed as constant, e.g., $r_\phi^k = 0$, the SIMO system might be

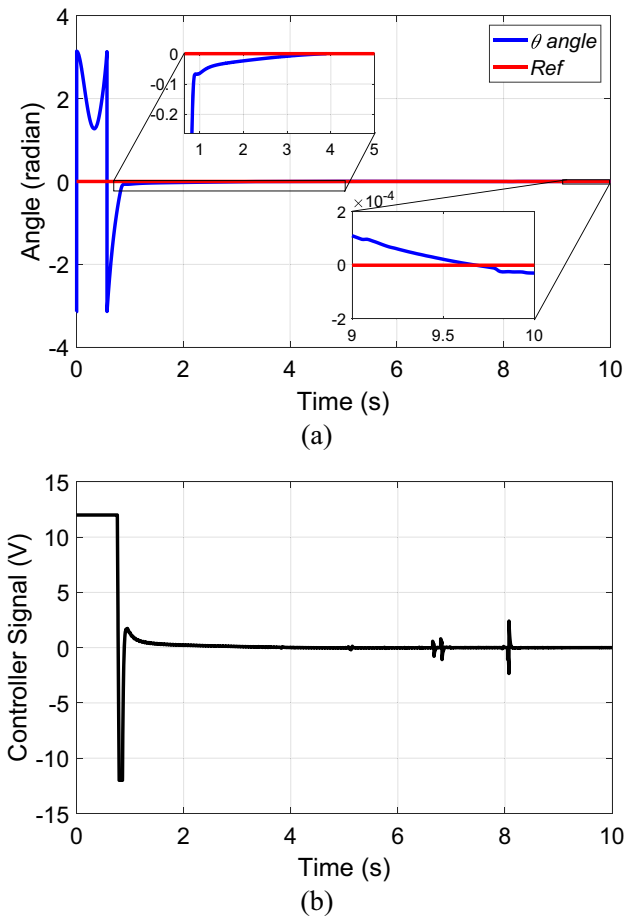


Fig. 6 **a** The tracking performance of the developed controller for pendulum angle θ and **b** developed controller's control signal

transformed into a SISO system depicted as a cascade control scheme borrowed from [36]. Herein, the rotating arm angle ϕ is controlled by a PD controller. Following the cascade control structure outlined in [36] and referring to Fig. 5, PD controller parameters were calculated using LQR technique [28] as $K_p = -1$, and $K_d = -1.2688$ for our case. The PD controller parameters are set to negative values. Thus, the balance control of the rotary inverted pendulum is achieved with the control signal generated by the developed LFL-based stable robust adaptive controller for θ^k in Fig. 5 [37]. In light of the cascaded control technique, it would be noted that the control of pendulum angle θ is not affected by the speed of the inner loop, because it responds quite faster than the outer loop [38].

The validity of the above approach of controlling a single output of a SIMO system by considering it as a SISO system for the considered output relies on the following fact. A SIMO system is reduced to a SISO system for the considered output if this output is associated with the slow dynamics of the system such that the fast dynamics are asymptotically stable so tending to the fixed reference

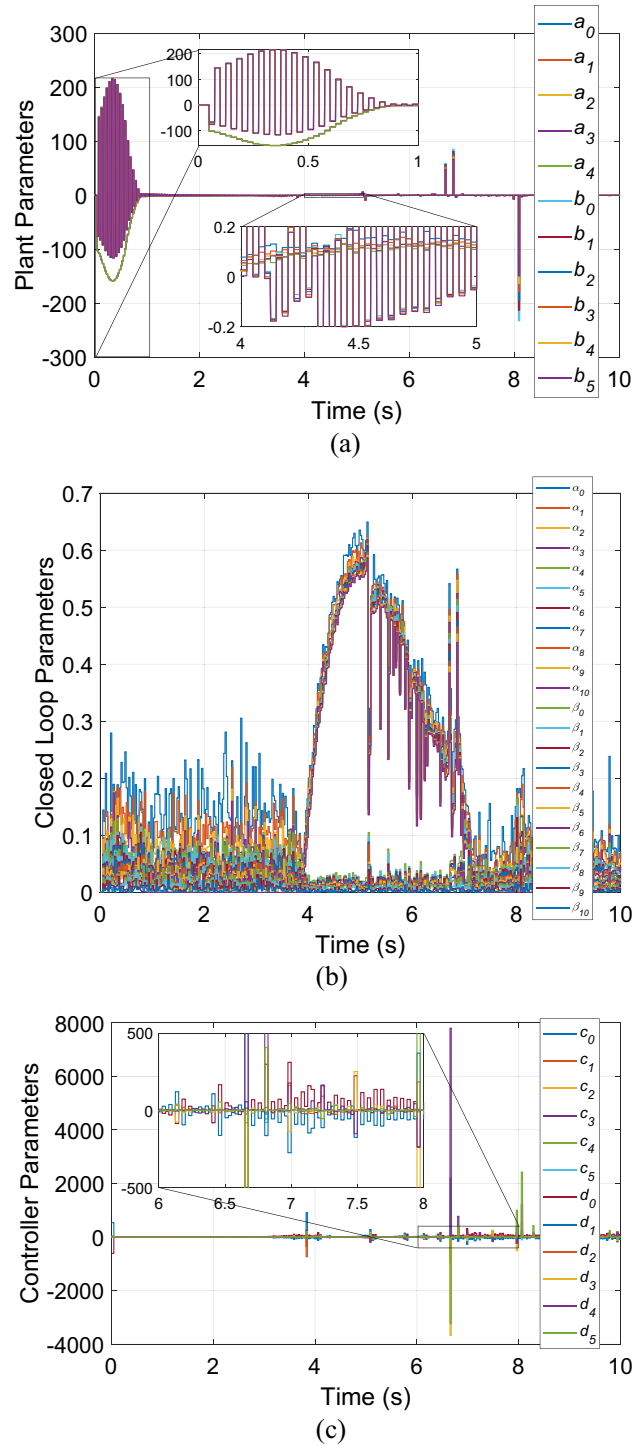


Fig. 7 **a** Time evolutions of the plant (a_n, b_n), **b** the closed-loop (α_n, β_n), and **c** the controller parameters (c_n, d_n)

points after their relatively short transient periods. For the ROTPEN, this can be seen that when fast dynamics associated with the rotating arm angle $x_1 = \phi(t)$ is settled down, so is its change $x_2 = \dot{\phi}(t)$, the slow dynamics determined by the last two recursions of (21) in terms of

Table 2 MSE performance of plant model identification (22) and tracking error (23) with and without noise for different ε -insensitiveness

ε	MSE (with noise)	MSE (without noise)
0	6.5664×10^{-4}	3.8650×10^{-4}
0.0001	5.8711×10^{-4}	5.2671×10^{-4}
0.001	6.0399×10^{-4}	3.8702×10^{-4}
0.01	6.3071×10^{-4}	3.9192×10^{-4}
0.1	5.1436×10^{-4}	4.3383×10^{-4}

the pendulum angle $x_3 = \theta(t)$ and its change $x_4 = \dot{\theta}(t)$ becomes to constitute an affine nonlinear SISO system.

In controlling the pendulum angle by the developed data-dependent feedback linearization method, (9) and (12) might be rewritten as (22) and (23), respectively.

$$\frac{1}{K} \sum_{s=0}^{K-1} \ell_{1,\varepsilon} \left(\frac{\theta_a^{k-s}, \sum_{n=1}^N a_n \theta^{k-s-n}}{\sum_{n=0}^M b_n v^{k-s-n}} \right) + \lambda \left\| \begin{matrix} a \\ b \end{matrix} \right\|_2^2 \quad (22)$$

$$\frac{1}{L} \sum_{s=0}^{L-1} \ell_{1,\varepsilon} \left(\frac{r_\theta^{k-s}, \sum_{n=1}^{\hat{N}} \alpha_n \theta^{k-s-n}}{\sum_{n=0}^M \beta_n r_\theta^{k-s-n}} \right) + \lambda \left\| \begin{matrix} \alpha \\ \beta \end{matrix} \right\|_2^2 \quad (23)$$

An energy-based swing-up algorithm is initially used for the balance control of the pendulum angle (θ) at the upright position. So, the pendulum angle is kept in a nearly linear range of -0.2618 rad and 0.2618 rad. The swing-up algorithm computes total energy of the system to use it as a feedback variable, so the pendulum total energy is driven to the upright position for reaching the linear range. Energy-based swing-up controller algorithm is given as in (24) [35, 39].

$$u_{\text{swing}} = \text{sat}_{\vartheta g} \left[\mu(E - E_d) \text{sign}(\dot{\theta} \cos(\theta)) \right] \quad (24)$$

where u_{swing} is control signal, μ is tunable design parameter, E is total energy of the system, E_d is total energy of upright position, and a linear function $\text{sat}_{\vartheta g}$ saturates u_{swing} control signal at $_{-}^{+}\vartheta g$ which ϑ is a design parameter and g is the gravity acceleration for limiting maximum acceleration of pivot. If the initial condition of θ is chosen as -3.14 rad and the upright position is chosen as 0 rad for θ , the potential and kinetic energy of the algorithm might be defined as $E_p = M_p g l_p (1 + \cos(\theta))$ and $E_k = \frac{1}{2} J_p \dot{\theta}^2$, respectively, according to chosen positions, so total energy becomes $E = E_p + E_k = M_p g l_p (1 + \cos(\theta)) + \frac{1}{2} J_p \dot{\theta}^2$. The potential energy is 0 when pendulum is at -3.14 rad and $E_d = 2M_p g l_p = 0.081J$ when pendulum is at the upright position. The ϑ and μ are chosen as 1 and 2.5, respectively, for the ROTPEN model [28, 39]. The swing-up algorithm

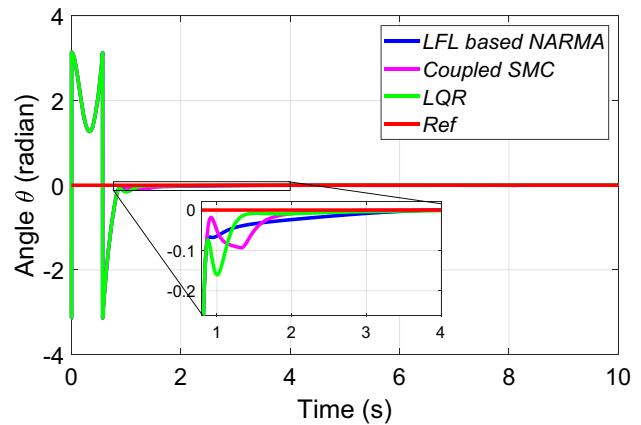


Fig. 8 ROTPEN plant model performance evaluation of controllers

Table 3 Comparison of the controller performances in terms of MSE

Controller	MSE
LQR	6.5817×10^{-4}
Coupled SMC	5.2586×10^{-4}
LFL-based NARMA	3.8650×10^{-4}

Table 4 Comparison of settling time and overshoot percentage performances of the controllers

Controller	Settling time (s)	Overshoot percentage (%)
LQR	1.296	- 16
Coupled SMC	1.71	- 9.3
LFL-based NARMA	2.1	0

is run when out of the linear range while a balance controller algorithm is run in the linear range via switching mechanism. During all experiments, the pendulum angle θ is plotted as $\text{mod}(\theta(t), 2\pi) - \pi$ [28].

After the pendulum has swung up, the proposed adaptive LFL-based NARMA controller takes its place and is applied to control the ROTPEN model as shown in Fig. 5. LFL-based NARMA controller, depicted in Fig. 5, consists

Table 5 Comparison of performances of the controllers in the presence of measurement noise and disturbance

Controller	MSE
LQR	7.8364×10^{-4}
Coupled SMC	6.4334×10^{-4}
LFL-based NARMA (epsilon = 0.1)	5.1436×10^{-4}

of an adaptive ARMA controller and a LFL block as presented in Fig. 1. Here, the LFL block is identified in an offline manner using $x(k)$ and v^k as input and u^k as output by a suitable MLP-ANN that possesses 2 hidden layers in the “feedforwardnet” structure and each of these layers has 10 neurons as described in Sect. 2.1. This MLP-ANN training has an epoch number of 1000 and 9 s of elapsed time. Besides, for the sliding window mode of the LFL-based NARMA controller, also called online (mini-batch) mode, the identification window of the plant model and the identification window of the control loop are taken as $K = 5, L = 30$, respectively, in (22) and (23). These loss functions are minimized so that $N = M = 5, \hat{N} = \hat{M} = 10$ by using the stochastic gradient descent algorithm borrowed from [40], and the sampling time is set to 0.001s during the simulation studies. Herein, the learning rate of stochastic gradient descent is 0.01 for minimization processes. As a result of minimization, feedback-linearized plant parameters a, b and closed-loop system parameters α, β are found, these parameters are evaluated by using (19) in an online manner along the sliding window to calculate the adaptive ARMA controller c_m, d_m . To test the tracking performance θ angle of the proposed controller, $r_\theta^k = 0rad$ is chosen for θ and $r_\phi^k = 0rad$. Initial value of θ is chosen as -3.14 rad and initial value of ϕ is chosen as 0 rad.

The results of tracking performance and time evolutions of the developed LFL-based adaptive controller with swing-up algorithm and developed LFL-based adaptive controller signal are depicted for the reference signals in Fig. 6a and b, respectively. As seen from Fig. 6a, the developed controller works well for the balance tracking performance. Time evolutions of the LFL-based linear plant model, the closed-loop, and the controller parameters are illustrated for the constant reference signal as 0 rad in Fig. 7. The value of regularization term as $\lambda = 0.05$ is found as the most appropriate value out of the set $\{0, 10^{-2}, 5 \times 10^{-2}, 10^{-1}\}$ for the online mode in order to limit the norm of both plant and controller parameters for better generalization.

To test the robustness performance of the proposed LFL-based adaptive controller, a white noise that has a 2 dB “SNR” is applied to the control signal v^k in (14) and a

40 dB “SNR” is applied to pendulum angle θ via SIMULINK with “awgn” function. The performances of the tracking error for balance control (23) are presented with different ε -insensitiveness values in terms of the MSE results with and without noise given in Table 2 where $MSE = \frac{1}{S} \sum_{i=1}^S e^2(i)$ defined in terms of S number of samples and $e(i)$ error signal. The eligible ε -insensitiveness of the tracking error might be used as 5.1436×10^{-4} and 3.8650×10^{-4} in terms of MSE with and without noise, respectively.

To compare the tracking performance of the proposed adaptive controller, LQR controller [28] and coupled SMC [29, 30] are applied to ROTPEN plant model. LQR design is based on the state feedback control signal $u = -K_k x$ where x is $N \times 1$ column vector and K_k is $1 \times N$ row vector. The goal is to find an optimum u value that minimizes a quadratic cost function as $J = \int_0^\infty (x^T Q x + u^T R u) dt$ where Q is chosen as

$$\begin{bmatrix} Q_{11} & 0 & 0 & 0 \\ 0 & Q_{22} & 0 & 0 \\ 0 & 0 & Q_{33} & 0 \\ 0 & 0 & 0 & Q_{44} \end{bmatrix}$$

that should be a positive semi definite matrix and R is a scalar for $x = [\phi; \theta; \dot{\phi}; \dot{\theta}]$ of the linearized model of ROTPEN from [35]. The simulations are implemented with the parameters given in Table 1 and initial conditions of $x = [0; -3.14; 0; 0]$. Q and R are chosen as

Table 6 Comparison of performances of the controllers in the presence of parameter uncertainties

Controller	MSE
LQR	6.6802×10^{-4}
Coupled SMC	5.5816×10^{-4}
LFL-based NARMA	5.4007×10^{-4}

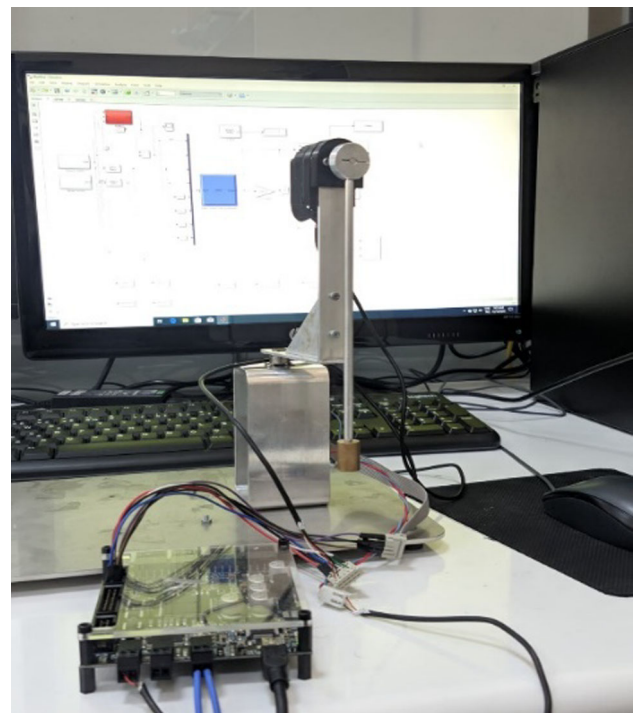


Fig. 9 ROTPEN real setup and V-DAQ data acquisition card

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1162 & 0 & 0 \\ 0 & 0 & 0.913 & 0 \\ 0 & 0 & 0 & 13 \end{bmatrix}$$

and 1, respectively, so that K_k is found as $[-1, 57.0528, -1.3636, 6.8465]$.

As for the coupled SMC [29, 30], (20) might be written as

$$\begin{aligned} \ddot{\phi}(t) &= f_{\phi}(\phi(t), \theta(t), \dot{\phi}(t), \dot{\theta}(t)) + g_{\phi}(\phi(t), \theta(t))u \\ \ddot{\theta}(t) &= f_{\theta}(\phi(t), \theta(t), \dot{\phi}(t), \dot{\theta}(t)) + g_{\theta}(\phi(t), \theta(t))u \end{aligned} \quad (25)$$

and sliding surface s is chosen as coupled sliding surface as follows

$$s = s_{\theta} + \bar{\lambda}_u s_{\phi} \quad (26)$$

where s_{θ} is sliding surface for pendulum angle, s_{ϕ} is sliding surface for rotary arm angle and state-dependent $\bar{\lambda}_u =$

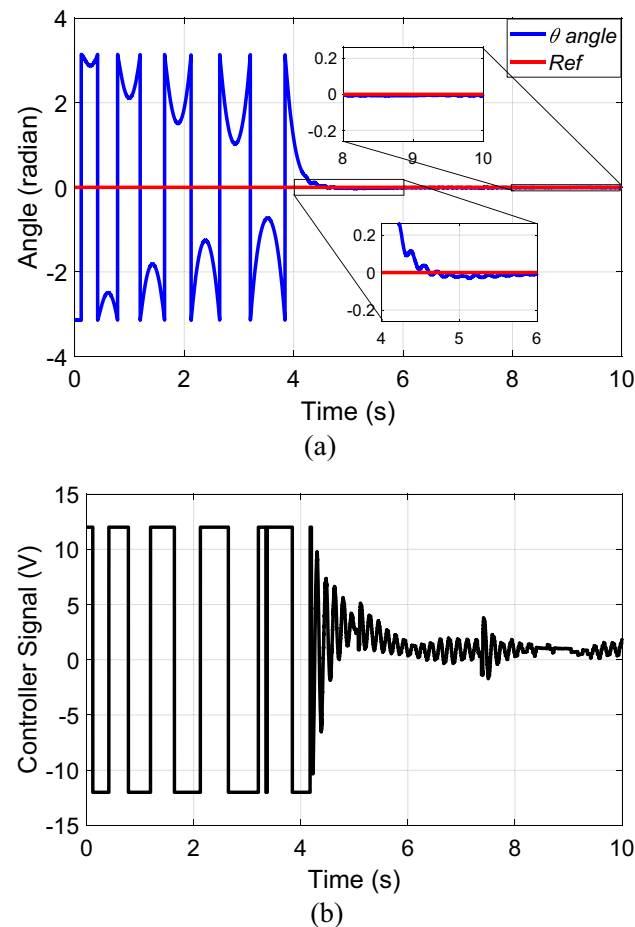


Fig. 10 **a** The tracking performance of the developed controller for pendulum angle θ and **b** developed controller's control signal

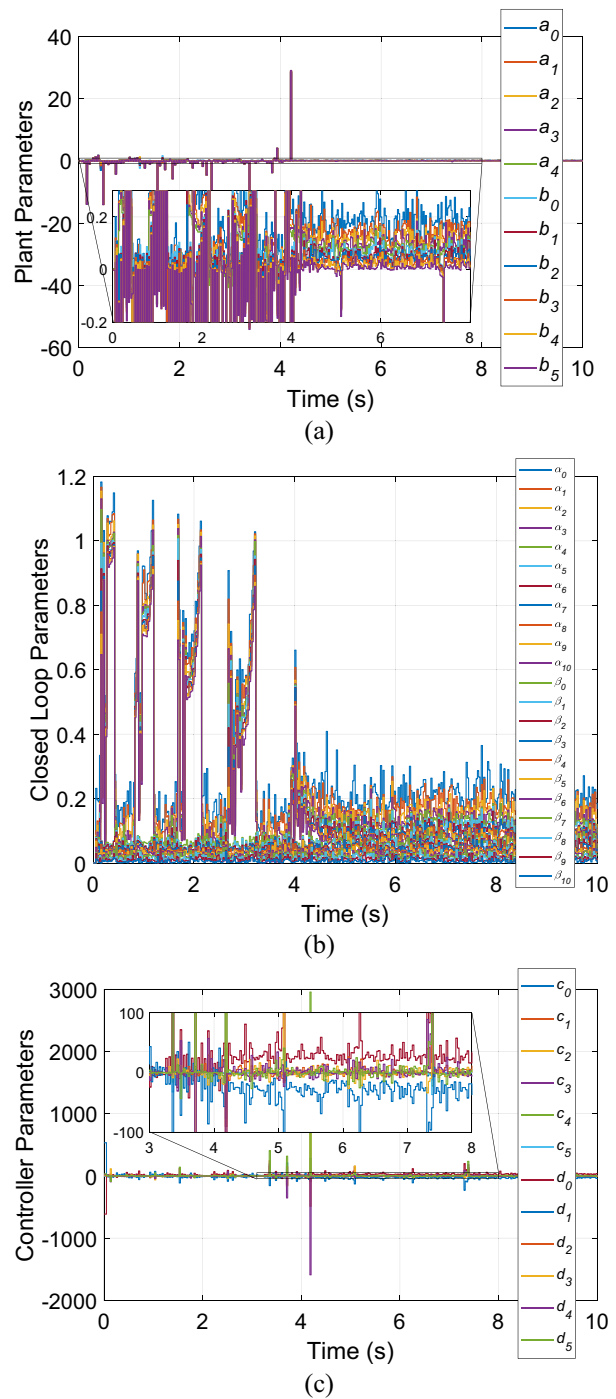


Fig. 11 **a** Time evolutions of the plant (a_n, b_n), **b** the closed-loop (α_n, β_n) and **c** the controller parameters (c_n, d_n)

$\lambda \cos(\theta(t))$ which has a λ_u constant coupling parameter in the following form

$$s_{\theta} = \dot{\theta}(t) + c_{\theta}\theta(t) \quad (27)$$

$$s_{\phi} = \dot{\phi}(t) + c_{\phi}\phi(t) \quad (28)$$

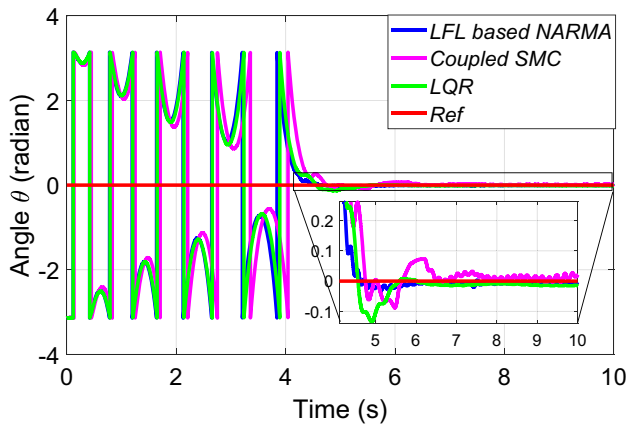


Fig. 12 ROTPEN real plant performance evaluation of the controllers

Table 7 Comparison of the controller performances in terms of MSE

Controller	MSE
LQR	32×10^{-4}
Coupled sliding mode	34×10^{-4}
LFL-based NARMA	9.0344×10^{-4}

Table 8 The evaluations of settling time and overshoot percentage in terms of the controller’s performance

Controller	Settling time (s)	Overshoot percentage (%)
LQR	5.412	– 13
Coupled sliding mode	6.43	– 8.7
LFL-based NARMA	5.174	– 2.6

where c_θ and c_ϕ are positive coefficients. The control law of the coupled SMC (u_{smc}) might be given as follows

$$u_{smc} = \frac{-\left(f_\theta + c_\theta \dot{\theta}(t) + \overline{\lambda}_u \left(f_\phi + c_\phi \dot{\phi}(t)\right)\right) - k \text{sign}(s) - \dot{\overline{\lambda}}_u s_\phi}{g_\theta + \overline{\lambda}_u g_\phi} \tag{29}$$

where k is a positive design parameter and $\dot{\overline{\lambda}}_u$ is time derivative of $\overline{\lambda}_u$. c_ϕ, c_θ, k , and $\overline{\lambda}_u$ were chosen as 7.5, 1.5, 3, and -0.35 , respectively.

Comparisons of the tracking performances of the developed controller and the other two controllers considered are given in Fig. 8 LFL-based NARMA controller provides good performance in terms of MSE as shown in Table 3. The overshoot percentage of the proposed controller is observed as 0 rad, and it is less than the other

controllers’ overshoot percentage values. The settling time of the proposed controller is computed as 2.1 s according to %2 criterion gives a good response as shown [41] in Table 4. The proposed controller performance might be an acceptable performance during the transient behavior because it avoids the swinging motion visible in the zoomed area in Fig. 8 observed in other controllers.

To compare the performance of the controllers in the presence of measurement noise and disturbance, white noise which has a 2 dB “SNR” is applied to the control signal of LQR, coupled SMC and the developed controllers and a 40 dB “SNR” is applied to pendulum angle θ of them via SIMULINK with “awgn” function. The obtained results are given in Table 5 where LFL-based controller provides the minimum MSE value. In addition to the noise and disturbance analysis, ten separate simulations are conducted to evaluate controller performance under parameter uncertainties. In each run, $\pm 5\%$ uncertainty is randomly added to both pendulum mass (M_p) and length (l_p) of the ROTPEN system, simulating deviations from the nominal model. Table 6 presents the average MSE values across these ten trials. The proposed LFL-based controller with its adaptive manner seems to perform slightly better than the coupled SMC and outperforms LQR since the proposed controller achieves the lowest overall MSE as shown in Table 6. Furthermore, the achievement of performance close to coupled SMC, which is known for its robustness to parameter uncertainties, underscores the significant strength of the proposed method.

3.2 Experiment: real ROTPEN Setup

Real ROTPEN setup is an example of a well-known under-actuated mechanical system [34, 42–45]. Incompletely driven mechanical systems are widely used in the field of robotics, and the main feature of these systems is that they have fewer actuators than degrees of freedom [46]. The inverted pendulum possesses unstable and nonlinear dynamical behaviors inherently. Another important feature that makes the rotary inverted pendulum more interesting is that it forms the basis of many new technologies such as seismometers, humanoid robots, unmanned air vehicles, and rockets. The real ROTPEN setup consists of mechanical design, data acquisition card, and software. The mechanical design of the pendulum is made by SolidWorks software. A direct current motor Maxon DC-max B773D74C44771 is used to rotate the ROTPEN arm horizontally. The pendulum is connected to the pendulum arm by the pivot. Thus, the pendulum will be able to oscillate easily. AVAGO HEDM-5505-j06 two-channel 1024 resolution encoder is located on the shaft. This encoder is used to measure the angle of the pendulum with the horizontal

plane and to implement the control system. The end of the L-shaped pendulum arm is mounted on the shaft of the dc motor. Due to the circular rotation of the motor shaft, the pendulum arm can be moved clockwise and counter-clockwise. The angle of the arm is calculated with the encoder mounted on the motor. A rotating arm in a horizontal axis and a rotating pendulum which is mounted on arm, in a vertical plane take part in the rotary inverted pendulum [42–44]. The final version of the successful ROTPEN setup is given in Fig. 9. For ROTPEN setup, the encoder reading for θ angle is set to be the same as the simulation, -3.14 rad initial condition as updown position and 0 rad as upright position. Theta encoder is set to 0 rad for initial condition. The proposed controller is tested on ROTPEN real plant via real-time desktop interface of SIMULINK environment [47].

To test the tracking performance of the proposed adaptive controller, it is used together with the swing-up algorithm described in the subSect. 3.1. The feedback-linearized plant model and the closed-loop system model have $N = M = 5$, and $\hat{N} = \hat{M} = 10$ degrees, respectively. For the online sliding window mode, the plant model identification window and the closed-loop identification window are taken as $K = 5$, $L = 30$ for (22) and (23), respectively. The loss functions are minimized with the stochastic gradient descent algorithm borrowed from [40], and the sampling time is chosen as 0.001 s. In order to test tracking performance θ angle of the proposed controller, $r_{\theta}^k = 0$ rad is chosen for θ angle (pendulum upright position). Herein, the parameters initial values are determined for the online mode after running a batch mode. $\lambda = 0.05$ is found the best as the regularization term out of the set $\{0; 10^{-2}; 5 \times 10^{-2}; 10^{-2}; 10^{-1}\}$. The results of tracking performance and time evolutions of the developed LFL-based adaptive controller and its signal are depicted for the reference signal in Fig. 10a and b, respectively. Time evolutions of the proposed LFL-based adaptive controller for the considered plant, closed-loop, and the developed controller parameters are depicted in Fig. 11a, b, and c, respectively.

In order to compare the performance of the LQR, coupled SMC, and the proposed controllers are applied to ROTPEN real plant. The comparison of the performances of the controllers is given in Fig. 12. LFL-based NARMA controller provides lower error than the other controllers according to the MSE error as shown in Table 7. According to Table 8, the overshoot percentage of the proposed controller is observed as -2.6% which is better than the other controllers. The 5.174 s settling time of the proposed controller that is found for the proposed controller is better than the other controllers.

4 Conclusions

This paper proposes a novel data-dependent NARMA controller for nonlinear affine systems. The controller parameters are adaptively learned using a sliding window approach. It builds on a previous work for linear systems [23] by extending it to nonlinear affine systems based on the data-dependent implementation of input–output feedback linearization. Firstly, the offline input–output feedback linearization is cascaded to the nonlinear plant to form a linear augmented plant. Subsequently, an online identification process takes place within the sliding window, modeling both the augmented plant, and the overall closed-loop system as linear ARMA models. The controller parameters are determined by considering Schur stability constraints during the closed-loop identification phase, guaranteeing the stability of the entire control system. Furthermore, the ε -insensitive $\ell_{1,\varepsilon}(\cdot, \cdot)$ loss function, employed for minimizing identification and tracking errors in the online learning algorithms, contributes to the NARMA controller's robustness against noise, disturbances, and modeling errors.

The rationale behind the usage of the proposed data-dependent control method based on the artificial neural networks-based approximate feedback linearization lies on the following facts. (1) Not all of the plants have well-defined realistic mathematical models that reflect the effects of all intrinsic nonlinearities, noise, disturbances, and parameter changes. So, there is a need to develop and implement data-dependent control methods that work with measurement data. (2) Usage of learning algorithms provides ways to determine the parameters of the identification model and also the controller model's parameters in an offline and also online manner. (3) Artificial neural networks with their universal approximation properties provide good candidates with the features such as being realistic, robust, and adaptive both for plant identification and controller models. (4) The performance of the data-dependent nonlinear controller design methods is generally lower than those of the linear ones. So, employing the feedback linearization first and then applying a data-dependent linear controller design method might be a viable approach that is also followed in the presented work.

The proposed NARMA controller is tested on a simulated ROTPEN model and on a real setup for angular pendulum position. Its performance is compared to those of LQR and coupled SMC. The proposed NARMA controller is observed to be superior in comparison with the other two controllers in terms of overshoot percentage and tracking performance in MSE.

The application of the developed NARMA controller is by no means restricted to ROTPEN and has the potential of

being used for controlling a large class of SISO nonlinear plants. As future works, the NARMA controller scheme developed for nonlinear SISO plants can be extended into multi-input-multi-output plants which might be non-affine. Developing methods for automatic determination of the sizes of sliding windows in online learning of the parameters of the augmented plant model and controller and investigation of the effects of the sliding window sizes in the tracking performance of the proposed NARMA controller are among possible future research subjects.

Author contributions All authors conceived and supervised the research. Mehmet Uğur Soydemir contributed to algorithm implementation, simulations, real system experiments, and paper writing. Savaş Şahin contributed to design and paper writing. Aykut Kocaoğlu contributed to design, simulations, and paper writing. Parvin Bulucu contributed to simulations, and Cüneyt Güzeliş contributed to design and paper writing. All authors read and approved the final manuscript.

Funding Open access funding provided by the Scientific and Technological Research Council of Türkiye (TÜBİTAK). This study was supported by the Scientific and Technological Research Council of Turkey (TÜBİTAK) under Grant 116E170.

Data and code availability The data and codes used for developing the algorithm are available upon direct request to the author, M. U. Soydemir.

Declarations

Conflict of interest There is no conflict of interest or competing interest.

Ethical approval Not applicable.

Consent to participate Not applicable.

Consent for publication Not applicable.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Slotine JJE, Li W (1991) Applied nonlinear control. Prentice-Hall
- Sastry SS, Isidori A (1989) Adaptive control of linearizable systems. *IEEE Trans Autom Control* 34(11):1123–1131. <https://doi.org/10.1109/9.40741>
- Zhihong M, Yu XH, Wu HR (1998) An RBF neural network-based adaptive control for SISO linearisable nonlinear systems. *Neural Comput Appl* 7:71–77. <https://doi.org/10.1007/BF01413711>
- Ammar A, Kheldoun A, Metidji B, Ameid T, Azzoug Y (2020) Feedback linearization based sensorless direct torque control using stator flux MRAS-sliding mode observer for induction motor drive. *ISA Trans* 98:382–392. <https://doi.org/10.1016/j.isatra.2019.08.061>
- Chen FC, Khalil HK (1995) Adaptive control of a class of nonlinear discrete-time systems using neural networks. *IEEE Trans Autom Control* 40(5):791–801. <https://doi.org/10.1109/9.384214>
- Deng H, Li HX, Wu YH (2008) Feedback-linearization-based neural adaptive control for unknown nonaffine nonlinear discrete-time systems. *IEEE Trans Neural Netw* 19(9):1615–1625. <https://doi.org/10.1109/TNN.2008.2000804>
- Lewis F, Jagannathan S, Yesildirek A (1999) Neural network control of robot manipulators and non-linear systems. Taylor & Francis, London
- He S, Unbehauen R (1998) Approximate feedback linearisation using multilayer neural networks. *Neural Process Lett* 2(8):131–144. <https://doi.org/10.1023/A:1009644612275>
- Pedro J, Dahunsi O (2011) Neural network based feedback linearization control of a servo-hydraulic vehicle suspension system. *Int J Appl Math Comput Sci* 21(1):137–147. <https://doi.org/10.2478/v10006-011-0010-5>
- Narendra KS, Mukhopadhyay S (1997) Adaptive control using neural networks and approximate models. *IEEE Trans Neural Netw* 8(3):475–485. <https://doi.org/10.1109/72.572089>
- Dheeraj K, Jacob J, Nandakumar M (2019) Direct adaptive neural control design for a class of nonlinear multi input multi output systems. *IEEE Access* 7:15424–15435. <https://doi.org/10.1109/ACCESS.2019.2892460>
- Sahin S (2016) Learning feedback linearization using artificial neural networks. *Neural Process Lett* 44(3):625–637. <https://doi.org/10.1007/s11063-015-9484-8>
- Xu K, Wang Z (2022) The design of a neural network-based adaptive control method for robotic arm trajectory tracking. *Neural Comput Appl*. <https://doi.org/10.1007/s00521-022-07646-y>
- Liu A, Zhao H, Song T et al (2021) Adaptive control of manipulator based on neural network. *Neural Comput Appl* 33:4077–4085. <https://doi.org/10.1007/s00521-020-05515-0>
- Cremer S, Das S, Wijayasinghe I, Popa D, Lewis F (2020) Model-free online neuroadaptive controller with intent estimation for physical human–robot interaction. *IEEE Trans Robot* 36(1):240–253. <https://doi.org/10.1109/TRO.2019.2946721>
- Cabrera JBD, Narendra KS (1999) Issues in the application of neural networks for tracking based on inverse control. *IEEE Trans Autom Control* 44(11):2007–2027. <https://doi.org/10.1109/9.802910>
- Pan Y, Zhang J, Yu H (2016) Model reference composite learning control without persistency of excitation. *IET Control Theory Appl* 10(16):1963–1971. <https://doi.org/10.1049/iet-cta.2016.0032>
- Moreno-Valenzuela J, Aguilar-Avelar C, Puga-Guzmán S, Santibáñez V (2016) Adaptive neural network control for the trajectory tracking of the Furuta pendulum. *IEEE Trans Cybern* 46(12):3439–3452. <https://doi.org/10.1109/tyb.2015.2509863>
- Dong Y, Yu Z, Yu Z, Li S, Li F (2017) Adaptive output feedback tracking control for switched nonstrict-feedback nonlinear systems with unknown control direction and asymmetric saturation

- actuators. *IET Control Theory Appl* 11(15):2539–2548. <https://doi.org/10.1049/iet-cta.2017.0124>
20. Huang J, Wang QG (2018) Decentralized adaptive control of interconnected nonlinear systems with unknown control directions. *ISA Trans* 74:60–66. <https://doi.org/10.1016/j.isatra.2018.01.008>
 21. Boroujeni M, Markadeh GR, Soltani J (2017) Torque ripple reduction of brushless DC motor based on adaptive input-output feedback linearization. *ISA Trans* 70:502–511. <https://doi.org/10.1016/j.isatra.2017.05.006>
 22. Bulucu P, Soydemir M, Sahin S, Kocaoğlu A, Güzeliş C (2020) Learning stable robust adaptive NARMA controller for UAV and its application to Twin Rotor MIMO systems. *Neural Process Lett* 52(1):353–383. <https://doi.org/10.1007/s11063-020-10265-0>
 23. Sahin S, Güzeliş C (2016) Online learning ARMA controllers with guaranteed closed-loop stability. *IEEE Trans Neural Netw Learn Syst* 27(11):2314–2326. <https://doi.org/10.1109/TNNLS.2015.2480764>
 24. Tibshirani R (1996) Regression shrinkage and selection via the lasso. *J R Stat Soc Ser B Methodol* 58(1):267–288. <https://doi.org/10.1111/j.2517-6161.1996.tb02080.x>
 25. Kukreja L, Löfberg S, Brenner JM (2006) A least absolute shrinkage and selection operator (lasso) for nonlinear system identification. *IFAC Proc* 39(1):814–819. <https://doi.org/10.3182/20060329-3-AU-2901.00128>
 26. de Jesús RJ (2018) Robust feedback linearization for nonlinear processes control. *ISA Trans* 74:155–164. <https://doi.org/10.1016/j.isatra.2018.01.017>
 27. Soydemir MU, Sahin S, Bulucu P, Kocaoğlu A, Güzeliş C (2019) Learning feedback linearization based stable robust adaptive NARMA controller design for rotary inverted pendulum. In: 2019 11th International Conference on Electrical and Electronics Engineering (ELECO), Bursa
 28. QUANSER, Rotary Pendulum (ROTPEN) inverted pendulum trainer (instructor manual)
 29. Rajan A, Kumar A, Kavitha C (2017) Robust control methods for swing-up and stabilization of a rotary inverted pendulum. In: 2016 International Conference on Emerging Technological Trends (ICETT), Kollam
 30. Park MS, Chwa D (2009) Swing-up and stabilization control of inverted-pendulum systems via coupled sliding-mode control method. *IEEE Trans Ind Electron* 56(9):3541–3555. <https://doi.org/10.1109/TIE.2009.2012452>
 31. Sahin S, İşler Y, Güzeliş C (2010) A microcontroller based test platform for controller design. In: 2010 IEEE International Symposium on Industrial Electronics, Bari-Italy
 32. Vapnik V (1998) *Statistical learning theory*, 1st edn. Wiley
 33. Gu K, Kharitonov V, Chen J (2003) *Stability of time-delay systems*. Birkhäuser, Boston
 34. QUANSER: QNET ROTPEN Workbook (Student). http://eelabs.faculty.unlv.edu/docs/labs/ee370L/ee370L_07_experiment_7.pdf. Accessed 16 June 2020
 35. Balula A (2016) *Nonlinear control of an inverted pendulum*. Master Thesis, Técnico Lisboa
 36. Oh SK, Jung SH, Pedrycz W (2009) Design of optimized fuzzy cascade controllers by means of hierarchical fair competition-based genetic algorithms. *Expert Syst Appl* 36:11641–11651. <https://doi.org/10.1016/j.eswa.2009.03.027>
 37. Levis M, Quanser. <https://www.quanser.com/blog/rotary-pendulum-control-challenge-with-qube-servo/> (2020). Accessed 12 June 2021
 38. Liptak B (2018) *Instrument engineers' handbook, vol 2, Process control and optimization*, CRC Press
 39. Mathew N, Rao K, Sivakumaran N (2013) Swing up and stabilization control of a rotary inverted pendulum. In: *IFAC Proceedings*. vol 46, no 32, pp 654–659. <https://doi.org/10.3182/20131218-3-IN-2045.00128>
 40. Ruder S (2016) An overview of gradient descent optimization algorithms. *ArXiv abs/1609.04747*
 41. Ogata K (2009) *Modern control engineering*. Pearson, London
 42. Pathak K, Franch J, Agrawal S (2005) Velocity and position control of a wheeled inverted pendulum by partial feedback linearization. *IEEE Trans Robot* 21(3):505–513. <https://doi.org/10.1109/TRO.2004.840905>
 43. Yu H, Liu Y, Yang T (2008) Closed-loop tracking control of a pendulum-driven cart-pole underactuated system. *Proc Inst Mech Eng I J Syst Control Eng* 222(2):109–125. <https://doi.org/10.1243/09596518JSCE460>
 44. Türker T, Görgün H, Cansever G (2012) Lyapunov's direct method for stabilization of the Furuta pendulum. *Turk J Electr Eng Comput Sci* 20(1):99–110. <https://doi.org/10.3906/elk-1007-653>
 45. Moreno-Valenzuela J, Aguilar-Avelar C (2018) Feedback linearization control of the Furuta Pendulum. *Motion control of underactuated mechanical systems intelligent systems, control and automation: science and engineering*, vol 88. Springer, New York. https://doi.org/10.1007/978-3-319-58319-8_5
 46. Liu Y, Yu H (2013) A survey of underactuated mechanical systems. *IET Control Theory Appl* 7(7):921–935. <https://doi.org/10.1049/iet-cta.2012.0505>
 47. MATLAB: R2017A (2017) Massachusetts: The MathWorks Inc

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.