IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 5 July 2023, accepted 19 July 2023, date of publication 24 July 2023, date of current version 1 August 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3298563

RESEARCH ARTICLE

Adaptive Back-Stepping Data-Driven Terminal Sliding-Mode Controller for Nonlinear MIMO Systems With Disturbance Observer

SINA NADERIAN[1] **AND MOHAMMAD FARROKHI**[2]

[1]School of Electrical Engineering, Iran University of Science and Technology, Tehran 16846-13114, Iran
[2]Center of Excellence for Modelling and Control of Complex Systems, Iran University of Science and Technology, Tehran 16846-13114, Iran

Corresponding author: Mohammad Farrokhi (farrokhi@iust.ac.ir)

ABSTRACT This paper presents an Adaptive Back-stepping Data-Driven Terminal Sliding Mode Controller (ABDTSMC) for non-affine MIMO systems with general disturbances including internal uncertainties and external disturbances. The proposed controller with new reaching law is used to reduce the controller's dependence on the mathematical model and eliminate the chattering phenomenon. Furthermore, to solve the problem of the coupling effect and to estimate the uncertainties and disturbances, the Disturbance Observer (DOB) based on neural network with adaptive weights is utilized. Afterwards, it is implied that the DOB variables are Uniformly Ultimately Bounded (UUB). Next, for the integrated controller, the closed-loop stability based on the Lyapunov and back-stepping theory is investigated and the new adaptive law for the Pseudo Jacobian Matrix (PJM) elements is derived. This method contributes to the reduction of complexity and conservatism, which facilitates analysis of the closed-loop stability. To evaluate performance of the controller, the proposed method is applied to a 2-DOF robot manipulator. The simulation results are compared with Model-Free Adaptive Sliding-Mode Controller (MFASMC) and Model-Free Adaptive controller (MFAC), which are reported recently in related literature. The results demonstrate the precision of the tracking capability is significantly enhanced in the presence of time-varying disturbances. Moreover, the chattering phenomenon is successfully removed. In addition, the number of required data is significantly reduced. Finally, to show practicality of the proposed controller, it is applied to the 2-DOF laboratory manipulator.

INDEX TERMS Back-stepping, data-driven, disturbance observer, neural network, sliding mode control.

I. INTRODUCTION

The model-based Sliding-Mode Controller (SMC) is one of the most robust and popular nonlinear control methods that has been employed by researchers, recently. This method offers some advantages like simplicity of design, fast convergence rate, and robustness against disturbances and uncertainties. In addition, it can be used in combination with a wide range of controllers to overcome some control problems. For instance, to increase convergence speed, the terminal sliding surface is used in some works, e.g. [1], [2], and [3]. In addition, to create a tradeoff between disturbance rejection and decrease of chattering phenomenon the SMC with adaptive switching gain and adaptive controller parameter is employed in [4] and [5]. Furthermore, in some papers like [6], [7], and [8], to minimize the control effort, an optimized SMC is designed. However, designing a model-based SMC is straightforward, but it has some challenges. For instance, difficulties in the stability proof of the high-order SMCs. Therefore, some works like [9], [10], and [11], have focused on the solution of this problem. In these papers, the authors implied that finding a proper Lyapunov function for the stability analysis is not an easy task. Hence, they combined the SMC with the back-stepping strategy to relax conservatism and ease the stability proof.

The associate editor coordinating the review of this manuscript and approving it for publication was Alfeu J. Sguarezi Filho.

Because of the powerful ability of the Neural Network (NN) observers to approximate some unknown functions like disturbances and uncertainties, mentioned in [12] and [13], some papers like [14], [15], [16], and [17] have developed the NN-based SMCs. In these papers, a new model-based method based on Dynamic Surface Control (DSC) is given. Moreover, the proposed method is combined with a NN observer to estimate the uncertainties, disturbances, coupling effects, and unknown nonlinearities. One of the advantages of these papers is that no knowledge about the uncertain dynamics, disturbances, and uncertainties is required. Furthermore, the upper bound of the uncertainties and disturbances is not a decisive criterion for the closed-loop stability.

With the development of digital controllers, the use of discrete-time SMCs has grown recently. However, their design is directly related to the mathematical input-output or state-space model of the system. Nevertheless, in many cases, this model is not accessible or difficult to determine. Recently, the design of these controllers have been witnessed remarkable developments. The most significant one may be the controller design based on the data instead of the mathematical model. These types of controllers are called data-driven controllers. Unlike the model-based controllers, the data-driven controllers do not rely on the mathematical or the state-space models. In the data-driven controllers, engineers usually attempt omitting or decreasing dependency on the mathematical models and designing the controller using the input-output measured data.

The data-driven controllers can be divided into two general categories 1) Direct data-driven and 2) Indirect data-driven controllers. In the direct data-driven, the controller is designed only based on data measurement and does not have any direct or indirect dependency on the mathematical model. However, the stability analysis is more challenging. In contrast, the indirect data-driven controller is a combination of the model-based methods and the data-driven schemes; hence, the stability analysis and the controller design may be simpler. The most significant advantage of this controller is reducing the dependency of the controller on the mathematical model. In other words, this method operates based on measuring the input-output data, but it can use the mathematics of the model-based method, especially for stability analysis. For instance, the Data-Driven Sliding-Mode Controllers (DDSMCs) and Model Free Adaptive controllers (MFACs) can be classified into the indirect data-driven methods that do not have direct dependency on the mathematical model [18], [19], [20], [21], [22], [23]. The following references support the use of DDSMCs.

Discrete DDSMCs are designed based on input-output data obtained from the system. Therefore, it is claimed that they are more robust against internal as well as external disturbances as compared with the model-based SMCs [24], [25], [26], [27]. To eliminate dependency on the mathematical model and replace it with the data model, DDSMC uses dynamic linearization method. This method uses a virtual parameter named Pseudo Partial Derivative (PPD) (for the SISO systems) or a virtual matrix named Pseudo Jacobean Matrix (PJM) (for the MIMO systems). Hence, it does not have any dependency to the mathematical model of the systems. Hence, DDSMCs have some merits but suffer from some problems. The most important of them are the coupling effects in the controller design for MIMO systems, zero-convergence of the PPD or PJM elements, complexity of the stability analysis, difficulty of the controller design for the continuous-time systems, lack of proper method to adjust control parameters, and the need for a large amount of data to achieve accurate control of the system. The following papers have tried to solve or relax these problems.

In [24] and [25], to achieve a faster transient response and to suppress the effect of uncertainties, disturbances, and coupling effects, an adaptive sliding-mode data-driven controller for unknown MIMO nonlinear discrete systems is designed. To cope with the coupling effects, a new mathematical method has been proposed. First, the coupling effect is decoupled and is considered in the general disturbance term. Then, it is estimated with the help of an Extended State Observer (ESO). This method contributes to coupling elimination that decreases conservatism and increases robustness of the controller.

The combination of a model-free adaptive controller and a model-free adaptive sliding-mode controller as a new DDSMC is given in [26]. The main drawback of this work is using an attenuation coefficient in the model-free adaptive sliding-mode controller part to remove the chattering effect. The authors concentrated on the dynamic linearization method to increase the precision of the tracking. In this work, a Full-Form Dynamic Linearization (FFDL) as a data model is used which has good precision but is complicated as compared with the Compact Form Dynamic Linearization (CFDL) method. In [27], a DDSMC is designed based on a novel reaching law for the SISO nonlinear systems. This article has focused on a new general reaching law as a novelty. The authors imply that one of the main disadvantages in the previous methods is ignoring the decreasing reaching time and the quasi-sliding domain. To overcome this problem, their control signal contains optimal and exponential switching terms.

Another significant problem in the data-driven controllers is the convergence of the PPD or PJM elements towards zero. In this regard, some papers have suggested different solutions. For example, adding the positive definite matrix or positive constant has been given in [24] and [25]. However, this method has its own drawback in determining the exact value of the PJM elements to control MIMO systems. Some works have tried to estimate the elements of PPD or PJM. In [28], an adaptive observer is proposed to calculate the PJM elements and a neural network is employed to estimate the parameters of the data model. In [29], a forecasting-based data-driven model-free adaptive sliding-mode control is performed for a combined spacecraft. In this article, a data model is considered as an FFDL. Therefore, a precise data model is reachable. However, the design of the controller is not straightforward, and the stability analysis is complicated.

Constrained input is one of the issues that has been studied in a few DDSMC papers. In [30], an anti-windup compensator is employed to consider the effects of the constraints on the input signal.

Although DDSMCs are identified as discrete-time controllers, the continuous-time model free SMCs based on the ultra-local model are introduced in some papers (e.g., [31] and [32]) as a subset of data-driven controllers. An important point of this method is that the differential equation of the system must be of the first order, which considerably restricts the application of the scheme.

DDSMCs with constraint tracking error has been reported in [33], [34], [35], [36], [37], and [38], where the authors have developed a new method, in which the tracking error converges to a predefined zone around of the origin. One of the main drawbacks of this method is complex stability analysis leading to conservative performance.

According to the aforementioned benefits of the DDSMCS, its applications on real systems has grown recently. For instance, in [39], [40], [41], and [42] the DDSMCs have been applied to inspection robots, rigid robot manipulators, under-actuated marine vessels, and cyber-physical systems against a class of actuator attacks. However, in these papers, there are no proper methods to adjust the controller parameters, which is another common problem among data-driven controllers.

As stated in the previous works (e.g., [21] and [22]), it seems that one of the best controllers for the non-affine and complicated system is the indirect data-driven controller. Since indirect data-driven controller does not have stability problem and can decrease dependency on the mathematical model of the system. Moreover, it can provide better robustness against uncertainties and external disturbances.

Based on the aforementioned literature review, the following statements are noteworthy about the DDSMCs:

- The DDSMCs cannot remove the chattering phenomenon by itself. As a result, researchers usually use new reaching laws, adaptive parameters in the sliding surface, or adaptive switching gains to reduce or eliminate it.
- The word *Adaptive* in the data-driven literature refers to PPD or PJM adaptation laws.
- The stability analysis of DDSMCs is more complex and conservative, because it consists of two steps: a) boundedness of the PPD or PJM parameters and b) stability of the closed-loop system.
- Another problem that exists in the DDSMCs is the *zero convergence* of PPD or PJM elements during the adaptation process leading to a singularity in the control signal. Thus, researchers usually apply additional matrices or parameters that must be accurately determined.

In this manuscript, Adaptive Back-stepping Data-Driven Terminal Sliding Mode Controller (ABDTSMC) with a Neural Network Disturbance Observer (NNDOB) for nonlinear MIMO systems is proposed. The contributions of the proposed method can be expressed as follows:

- Less dependency of the controller on the mathematical model of the system using a novel adaptation method for the PJM parameters.
- As compared with similar methods in literature (e.g., [24], [25], [26]), the PJM elements are estimated based on an Adaptive Sliding-Mode Observer (ASMO), while in the previous works, the PJM elements are calculated based on the rules given in [19]. The main problem with using this rule is the complexity of the stability analysis, which will be relaxed in this paper.
- To overcome the chattering phenomenon, a new reaching law based on the saturation function will be presented.
- To increase the convergence speed, the terminal sliding surface is employed.
- To control the system in the presence of time-varying disturbances and to compensate the missed data related to the general disturbances, a DOB using Radial Basis Function (RBF) neural network with adaptive weights is proposed.
- The data-driven dynamical model of the NNDOB will be presented. This model does not have any relationship with the mathematical model of the system.
- Using the designed NNDOB, no prior knowledge about the uncertainties and disturbances is needed. In addition, the upper bound of the uncertainties and disturbances does not have any effect on the stability analysis.
- The stability analysis of the closed-loop system with less conservatism based on the back-stepping and Lyapunov stability methods is provided.
- The proposed method can significantly reduce the number of required data, which makes it easier to implement.
- The proposed method is implemented on a 2-DOF laboratory manipulator.

This paper is organized as follows. Section II presents the problem formulation. Section III gives the data-driven mathematics and essential Lemmas. The proposed controller along with the NNDOB design, the PJM ASMO design, and the closed-loop stability are provided in Section IV. Simulation and experimental results are given in Section V. Conclusions are drawn in Section VI.

II. PROBLEM FORMULATION

Consider the following MIMO non-affine discrete-time system:

$$\begin{aligned}\mathbf{y}(k+1) = \mathbf{F}\big(&\mathbf{y}(k), \mathbf{y}(k-1), \ldots, \mathbf{y}(k-n_y),\\ &\times \mathbf{u}(k), \mathbf{u}(k-1), \ldots, \mathbf{u}(k-n_u),\\ &\times \mathbf{d}(k), \mathbf{d}(k-1), \ldots, \mathbf{d}(k-n_d)\big)\end{aligned} \quad (1)$$

where $\mathbf{F}(\cdot) = [F_1(\cdot), \ldots, F_n(\cdot)]^T \in \mathbf{R}^n$ denotes an unknown nonlinear vector-valued function, $\mathbf{u}(k) = [u_1(k), \ldots, u_n(k)]^T \in \mathbf{R}^n$ is the input, $\mathbf{y}(k) = [y_1(k), \ldots, y_n(k)] \in \mathbf{R}^n$ is the output, $\mathbf{d}(k) = [d_1(k), \ldots, d_n(k)] \in \mathbf{R}^n$ is a generalized disturbance vector including internal uncertainties and external disturbances, and n_y, n_u, and n_d are the unknown order of the outputs, inputs, and disturbances, respectively.

Remark 1: The data-driven controllers have no direct dependency on the system's mathematical model. As a result, system (1) is only used to generate the input-output data for simulation purposes.

In this paper, it is assumed that the following assumptions are satisfied by system (1).

Assumption 1: The partial derivatives of $F_i(\cdot)$ $(i = 1, \ldots, n)$ with respect to every control input signal and disturbance are continuous.

Remark 2: Assumption 1 is required for the controller design because the partial derivatives $(\partial F_i/\partial u(k))\Delta u(k)$ and $(\partial F_i/\partial d(k))\Delta d(k)$ appear in the data model calculations [19], [24], [25].

Assumption 2: The following generalized Lipschitz conditions are met by system (1):

$$\begin{cases} \| \mathbf{y}(k_1+1) - \mathbf{y}(k_2+1) \| \le b_1 \| \mathbf{u}(k_1) - \mathbf{u}(k_2) \| \\ \| \mathbf{y}(k_1+1) - \mathbf{y}(k_2+1) \| \le b_2 \| \mathbf{d}(k_1) - \mathbf{d}(k_2) \| \end{cases}$$
$$\begin{cases} k_1 \ne k_2, \;\; k_1, k_2 \ge 0 \\ \mathbf{u}(k_1) \ne \mathbf{u}(k_2), \;\; \mathbf{d}(k_1) \ne \mathbf{d}(k_2) \\ b_1, b_2 > 0 \end{cases} \tag{2}$$

Remark 3: Assumption 2 is equal to the Bounded-Input Bounded-Output (BIBO) stability condition for an open-loop system [19], [24], [25].

Remark 4: Although Assumptions 1 and 2 are restrictive, they are necessary for the data-driven formulation, which will be presented in Section III.

Assumption 3: System (1) must be controllable and observable such that in the presence of disturbances and uncertainties, there are bounded control input signals that can drive the outputs to the desired trajectories.

III. DATA-DRIVEN MATHEMATICS AND ESSENTIAL LEMMAS

To design the data-driven controller, a proper data model must be selected first. In this manuscript, based on the method of dynamic linearization, the CFDL method is employed. Hence, system (1) will be transformed into the data model based on the following lemma.

Lemma 1: If system (1) satisfies Assumptions 1 and 2, and $\|\Delta\mathbf{u}(k)\| \ne 0$, then there exists a time-varying matrix $\Phi_c(k) \in \mathbf{R}^{n\times n}$, which is called Pseudo Jacobean Matrix (PJM) such that system (1) can be defined by the following data model:

$$\Delta\mathbf{y}(k+1) = \Phi_c(k)\Delta\mathbf{u}(k) \tag{3}$$

where $\Delta\mathbf{y}(k+1) = \mathbf{y}(k+1) - \mathbf{y}(k)$, $\Delta\mathbf{u}(k) = \mathbf{u}(k) - \mathbf{u}(k-1)$ and PJM elements are bounded [19].

*Proof.*See [19].

Adding external disturbances $\mathbf{d}_1(k) \in \mathbf{R}^n$ to (3) yields

$$\Delta\mathbf{y}(k+1) = \Phi_c(k)\Delta\mathbf{u}(k) + \mathbf{d}_1(k). \tag{4}$$

Since the coupling effect is one of the main problems of the data-driven controller, in [15] and [16], a new technique is employed to eliminate the coupling effect and to consider it in the general disturbance term. In this technique, (4) is written as follows:

$$\Delta\mathbf{y}(k+1) = \Phi(k)\Delta\mathbf{u}(k) + \mathbf{d}(k) \tag{5}$$

where $\Phi(k) \in \mathbf{R}^{n\times n}$ and $\mathbf{d}(k) \in \mathbf{R}^n$ denote the diagonal PJM and the generalized disturbance, respectively, and are described as follows:

$$\Phi(k) = \begin{bmatrix} \Phi_{11}(k) & 0 & \cdots & 0 \\ 0 & \Phi_{22}(k) & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & \Phi_{nn}(k) \end{bmatrix} \tag{6}$$

$$\mathbf{d}(k) = \begin{bmatrix} d_{11}(k) \\ d_{12}(k) \\ \vdots \\ d_{1n}(k) \end{bmatrix} + \begin{bmatrix} 0 & \Phi_{12}(k) & \cdots & \Phi_{1n}(k) \\ \Phi_{21}(k) & 0 & \cdots & \Phi_{2n}(k) \\ \vdots & \vdots & \ddots & \vdots \\ \Phi_{n1}(k) & \Phi_{n2}(k) & \cdots & 0 \end{bmatrix} \begin{bmatrix} \Delta u_1(k) \\ \Delta u_2(k) \\ \vdots \\ \Delta u_n(k) \end{bmatrix}. \tag{7}$$

For simplicity of the stability analysis, the elements of diagonal matrix $\Phi(k)$ are defined as the following vector:

$$\begin{aligned} \Theta(k) &:= \left[\, \theta_1(k) \cdots \theta_n(k) \,\right]^T \\ &= \left[\, \Phi_{11} \cdots \Phi_{nn} \,\right]^T \in \mathbf{R}^n. \end{aligned} \tag{8}$$

Assumption 4: The PJM elements are positive.

Remark 5: In this paper, the generalized disturbance term will be estimated based on the neural network observer that will be explained in the next section.

Remark 6: Assumption 4 is theoretically necessary for the stability analysis presented in Section V (for the transient part). However, it does not create any restrictions on the application of the method, because due to a special property of the proposed method, when the sign of the PJM elements are initially selected positive, it will remain positive during operation of the closed-loop system (in the steady-state part). This claim will be proved in Corollary 2, in Section IV.

Lemma 2: If $\rho \in \mathbf{R}^{n\times n}$ is a positive definite matrix, then $-\mathbf{s}^T(k)\rho\,\mathrm{sat}(\mathbf{s}(k))$ is negative semidefinite.

Proof. Function $\mathrm{sat}(\mathbf{s}(k))$ is defined as

$$\mathrm{sat}(\mathbf{s}(k)) := \begin{cases} \mathrm{sgn}(\mathbf{s}(k)) & \|s(k)\| \ge 1 \\ \mathbf{s}(k) & \text{otherwise} \end{cases}. \tag{9}$$

Multiplying both sides of (9) by $\mathbf{s}^T(k)\rho$yields

$$\mathbf{s}^T(k)\rho\mathrm{sat}(\mathbf{s}(k)) = \begin{cases} \mathbf{s}^T(k)\rho\mathrm{sgn}(\mathbf{s}(k)) & \|s(k)\| \geq 1 \\ \mathbf{s}^T(k)\rho\mathbf{s}(k) & \text{otherwise} \end{cases} \tag{10}$$

Since $\mathbf{s}^T(k)\rho\mathrm{sgn}(\mathbf{s}(k))$ and $\mathbf{s}^T(k)\rho\mathbf{s}(k)$ are positive semidefinite, then $-\mathbf{s}^T(k)\rho\mathrm{sat}(\mathbf{s}(k))$ is negative semidefinite. ■

Lemma 2 will be used in the stability analysis in Section V.

IV. ADAPTIVE BACK-STEPPING DATA-DRIVEN TERMINAL SLIDING-MODE CONTROLLER WITH DOB

In this section, the main controller is designed based on the contributions described in Introduction.

The tracking error between the measured output and the desired output is considered as:

$$\mathbf{e}_1(k) = \mathbf{y}(k) - \mathbf{y}_\mathrm{d}(k) \tag{11}$$

where $\mathbf{y} \in \mathbf{R}^n$ and $\mathbf{y}_\mathrm{d} \in \mathbf{R}^n$ are the system output vector and the desired output vector, respectively. To design the controller, the tracking error in (11) must be as small as possible.

The Lyapunov function is defined as follows:

$$V_1(k) := \frac{1}{2}\mathbf{e}_1^T(k)\mathbf{K}_1\mathbf{e}_1(k) \tag{12}$$

where $\mathbf{K}_1 \in \mathbf{R}^{n\times n}$ is a diagonal positive definite matrix. The first difference of (12) can be given as

$$\begin{aligned} \Delta V_1(k+1) &= \mathbf{e}_1(k)\mathbf{K}_1\Delta\mathbf{e}_1(k+1) \\ &\quad + \frac{1}{2}\Delta\mathbf{e}_1^T(k)\mathbf{K}_1\Delta\mathbf{e}_1(k+1) \\ &= \frac{1}{2}\left(\mathbf{e}_1(k+1) + \mathbf{e}_1(k)\right)^T \mathbf{K}_1\Delta\mathbf{e}_1(k+1). \end{aligned} \tag{13}$$

By defining the second error vector as

$$\mathbf{e}_2(k) = \mathbf{K}_1\Delta\mathbf{e}_1(k+1) + \alpha(k) \tag{14}$$

where $\Delta\mathbf{e}_1(k+1) = \mathbf{e}_1(k+1) - \mathbf{e}_1(k)$ and $\alpha(k)$ is a virtual control, which is defined as

$$\alpha(k) := \mathbf{e}_1(k). \tag{15}$$

According to (14) and (15), we have $\mathbf{K}_1\Delta\mathbf{e}_1(k+1) = \mathbf{e}_2(k) - \mathbf{e}_1(k)$ and it results $\mathbf{e}_1(k+1) = \mathbf{k}_1^{-1}\mathbf{e}_2(k) + (\mathbf{I} - \mathbf{k}_1^{-1})\mathbf{e}_1(k)$. Therefore, by defining $\mathbf{e}_3(k) := [\,\mathbf{e}_1(k)\ \mathbf{e}_2(k)\,]^T \in \mathbf{R}^{2m\times 1}$, (13) can be rewritten as:

$$\Delta V_1(k+1) = -\mathbf{e}_3^T(k)\mathbf{F}\,\mathbf{e}_3(k) \tag{16}$$

where $\mathbf{F} = \begin{bmatrix} -\frac{1}{2}\left(\mathbf{K}_1^{-1} - 2\mathbf{I}\right) & \frac{1}{2}\left(\mathbf{K}_1^{-1} - 2\mathbf{I}\right) \\ \frac{1}{2}\mathbf{K}_1^{-1} & -\frac{1}{2}\mathbf{K}_1^{-1} \end{bmatrix} \in \mathbf{R}^{2m\times 2m}$.

Lemma 3: $\Delta V_1(k+1)$ is negative semidefinite provided that $\mathbf{F}$ is positive semidefinite.

Proof. Since $\det(\mathbf{F}) = 0$, therefore the only condition for $\mathbf{F}$ to be positive semidefinite can be $\mathbf{K}_1^{-1} - 2\mathbf{I} \leq \mathbf{0} \Rightarrow \mathbf{K}_1 \geq (1/2)\,\mathbf{I}$. ■

Lemma 3 will be used in the stability analysis that will be expressed in Theorem 2.

In this paper, the terminal sliding mode surface $\mathbf{s}(k) \in \mathbf{R}^n$ is defined as

$$\mathbf{s}(k) := \mathbf{K}_2\mathbf{e}_2^{p/q}(k-1) + \mathbf{K}_3\mathbf{e}_1(k) \tag{17}$$

where p and q are odd positive integers and $\mathbf{K}_2, \mathbf{K}_3 \in \mathbf{R}^{n\times n}$ are positive definite matrices, and $\mathbf{e}_1(k)$ and $\mathbf{e}_2(k)$ are the same as defined before in (11) and (14), respectively. One can get

$$\mathbf{s}(k+1) = \mathbf{K}_2\mathbf{e}_2^{p/q}(k) + \mathbf{K}_3\mathbf{e}_1(k+1) \tag{18}$$

where $\mathbf{e}_1(k+1)$ can be obtained as

$$\mathbf{e}_1(k+1) = \mathbf{y}(k+1) - \mathbf{y}_\mathrm{d}(k+1). \tag{19}$$

Substituting (5) in (19) yields

$$\mathbf{e}_1(k+1) = \mathbf{y}(k) + \Phi(k)\Delta\mathbf{u}(k) + \mathbf{d}(k) - \mathbf{y}_\mathrm{d}(k+1). \tag{20}$$

Since $\mathbf{d}(k) \in \mathbf{R}^n$ and $\Phi(k) \in \mathbf{R}^{n\times n}$ are unknown for the controller design, in the followings, they are estimated using a Disturbance Observer (DOB) based on neural networks and an Adaptive Sliding-Mode Observer (ASMO), respectively.

A. DOB DESIGN

According to Assumption 3, system (1) is observable. The DOB is designed using Radial-Basis Function (RBF) neural network. According to (5)–(8), the relationship between each input and output, considering a general disturbance, can be written as follows:

$$\Delta y_i(k+1) = \theta_i(k)\Delta u_i(k) + d_i(k),\ i = 1, \ldots, n. \tag{21}$$

The dynamical model of the DOB can be defined as

$$\begin{aligned} \Delta\beta_i(k+1) &= \theta_i(k)\Delta u_i(k) + \hat{d}_i(k) \\ &\quad + \kappa_i(y_i(k) - \beta_i(k)),\ i = 1, \ldots, n \end{aligned} \tag{22}$$

where κ_i is a positive constant and $\hat{d}_i(k)$ denotes the approximation of $d_i(k)$.

Remark 7: The dynamical model of the DOB in (22) is proposed based on the idea in [14], where the model has been presented for the continuous-time model-based observer, which has a direct connection with the mathematics of the systems. In this manuscript, this "direct connection" is eliminated by using a CFDL data model. In other words, the obtained model from the CFDL data is created and employed for the dynamical model of the DOB in [14].

By defining the auxiliary observer error as $q_i(k) := y_i(k) - \beta_i(k)$, one can get

$$\begin{aligned} \Delta q_i(k+1) &= \Delta y_i(k+1) - \Delta\beta_i(k+1) \\ &= d_i(k) - \hat{d}_i(k) - \kappa_i q_i(k). \end{aligned} \tag{23}$$

Based on the universal approximation theorem of the RBF networks [43], it can be claimed that

$$h_i(k) = \mathbf{W}_j^{*^T}\Psi_j(\mathbf{x}_i(k)),\quad j = 1, \ldots, n_h\ ,\ i = 1, \ldots, n \tag{24}$$

where n is the number of outputs of the network, n_h is the number of neurons in the hidden layer, $\mathbf{x}(k) = [\,\mathbf{y}(k)\ \mathbf{y}(k-1)\ \mathbf{u}(k)\ \mathbf{u}(k-1)\,] \in \mathbf{R}^{n\times 4}$is the input vector, and $h_i(k)$ is the approximation of $d_i(k)$ and it denotes the output of the RBF network. Moreover, $\Psi_j(\mathbf{x}_i(k)) \in \mathbf{R}^{n_h}$ is the activation vector function and $\mathbf{W}_j^* \in \mathbf{R}^{n_h}$ is the optimal weight vector that are defined as

$$\begin{cases} \Psi_j\left(\mathbf{x}_i(k)\right) = \left[\ \psi_1\left(\mathbf{x}_i(k)\right) \cdots \psi_{n_h}\left(\mathbf{x}_i(k)\right) \right]^T \\ \mathbf{W}_j^{*^T} = [\,w_{1i}^* \cdots w_{n_h i}^*\,]^T \end{cases} \tag{25}$$

where the activation function ψ_j is of Gaussian type, described by

$$\psi_j(\mathbf{x}_i) = \exp\left(-\frac{\left\|\mathbf{x}_i - \mu_j\right\|^2}{2\sigma_j}\right),\ j = 1, \ldots, n_h,$$
$$i = 1, \ldots, n \tag{26}$$

in which $\mu \in \mathbf{R}^{n_h\times 4}$ and $\sigma \in \mathbf{R}^{n_h}$ denote the center and width of the Gaussian function, respectively. Based on the universal approximation theorem, (23) can be rewritten as

$$\begin{aligned} \Delta q_i(k+1) &= \Delta y_i(k+1) - \Delta\beta_i(k+1) \\ &= h_i(k) - \hat{h}_i(k) - \kappa_i q_i(k) \end{aligned} \tag{27}$$

where $\hat{h}_i(k)$ is the approximation of $d_i(k)$ with adaptive output weights that will be estimated in the following Theorem.

Theorem 1: Consider (21) and (22) as the system and the DOB output equations, respectively, and the adaptive output weights of the RBF network as

$$\hat{\mathbf{W}}_i(k+1) = \hat{\mathbf{W}}_i(k) + \Lambda_i^{-1}\Psi_i\left(\mathbf{x}(k)\right) q_i(k),\ i = 1, \ldots, n. \tag{28}$$

where $\Lambda_i \in \mathbf{R}^{n_h\times n_h}$ is a positive definite diagonal matrix containing the learning coefficients and $\hat{\mathbf{W}}_i \in \mathbf{R}^{n_h}$ is the actual weight vector. Then, the disturbance observer error $(q_i(k))$ and the weights error $(\tilde{\mathbf{W}}(k) = \mathbf{W}^* - \hat{\mathbf{W}}(k))$ will be Uniformly Ultimately Bounded (UUB).

*Proof.*The Lyapunov function is defined as follows:

$$V_2(k) := \frac{1}{2}q_i^2(k) + \frac{1}{2}\tilde{\mathbf{W}}_i^T(k)\Lambda_i\tilde{\mathbf{W}}_i(k). \tag{29}$$

The first difference of (29) is

$$\begin{aligned} \Delta V_2(k+1) = q_i(k)\Delta q_i(k+1) &+ \frac{1}{2}\left(\Delta q_i(k+1)\right)^2 \\ &+ \tilde{\mathbf{W}}_i^T(k)\Lambda_i\Delta\tilde{\mathbf{W}}_i(k+1) \\ &+ \frac{1}{2}\Delta\tilde{\mathbf{W}}_i^T(k+1)\Lambda_i\Delta\tilde{\mathbf{W}}_i(k+1) \end{aligned} \tag{30}$$

Substituting $\Delta\tilde{\mathbf{W}}(k) = -\Delta\hat{\mathbf{W}}(k)$and (27) in (30) yields

$$\begin{aligned} \Delta V_2(k+1) = q_i(k)&\left(h_i(k) - \hat{h}_i(k) - \kappa_i q_i(k)\right) \\ &+ \frac{1}{2}\left(\Delta q_i(k+1)\right)^2 - \tilde{\mathbf{W}}_i^T(k)\Lambda_i\Delta\hat{\mathbf{W}}_i(k+1) \\ &+ \frac{1}{2}\Delta\hat{\mathbf{W}}_i^T(k+1)\Lambda_i\Delta\hat{\mathbf{W}}_i(k+1). \end{aligned} \tag{31}$$

By assuming $\hat{h}_i(k)$ as the output of RBFNN disturbance observer with adaptive output weights, it yields

$$\hat{h}_i(k) = \hat{\mathbf{W}}_i^T(k)\Psi_i\left(\mathbf{x}(k)\right). \tag{32}$$

Consequently the error between $h_i(k)$ and $\hat{h}_i(k)$ can be obtained as

$$h_i(k) - \hat{h}_i(k) = \tilde{\mathbf{W}}_i^T(k)\Psi_i(x(k)). \tag{33}$$

Then, by substituting (33) in (31), we have

$$\begin{aligned} \Delta V_2(k+1) = q_i(k)&\left(\tilde{\mathbf{W}}_i^T(k)\Psi_i\left(\mathbf{x}(k)\right) - \kappa_i q_i(k)\right) \\ &+ \tilde{\mathbf{W}}_i^T(k)\Lambda_i\Delta\hat{\mathbf{W}}(k+1)\,\frac{1}{2}\left(\Delta q_i(k+1)\right)^2 \\ &+ \frac{1}{2}\Delta\hat{\mathbf{W}}_i^T(k+1)\Lambda_i\Delta\hat{\mathbf{W}}_i(k+1). \end{aligned} \tag{34}$$

After simplifying, (34) can be rewritten as

$$\begin{aligned} \Delta V_2(k+1) = \tilde{\mathbf{W}}_i^T(k)&\left(\Psi_i\left(\mathbf{x}(k)\right)q_i(k) - \Lambda_i\Delta\hat{\mathbf{W}}_i(k+1)\right) \\ &- \kappa q_i^2(k) + \frac{1}{2}\left(\Delta q_i(k+1)\right)^2 \\ &+ \frac{1}{2}\Delta\hat{\mathbf{W}}_i^T(k+1)\Lambda_i\Delta\hat{\mathbf{W}}_i(k+1). \end{aligned} \tag{35}$$

Using the adaptation law (28) in (35) yields

$$\begin{aligned} \Delta V_2(k+1) = -\kappa q_i^2(k) &+ \frac{1}{2}\left(q_i(k+1) - q_i(k)\right)^2 \\ &+ \frac{1}{2}q_i(k)\Psi_i^T\left(\mathbf{x}(k)\right)\Lambda_i^{-1}\Lambda_i\Lambda_i^{-1}\Psi_i\left(\mathbf{x}(k)\right)q_i(k). \end{aligned} \tag{36}$$

Simplifying (36) results in

$$\begin{aligned} &\Delta V_2(k+1) \\ &= -\left(-\frac{1}{2}\Psi_i^T\left(\mathbf{x}_i(k)\right)\Lambda_i^{-1}\Psi_i\left(\mathbf{x}_i(k)\right) + \kappa_i + \frac{1}{2}\right)q_i^2(k) \\ &\quad + \frac{1}{2}q_i^2(k+1) - q_i(k+1)q_i(k). \end{aligned} \tag{37}$$

According to Young's inequality

$$q_i(k+1)q_i(k) \le \frac{1}{2}q_i^2(k+1) + \frac{1}{2}q_i^2(k), \tag{38}$$

equation (37) can be rewritten as

$$\begin{aligned} &\Delta V_2(k+1) \\ &= -\left(-\frac{1}{2}\Psi_i^T\left(\mathbf{x}_i(k)\right)\Lambda_i^{-1}\Psi_i\left(\mathbf{x}_i(k)\right) + \kappa_i + \frac{1}{2}\right)q_i^2(k) \\ &\quad + \frac{1}{2}q_i^2(k+1) - \frac{1}{2}q_i^2(k+1) - \frac{1}{2}q_i^2(k). \end{aligned} \tag{39}$$

Hence, (39) becomes

$$\begin{aligned} &\Delta V_2(k+1) \\ &= -\left(-\frac{1}{2}\Psi_i^T\left(\mathbf{x}_i(k)\right)\Lambda_i^{-1}\Psi_i\left(\mathbf{x}_i(k)\right) + \kappa_i\,\frac{1}{2} + 1\right)q_i^2(k). \end{aligned} \tag{40}$$

By defining

$$\Omega_i(k) := \Psi_i^T\left(\mathbf{x}_i(k)\right)\Lambda_i^{-1}\Psi_i\left(\mathbf{x}_i(k)\right)\quad (i = 1, \ldots, n),$$

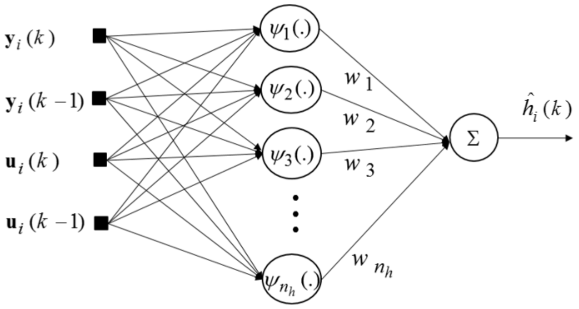


FIGURE 1. Block diagram of NNDOB

the UUB condition for $\Delta V_2(k)$ to be a negative definite function can be written as

$$-\frac{1}{2}\Omega_i(k)+\kappa_i+1>0 \Rightarrow \Omega_i(k)<2\kappa_i+2. \tag{41}$$

Therefore, if condition (41) holds, then ΔV_2 will be negative definite and the observer error and weights error will be UUB. ■

The block diagram of the Neural Network-based DOB (NNDOB) is shown in FIGURE. 1.

Next, the estimated disturbance based on the NNDOB is substituted into (18) and the result is substituted into (20). It gives

$$\mathbf{s}(k+1)=\mathbf{K}_2\mathbf{e}_2^{p/q}(k) \\ +\mathbf{K}_3\left(\mathbf{y}(k)+\Phi(k)\Delta\mathbf{u}(k)\ +\hat{\mathbf{d}}(k)-\mathbf{y}_{\mathrm{d}}(k+1)\right). \tag{42}$$

The reaching law is defined as

$$\Delta\mathbf{s}(k+1):=-\mathbf{s}(k)\frac{\Theta^T\lambda}{\|\mathbf{s}(k)\|^2}\mathbf{s}(k)-\rho\mathrm{sat}(\mathbf{s}(k)) \tag{43}$$

which can be rewritten as

$$\mathbf{s}(k+1)=-\mathbf{s}(k)\frac{\Theta^T\lambda-\mathbf{s}^T(k)}{\|\mathbf{s}(k)\|^2}\mathbf{s}(k)-\rho\mathrm{sat}(\mathbf{s}(k)) \tag{44}$$

where $\rho\in\mathbf{R}^{n\times n}$ is a positive definite diagonal matrix and denotes the switching matrix. By substituting (44) into (42) and simplifying it, the control input signal can be obtained as

$$\mathbf{u}(k)=\mathbf{u}(k-1)+(\mathbf{K}_3\Phi(k))^{-1}\Big(\mathbf{K}_3\mathbf{y}_{\mathrm{d}}(k+1)-\mathbf{K}_2\mathbf{e}_2^{p/q}(k) \\ -\frac{\Theta^T\lambda-\mathbf{s}^T(k)}{\|\mathbf{s}(k)\|^2}\mathbf{s}(k) \\ -\rho\mathrm{sat}(\mathbf{s}(k))-\mathbf{K}_3\mathbf{y}(k)-^T\mathbf{K}_3\hat{\mathbf{d}}(k)\Big). \tag{45}$$

In (45), the PJM is unknown. Therefore, in the following section, the ASMO is presented to estimate the PJM elements.

B. PJM ASMO DESIGN AND CLOSED-LOOP STABILITY

Theorem 2: If system (1) satisfies Assumptions 1– 4 and the following adaptation law is employed to estimate the PJM elements, then the closed-loop system is stable:

$$\hat{\Theta}(k+1)=\hat{\Theta}(k)-\Gamma^{-1}\lambda\,\mathbf{s}(k) \tag{46}$$

where $\Gamma\in\mathbf{R}^{n\times n}$ and $\lambda\in\mathbf{R}^{n\times n}$ are diagonal positive definite matrices, $\hat{\Theta}(k)$ is an estimation of $\Theta(k)$ that was defined in (8), and $\mathbf{s}(k)$ is the terminal sliding surface.

*Proof.*A comprehensive Lyapunov function is defined as follows:

$$V_3(k):=\frac{1}{2}\mathbf{s}^T(k)\mathbf{s}(k)+\frac{1}{2}\tilde{\Theta}^T(k)\Gamma\tilde{\Theta}(k) \\ +V_1(k)+V_2(k) \tag{47}$$

where $\mathbf{s}(k)$ is the terminal sliding surface and $\tilde{\Theta}(k)=\Theta(k)-\Theta$. The first difference of (47) can be obtained as follow:

$$\Delta V_3(k+1)=\mathbf{s}^T(k)\Delta\mathbf{s}(k+1)+\frac{1}{2}\Delta\mathbf{s}^T(k+1)\Delta\mathbf{s}(k+1) \\ +\tilde{\Theta}^T(k)\Gamma\Delta\hat{\Theta}(k+1) \\ +\frac{1}{2}\Delta\hat{\Theta}^T(k+1)\Gamma\Delta\hat{\Theta}(k+1) \\ +\Delta V_1(k+1)+\Delta V_2(k+1) \tag{48}$$

where $\Delta\mathbf{s}(k+1)=\mathbf{s}(k+1)-\mathbf{s}(k)$. According to (43), (48) becomes

$$\Delta V_3(k)=-\mathbf{s}^T(k)\left(\frac{\mathbf{s}(k)\Theta^T\lambda}{\|\mathbf{s}(k)\|^2}\mathbf{s}(k)+\rho\mathrm{sat}(\mathbf{s}(k))\right) \\ +\frac{1}{2}\Delta\mathbf{s}^T(k+1)\Delta\mathbf{s}(k+1) \\ +(\hat{\Theta}(k)-\Theta)^T\Gamma\Delta\hat{\Theta}(k+1) \\ +\frac{1}{2}\Delta\hat{\Theta}^T(k+1)\Gamma\Delta\hat{\Theta}(k+1)\ +\Delta V_1(k+1) \\ +\Delta V_2(k+1). \tag{49}$$

With the help of the adaptation law in (46), (49) can be written as

$$\Delta V_3(k)=-\mathbf{s}^T(k)\rho\mathrm{sat}(\mathbf{s}(k))+\frac{1}{2}\Delta\mathbf{s}^T(k+1)\Delta\mathbf{s}(k+1) \\ -\frac{1}{2}\left(\tilde{\Theta}^T(k)+\hat{\Theta}^T(k+1)\right)\lambda\mathbf{s}(k) \\ +\Delta V_1(k+1)+\Delta V_2(k+1). \tag{50}$$

Substituting (46) in (50) yields

$$\Delta V_3(k+1)=-\mathbf{s}^T(k)\rho\mathrm{sat}(\mathbf{s}(k))+\frac{1}{2}\Delta\mathbf{s}^T(k+1)\Delta\mathbf{s}(k+1) \\ -\tilde{\Theta}^T(k)\lambda\mathbf{s}(k)-\frac{1}{2}\mathbf{s}^T(k)\lambda\Gamma^{-1}\lambda\mathbf{s}(k) \\ +\Delta V_1(k+1)\ +\Delta V_2(k+1). \tag{51}$$

To ensure stability of the closed-loop system, the first difference of the Lyapunov function must be negative semidefinite. As we know $-(1/2)\mathbf{s}^T(k)\lambda\Gamma^{-1}\lambda\mathbf{s}(k)$ is negative semidefinite. In addition, according to Lemma 2, $-\mathbf{s}(k)\rho\mathrm{sat}(\mathbf{s}(k))$ is negative semidefinite. Therefore, the stability conditions can be given as follows:

1. $\Delta V_1(k)$ must be negative semidefinite. Therefore, according to Lemma 3, the first stability condition can be described as

$$\mathbf{K}_1^{-1}-2\mathbf{I}\leq\mathbf{0}\Rightarrow\mathbf{K}_1\geq\frac{1}{2}\mathbf{I}. \tag{52}$$

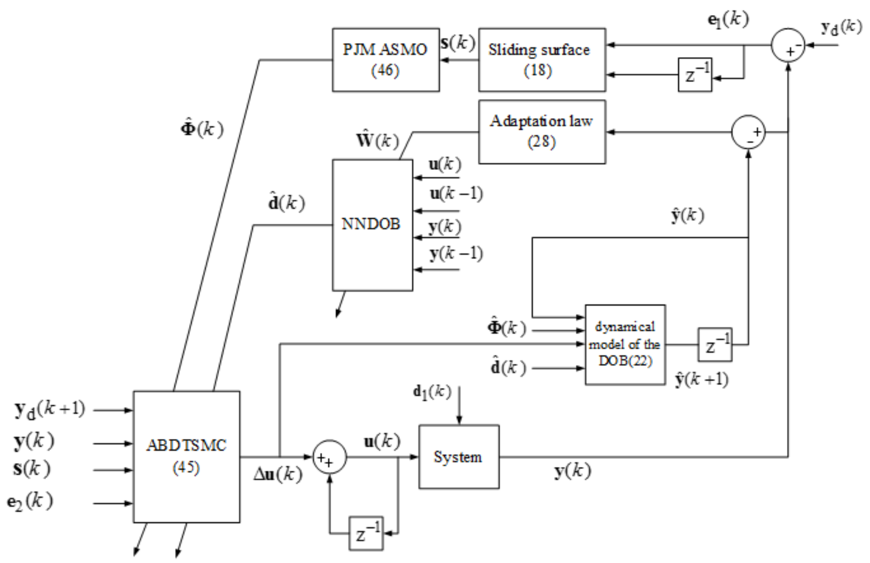


FIGURE 2. Block diagram of the proposed controller.

2. $\Delta V_2(k)$ must be negative semidefinite. Hence, according to Theorem 1, the second stability condition is

$$\Omega_i(k) \le 2\kappa_i + 2 \quad (i = 1, \ldots, n) \tag{53}$$

where $\Omega_i(k)$ is the same as in (41).

3. For the rest of the terms in (51) to be negative semidefinite, we must have

$$\|\Delta\mathbf{s}(k+1)\|^2 \le 2\hat{\Theta}^T(k)\lambda\mathbf{s}(k) + 2\mathbf{s}^T(k)\rho\,\mathrm{sat}(\mathbf{s}(k)) + \mathbf{s}^T(k)\lambda\Gamma^{-1}\lambda\mathbf{s}(k) \tag{54}$$

where $\|\cdot\|$ is the L_2 norm.

Consequently, according to conditions (52)–(54), the closed-loop system is stable. ■

Corollary 1: The tracking error and the sliding surface converge to zero if the closed-loop system is stable under conditions (52)–(54).

Proof. According to Theorem 2, $V_3(k)$ is a non-increasing function with respect to time, because $V_3(k)$ is positive definite and $V_3(k)$ is negative semidefinite under condition (52)–(54). Since $V_3(k)$ comprises of $V_1(k)$; hence, $V_1(k)$ is also non-increasing and the tracking error $\mathbf{e}_1(k)$ converges to zero (i.e., $\lim_{k\to\infty}\mathbf{e}_1(k) \to 0$). In addition, it was shown that $-\mathbf{e}_3^T(k)\mathbf{F}\,\mathbf{e}_3(k)$ is negative semidefinite under condition (52); hence, $\mathbf{e}_2(k)$ will also converge to zero (i.e., $\lim_{k\to\infty}\mathbf{e}_2(k) \to 0$). Therefore, for the sliding surface, we have the following result:

$$\begin{aligned}\lim_{k\to\infty}\mathbf{s}(k) &= \lim_{k\to\infty}\mathbf{K}_2\mathbf{e}_2^{p/q}(k-1) + \lim_{k\to\infty}\mathbf{K}_3\mathbf{e}_1(k)\\ &= \mathbf{K}_2\lim_{k\to\infty}\mathbf{e}_2^{p/q}(k-1) + \mathbf{K}_3\lim_{k\to\infty}\mathbf{e}_1(k)\\ &\xrightarrow[\lim_{k\to\infty}\mathbf{e}_2(k)\to 0]{\lim_{k\to\infty}\mathbf{e}_1(k)\to 0} \lim_{k\to\infty}\mathbf{s}(k) \to 0\end{aligned} \tag{55}$$

This concludes the proof.

Corollary 2: If the sliding surface converges to zero, then the PJM elements will be bounded.

*Proof.*In accordance with Corollary 1, the sliding surface converges to zero. Thus, according to (46) we will have:

$$\begin{aligned}\lim_{k\to\infty}\hat{\Theta}(k) &= \lim_{k\to\infty}\hat{\Theta}(k-1) - \lim_{k\to\infty}\Gamma^{-1}\lambda\,\mathbf{s}(k)\\ &= \lim_{k\to\infty}\hat{\Theta}(k-1) - \Gamma^{-1}\lambda\lim_{k\to\infty}\mathbf{s}(k)\\ &\xrightarrow{\lim_{k\to\infty}\mathbf{s}(k)\to 0} \lim_{k\to\infty}\hat{\Theta}(k) = \lim_{k\to\infty}\hat{\Theta}(k-1)\end{aligned} \tag{56}$$

Hence, the PJM elements are bounded. ■

Corollary 3: If the closed-loop system is stable, then NNDO weights will be bounded.

Proof. Pursuant to Corollary 1, $V_3(k)$ is a non-increasing function. Moreover, $V_3(k)$ includes $V_2(k)$. As a result $V_2(k)$ is also non-increasing and negative semidefinite function under condition (53). Therefore, according to (40), the disturbance observer error will converge to zero (i.e., $\lim_{k\to\infty} q_i(k) \to 0$) and it results

$$\begin{aligned}\lim_{k\to\infty}\hat{\mathbf{W}}_i(k+1) &= \lim_{k\to\infty}\hat{\mathbf{W}}_i(k) + \lim_{k\to\infty}\Lambda_i^{-1}\Psi_i(\mathbf{x}(k))\,q_i(k)\\ &= \lim_{k\to\infty}\hat{\mathbf{W}}_i(k) + \Lambda_i^{-1}\lim_{k\to\infty}\Psi_i(\mathbf{x}(k))\,q_i(k)\\ &\xrightarrow{\lim_{k\to\infty} q_i(k)\to 0} \lim_{k\to\infty}\hat{\mathbf{W}}_i(k+1) \to \lim_{k\to\infty}\hat{\mathbf{W}}_i(k)\end{aligned} \tag{57}$$

This concludes the proof. ■

The block diagram of the proposed controller is illustrated in FIGURE 2.

The algorithm of the proposed method, comprising of the ABDTSMC and the NNDOB is as follows:

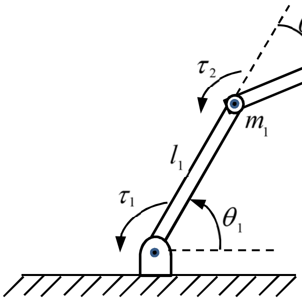


FIGURE 3. 2-DOF robotic manipulator.

1. Initialize: Select positive initial values for $\mathbf{u}(0)$, $\mathbf{y}(0)$, μ_j, σ_j and $\Theta(0)$

2. Initialize Controller parameters: Select appropriate values for $\mathbf{K}_1, \mathbf{K}_2, \mathbf{K}_3, \Gamma, \lambda, \rho, \kappa_i$, and Λ.

3. Select $k = 1$.

4. Errors: Calculate $\mathbf{e}_1(k)$ and $\mathbf{e}_2(k)$ using (11) and (14), respectively.

5. Sliding surface: Calculate the sliding surface using (17).

6. Adaptive parameters: Obtain the NNDO weights and the PJM elements using (28) and (46), respectively.

7. Control input: Calculate the control input using (45).

8. Output: Applied the control input to the discretized system and obtain the output.

9. $k = k + 1$ and go back to step 4.

Remark 8: The optimal value for parameters $\mathbf{K}_1$, $\mathbf{K}_2, \mathbf{K}_3, \Gamma, \lambda, \rho, \kappa_i$ and Λ_i can be found using any optimization method like the PSO algorithm, employed in this paper. This algorithm is used offline and in every iteration, the stability conditions (52)–(54) are completely checked. If the conditions are not satisfied the algorithm is repeated, otherwise the algorithm ends. Moreover, other parameters (i.e., μ_i and σ_j) are adjusted using the k-means method. Other parameters (i.e., initial values of the PJM elements) can be obtained using some methods like trial-and-error.

V. SIMULATING AND EXPERIMENTAL RESULTS

The performance of the proposed method is evaluated in simulations by applying it to a 2-Degree-Of-Freedom (DOF) robotic manipulator (FIGURE. 3).

By neglecting the effects of the friction force, the dynamic equations of this system are given as [35]:

$$\mathbf{E}(\theta)\ddot{\theta} + \mathbf{F}(\theta)\dot{\theta} + \mathbf{G}(\theta) = \tau \tag{58}$$

where $\theta, \dot{\theta}, \ddot{\theta} \in \mathbf{R}^2$ denote the position, velocity, and acceleration of the joints, respectively, and $\mathbf{E} \in \mathbf{R}^{2\times 2}$, $\mathbf{F} \in \mathbf{R}^2$, $\mathbf{G} \in \mathbf{R}^2$, and $\tau \in \mathbf{R}^2$ express the inertia matrix, the Coriolis, the gravitational, and the torque vectors, respectively, given by, as in (59)–(61), shown at the bottom of the next page, where m_i and l_i are the mass and the length of the ith link, respectively.

In this paper, the joints positions $(\theta(t))$ and the torque applied to the joints $(\tau(t))$ are regarded as the outputs and the inputs of the system, respectively.

To obtain input-output data from the system, (58) is discretized by a 1 msec. The simulating results of the proposed method are compared with the MFAC method in [23] and the MFASMC method in [26]. For the closed-loop operation, the best result of the MFASMC and MFAC methods are obtained with 1 msec. and 0.2 msec. sampling time, respectively. While the sample time of the proposed ABDTSMC method is 16 msec. To evaluate the tracking ability, the desired trajectory is regarded as follow:

$$\mathbf{y}_{\mathrm{d}}(k) = \begin{bmatrix} \frac{3}{2} - \frac{\pi}{3}\sin\left((1-k)\pi\right) \\ \frac{3}{2} + \frac{\pi}{3}\sin\left((1-k)\pi\right) \end{bmatrix}. \tag{62}$$

The parameters of the system and controllers are presented in TABLE 1. As stated in Remark 8, the optimal values of some of the design parameters are calculated using the PSO algorithm for all three controllers (see TABLE 1). Moreover, the center (μ_j) and the widths (σ_j) of the Gaussian functions in the RBF network are determined using the k-means algorithm.

The following time-varying output disturbance is considered:

$$d_1 = \begin{cases} 0 & t < 1.5 \\ -0.07\cos(3\pi t) + 0.16 & 1.5 \le t < 3.5 \\ 0.09 & 3.5 \le t < 5 \end{cases}$$
$$d_2 = \begin{cases} 0 & t < 1.5 \\ 0.07\sin(2\pi t - \frac{\pi}{2}) + 0.1 & 1.5 \le t < 3.5 \\ 0.17 & 3.5 \le t < 5 \end{cases} \tag{63}$$

A. SIMULATION RESULTS

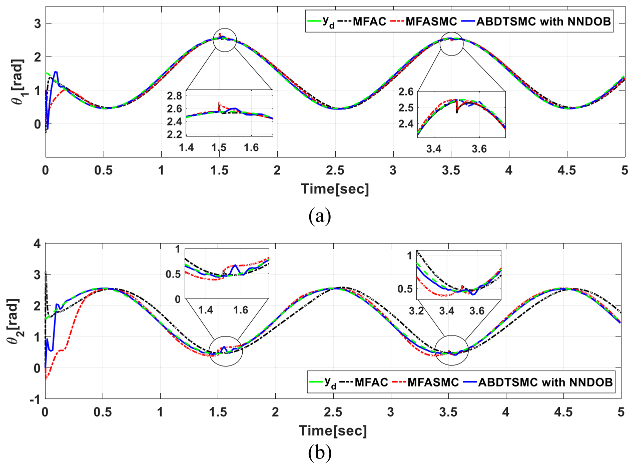
The simulation results are shown in FIGURES 4–13.

FIGURES 4 and 5 show the joints positions and the applied torques, respectively. As FIGURE 4 shows, the proposed ABDTSMC method has faster convergence rate and better tracking accuracy than the other data-driven controllers in various situations including the transient and steady states against external disturbances.

It should be emphasized that the sampling time of the proposed method is 16 msec. while for the MFASMC and MFAC it is 1 msec. and 0.2 msec., respectively. In other words, better performance of the ABDTSMC has been achieved with much fewer data than the MFASMC and the MFAC. Moreover, the data model of the proposed method has less complexity as compared with MFASMC and MFAC. In MFASMC and MFAC, the Full Form Dynamic Linearization (FFDL) method is employed, while the proposed controller is utilizing the CFDL method.

To better mark the differences between these controllers, the Mean of Absolute Error (MAE) is defined as

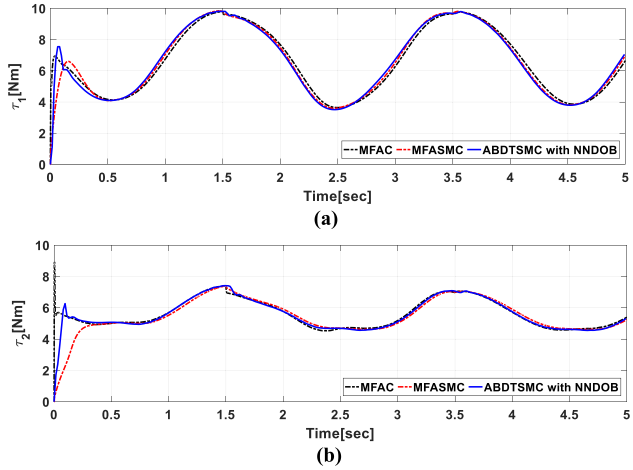
$$\mathrm{MAE}_{\mathrm{DDABTSMC}} = \frac{1}{313}\sum_{k=1}^{313} |\mathbf{e}(k)|$$



(a)

(b)

FIGURE 4. Joints position: (a) θ_1 and (b) θ_2.



(a)

(b)

FIGURE 5. Torque of each joint: (a) τ_1 and (b)τ_2.

$$\mathrm{MAE}_{\mathrm{MFASMC}} = \frac{1}{5000}\sum_{k=1}^{5000}|\mathbf{e}(k)|$$

$$\mathrm{MAE}_{\mathrm{MFAC}} = \frac{1}{25000}\sum_{k=1}^{25000}|\mathbf{e}(k)| \tag{64}$$

The results for the MAE and the maximum steady-state errors are presented in TABLE 2.

FIGURE 5 indicates that the chattering phenomenon is successfully eliminated by the ABDTSMC and MFASMC methods. In addition, FIGURE 5 illustrates that the control input effort is almost the same for all of the controllers, nevertheless, the ABDTSMC has better performance.

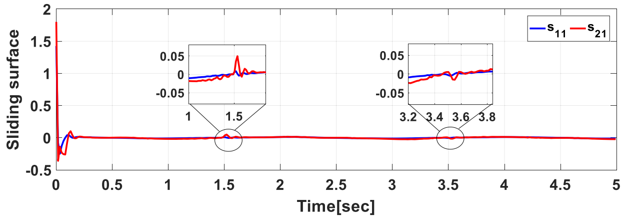


FIGURE 6. ABDTSMC sliding surfaces.

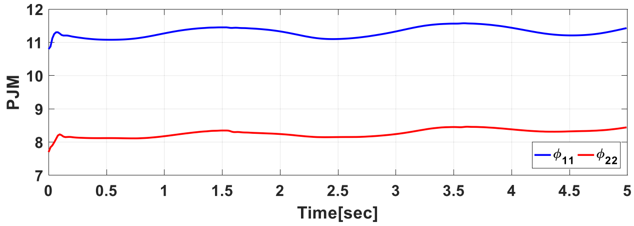


FIGURE 7. PJM components of DDBATSMC.

The sliding surfaces of the proposed methods is depicted in FIGURE 6. As Corollary 1 indicates, the sliding surface of the proposed method converges to zero with some minor fluctuations around the origin, due to the disturbances. The PJM elements, which are estimated by the new adaptation law in (46), are exhibited in FIGURE 7. As claimed in Corollary 2, the PJM components are bounded. FIGURE 7 confirms this assertion.

The first difference of the Lyapunov functions are presented in FIGURE 8. According to this figure, the first difference of the Lyapunov function remains negative semidefinite under conditions (52) to (54), at all time, except when the external disturbances occur. The stability conditions in (53) and (54) are illustrated in FIGURE 9, which indicates that all of the stability conditions are met except for a short time in the vicinity of 1.5 sec. and 3.5 sec. due to the external disturbances. The abrupt change in FIGURE 9 (a) is due to the end of data collection for the k-means algorithm. After the data collection, the center and widths of the RBF neural network are determined.

The estimation of disturbances and the adaptive weights of the DOB are shown in FIGUREs 10 and FIGURE 11,

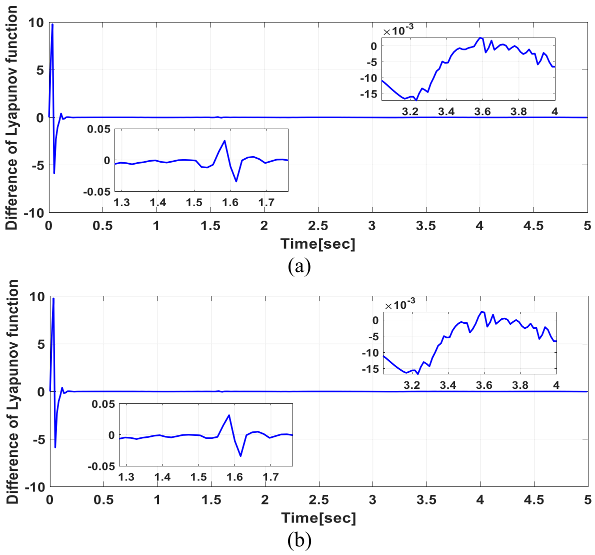
$$\mathbf{E}(\theta) = \begin{bmatrix} \begin{pmatrix} m_2 l_2^2 + 2m_2 l_1 l_2 \cos(\theta_2) \\ +(m_1 + m_2) l_1^2 \end{pmatrix} & m_2 l_2^2 + m_2 l_1 l_2 \cos(\theta_2) \\ m_2 l_2^2 + m_2 l_1 l_2 \cos(\theta_2) & m_2 l_2^2 \end{bmatrix} \tag{59}$$

$$\mathbf{F}\left(\theta, \dot{\theta}\right)\dot{\theta} = \begin{bmatrix} -m_2 l_1 l_2 \sin(\theta_2) q^2 - 2m_2 l_1 l_2 \sin(\theta_2)\dot{\theta}_1\dot{\theta}_2 \\ m_2 l_1 l_2 \sin(\theta_2)\dot{\theta}_2^2 \end{bmatrix} \tag{60}$$

$$\mathbf{G}(\theta) = \begin{bmatrix} m_2 l_2 g \cos(\theta_1 + \theta_2) + (m_1 + m_2)\, l_1 g \cos(\theta_1) \\ m_2 l_2 g \cos(\theta_1 + \theta_2) \end{bmatrix} \tag{61}$$

TABLE 1. System and controller parameters.

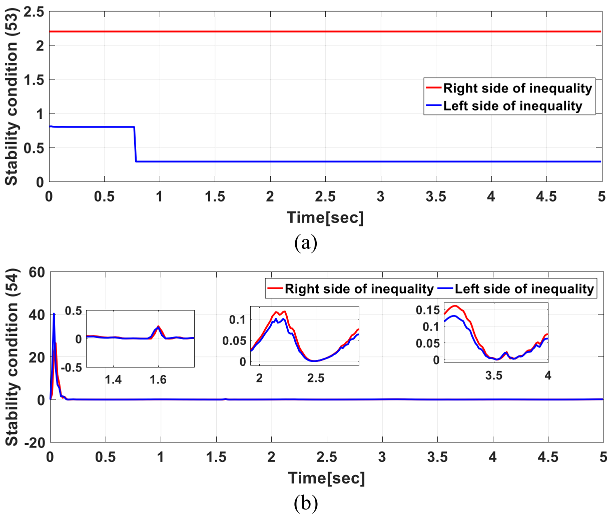
Parameter	VALUE	REMARK
m_1, m_2	1kg, 1kg	Mass of links
l_1, l_2	0.6 m, 0.5 m	Length of links
$\mathbf{\Theta}(1)$	$\begin{bmatrix} 10.8 & 7.7 \end{bmatrix}^T$	Initial value of the PJM elements
$\boldsymbol{\lambda}$	$\begin{bmatrix} 1e-5 & 0 \\ 0 & 0.8e-5 \end{bmatrix}$	ABDTSMC coefficients ($\mathbf{K}_1$, $\mathbf{K}_2$, $\mathbf{K}_3$, $\mathbf{\Gamma}$, $\boldsymbol{\lambda}$ and $\boldsymbol{\rho}$ are adjusted by PSO algorithm)
$\mathbf{\Gamma}$	$\begin{bmatrix} 1e-5 & 0 \\ 0 & 2e-5 \end{bmatrix}$	
$\mathbf{K}_1, \mathbf{K}_2$	$\begin{bmatrix} 0.5 & 0 \\ 0 & 5 \end{bmatrix}, \begin{bmatrix} 1e-4 & 0 \\ 0 & 1e-4 \end{bmatrix}$	
$\mathbf{K}_3, \boldsymbol{\rho}, p, q$	$\begin{bmatrix} 0.3 & 0 \\ 0 & 0.24 \end{bmatrix}, \begin{bmatrix} 25 \\ 9 \end{bmatrix}, 7, 9$	
n_{h_1}, n_{h_2}	2, 2	NNDOB coefficients (κ_i and $\mathbf{\Lambda}_i$ are adjusted by PSO, $\boldsymbol{\mu}_i$ and $\boldsymbol{\sigma}_i$ are adjusted by k-means method)
$\boldsymbol{\mu}_1$	$\begin{bmatrix} 5.1 & 5.1 & 2.3 & 2.2 \end{bmatrix}$	
$\boldsymbol{\mu}_2$	$\begin{bmatrix} 1.51 & 0.82 & 0.4 & 2.1 \end{bmatrix}$	
σ_1, σ_2	3.06, 8	
$\mathbf{\Lambda}_1, \mathbf{\Lambda}_2$	$\begin{bmatrix} 0.36 & 0 \\ 0 & 0.42 \end{bmatrix}, \begin{bmatrix} 0.35 & 0 \\ 0 & 0.44 \end{bmatrix}$	
κ_1, κ_2	0.1, 0.1	
$\mathbf{\Phi}_1(1) = \mathbf{\Phi}_1(2) = \mathbf{\Phi}_2(1) = \mathbf{\Phi}_2(2)$	$\begin{bmatrix} 14.5 & 0 \\ 0 & 14.5 \end{bmatrix}$	Initial value of the PJM in MFASMC in [26]
μ_1, η_{11}	0.1, 0.1	MFASMC coefficients in [26] (adjusted by PSO algorithm)
$\mathbf{q}$	$\begin{bmatrix} 0.2e4 & 0 \\ 0 & 0.4e3 \end{bmatrix}$	
λ_1	1e-3	
$\boldsymbol{\rho}$	$\begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix}$	
$\boldsymbol{\varepsilon}_1, \boldsymbol{\gamma}_1$	$\begin{bmatrix} 30 \\ 15 \end{bmatrix}, \begin{bmatrix} 0.6 & 0 \\ 0 & 0.6 \end{bmatrix}$	
$\eta, \boldsymbol{\rho},$	$5.9, \begin{bmatrix} 1e-2 & 1e-2 & 1 & 1e-2 \end{bmatrix},$	MFAC coefficients in [23] (adjusted by PSO algorithm)
μ, λ	$1e-4, 200$	
$\mathbf{\Phi}_f(1) = \mathbf{\Phi}_f(2) = \mathbf{\Phi}_f(3)$	$\begin{bmatrix} 0.4 & 0.4 & 0.4 & 0.4 & 0.4 & 0.4 & 0.4 & 0.4 \\ 0.8 & 0.8 & 0.8 & 0.8 & 0.8 & 0.8 & 0.8 & 0.8 \end{bmatrix}$	
P, M, C_1, C_2	10, 40, 1.5, 1.5	Population size, max. number of iterations, and acceleration constants of PSO



(a)

(b)

FIGURE 8. First difference of Lyapunov function (a) i =1, (b) i =2.



(a)

(b)

FIGURE 9. Stability conditions (a) (53) and (b) (54).

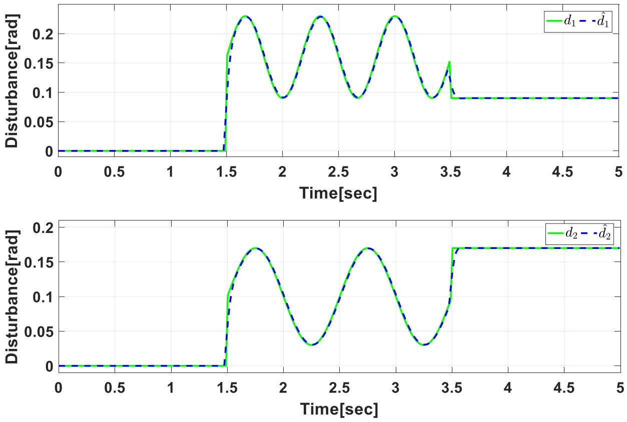
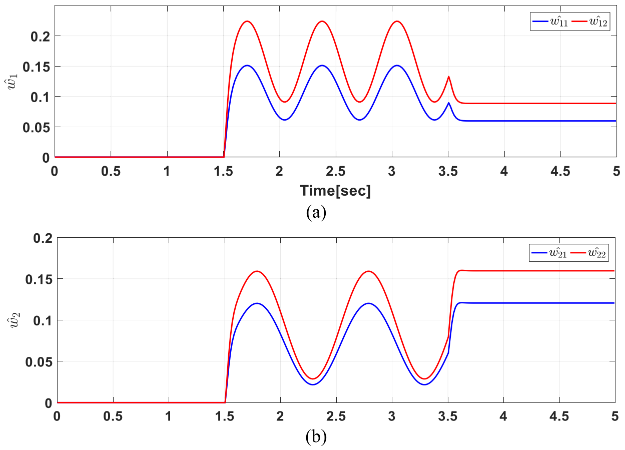


FIGURE 10. Disturbances and their estimations.

respectively. As shown in FIGUREs 10, the NNDOB can estimate disturbances with very precise accuracies. FIGURE 11 shows how the weights of the NNDOB change over time to cope with the disturbances incurred to the system.

TABLE 2. Mean absolute errors (MAE) and maximum errors.

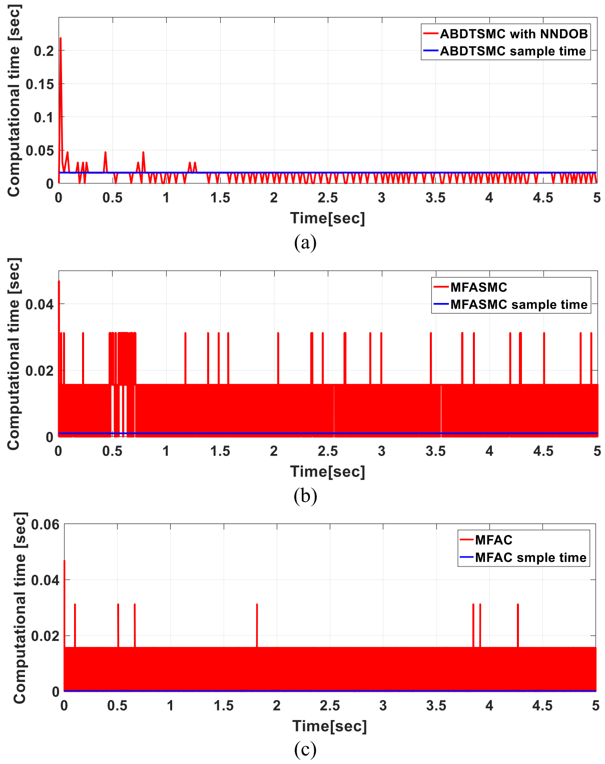
Controller	MAE for θ_1 [rad.]	Max. (e_1) [rad.]	MAE for θ_2 [rad.]	Max. (e_2) [rad.]
DDBATSMC	0.0778	0.0837	0.0777	0.2066
MFASMC	0.0937	0.1521	0.1363	0.2230
MFAC	0.1005	0.1852	0.1479	0.2732



(a)

(b)

FIGURE 11. NNDOB Adaptive weights (a) for $i = 1$, (b) for $i = 2$.



(a)

(b)

(c)

FIGURE 12. Computational time in each sample time (a) ABDTSMC, (b) MFASMC, (c) MFAC.

FIGURE 12 shows the computational time of the methods. It is obvious that the proposed method requires much less computational time than the other method. Hence, the computational expenses are drastically reduced. Moreover, it can be claimed that the proposed method can be implemented. In the next subsection, the experimental results will be presented.

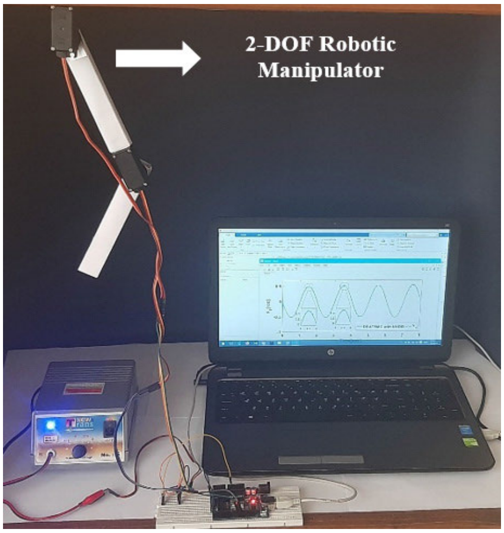
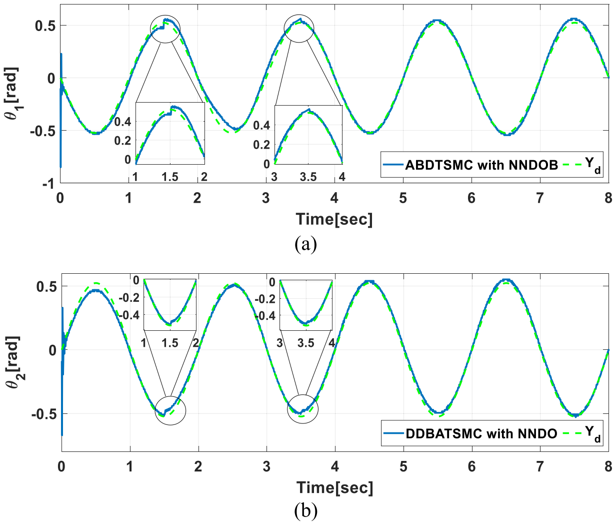


FIGURE 13. 2-DOF Robotic manipulator with controller.



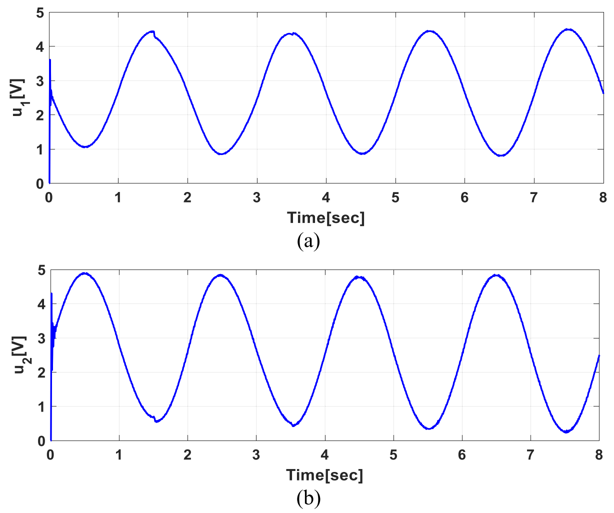
(a)

(b)

FIGURE 14. Joints position: (a) θ_1 and (b) θ_2.

B. EXPERIMENTAL RESULTS

In this part, the proposed controller is implemented to a 2-DOF manipulator moving in vertical plane. The 2-DOF



(a)

(b)

FIGURE 15. Applied voltages: (a) u_1 and (b) u_2.

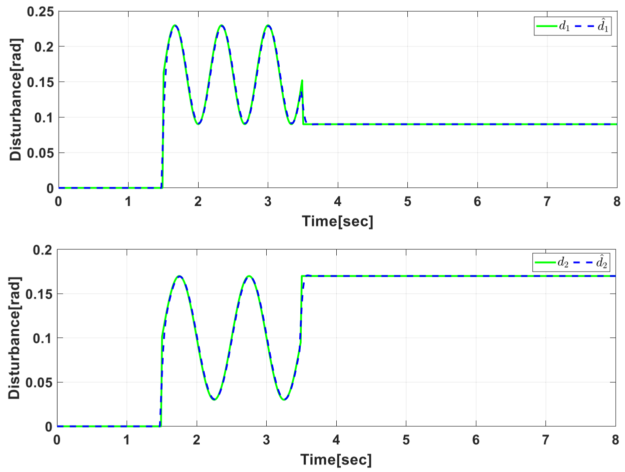


FIGURE 16. Disturbances and their estimations in experimental test.

manipulator contains two PWM servomotors that are actuated with a 6.5 volts power supply and derived with an Arduino UNO controller board. The sampling time for the proposed controller is considered as 4 msec. FIGURE 13 shows the experimental setup. It should be mentioned that due to the sampling time restriction on this experimental setup, the MFASMC and MFAC could not be implemented. The desired trajectory is considered as

$$\mathbf{y}_{\mathrm{d}}(k) = (\pi/6)\left[-\sin\left((1-k)\pi\right)\ \sin\left((1-k)\pi\right)\right]^T.$$

The experimental results are carried out for two cases: 1) time-varying external disturbances and 2) measurement noises.

1) RESULTS FOR TIME-VARYING EXTERNAL DISTURBANCES

FIGURES 14–17 depict the experimental results with time-varying external disturbances. As these figures show, very accurate tracking is achieved. Moreover, the chattering is drastically decreased. In addition, the sliding surface gradually converges to zero. Furthermore, the NNDOB estimates the external disturbances with high precisions.

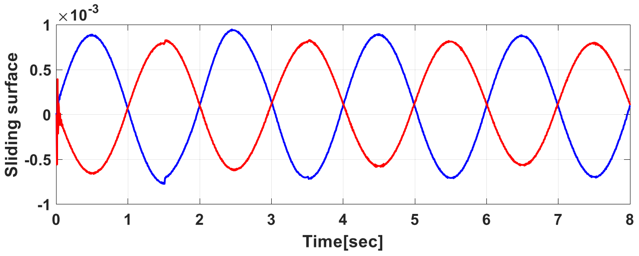
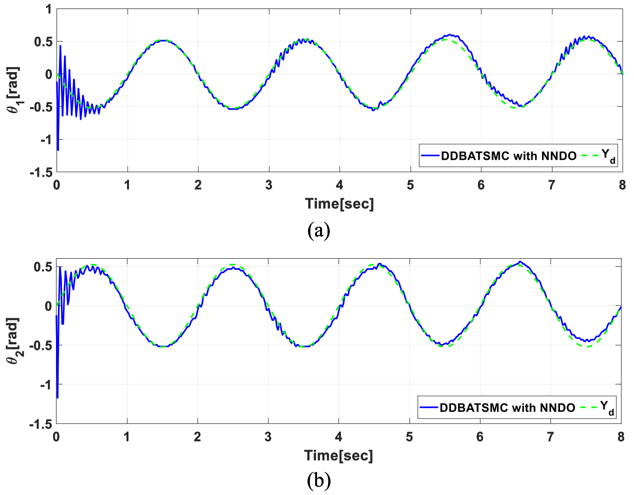


FIGURE 17. Sliding surface for the experimental test.



(a)

(b)

FIGURE 18. Joints position with measurement noise: (a) θ_1 and (b) θ_2.

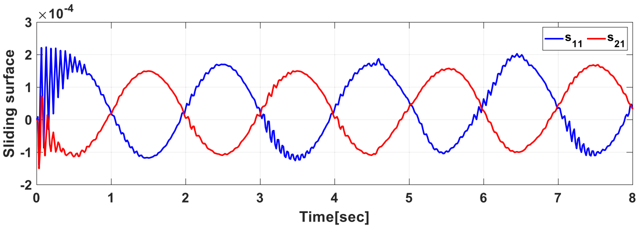
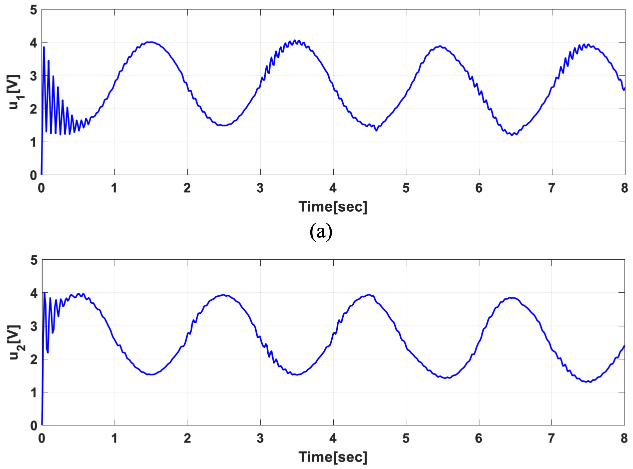


FIGURE 19. Sliding surface for the experimental test with measurement noise.

2) RESULTS FOR MEASUREMENT NOISES

One of the main problems of data-driven controllers is the controller performance in the presence of the measurement noise. Because the mathematical model is not used, the measurement noise can change stability and tracking accuracy of the closed-loop system. In this subsection, the experimental results will be depicted in the presence of the measurement noises (FIGURES 18–21). As shown in these figures, in the presence of measurement noise that is shown in FIGURE 21, the tracking accuracy is slightly decreased and the chattering



(a)

(b)

FIGURE 20. Applied voltages with measurement noise: (a) u_1 and (b)u_2.

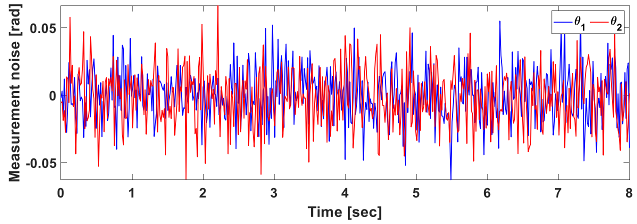


FIGURE 21. Measurement noises.

phenomenon is relatively increased. However, the overall controller performance is acceptable.

VI. CONCLUSION

In this paper, an adaptive back-stepping data-driven terminal sliding mode controller for non-affine MIMO systems is proposed. To overcome the complexity in the stability analysis and to decrease the conservatism in the stability condition, a new adaptation law was proposed for the PJM components. In addition, to achieve a control of system in the presence of time-varying disturbance and to compensate the missed data related to general disturbances, the neural network disturbance observer with adaptive weights was designed. The stability analysis based on the Lyapunov theory and back-stepping method was performed, and the appropriate regions for the stability of the closed-loop system were obtained. The PSO algorithm is used to calculate the suitable values for the controller parameters. Finally, the proposed method was simulated, implemented and applied to a 2-DOF robotic manipulator. The simulation and experimental results verified better performance of the proposed method with fewer data than the other data-driven methods. In comparison with the other data-driven methods, the effect of general disturbances was successfully suppressed, the chattering phenomenon was removed, the tracking ability was appropriately modified and the sampling time was significantly increased.

DECLARATION OF INTERESTS

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this article.

REFERENCES

[1] Y. Tian, Y. Cai, and Y. Deng, "A fast nonsingular terminal sliding mode control method for nonlinear systems with fixed-time stability guarantees," *IEEE Access*, vol. 8, pp. 60444–60454, 2020.

[2] X. Liu, H. Yu, J. Yu, and L. Zhao, "Combined speed and current terminal sliding mode control with nonlinear disturbance observer for PMSM drive," *IEEE Access*, vol. 6, pp. 29594–29601, 2018.

[3] H. Delavari and S. Naderian, "Backstepping fractional terminal sliding mode voltage control of an islanded microgrid," in *Proc. 5th Int. Conf. Control, Instrum., Autom. (ICCIA)*, Shiraz, Iran, Nov. 2017, pp. 167–172.

[4] S. Baek, J. Baek, and S. Han, "An adaptive sliding mode control with effective switching gain tuning near the sliding surface," *IEEE Access*, vol. 7, pp. 15563–15572, 2019.

[5] H. Delavari and S. Naderian, "Design and HIL implementation of a new robust fractional sliding mode control of microgrids," *IET Gener., Transmiss. Distrib.*, vol. 14, no. 26, pp. 6690–6702, Dec. 2020.

[6] X. Xu, L. Liu, and G. Feng, "Consensus of discrete-time linear multiagent systems with communication, input and output delays," *IEEE Trans. Autom. Control*, vol. 63, no. 2, pp. 492–497, Feb. 2018.

[7] M. Furat and G. G. Cücü, "Design, implementation, and optimization of sliding mode controller for automatic voltage regulator system," *IEEE Access*, vol. 10, pp. 55650–55674, 2022.

[8] F. Soltanian, M. Shasadeghi, S. Mobayen, and A. Fekih, "Adaptive optimal multi-surface back-stepping sliding mode control design for the Takagi–Sugeno fuzzy model of uncertain nonlinear system with external disturbance," *IEEE Access*, vol. 10, pp. 14680–14690, 2022.

[9] L. Zhang, H. Ding, J. Shi, Y. Huang, H. Chen, K. Guo, and Q. Li, "An adaptive backstepping sliding mode controller to improve vehicle maneuverability and stability via torque vectoring control," *IEEE Trans. Veh. Technol.*, vol. 69, no. 3, pp. 2598–2612, Mar. 2020.

[10] D. J. Almakhles, "Robust backstepping sliding mode control for a quadrotor trajectory tracking application," *IEEE Access*, vol. 8, pp. 5515–5525, 2020.

[11] Y. Wang and R.-J. Wai, "Design of discrete-time backstepping sliding—Mode control for LCL-type grid-connected inverter," *IEEE Access*, vol. 8, pp. 95082–95098, 2020.

[12] W. He, Y. Sun, Z. Yan, C. Yang, Z. Li, and O. Kaynak, "Disturbance observer-based neural network control of cooperative multiple manipulators with input saturation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 5, pp. 1735–1746, May 2020.

[13] Z. Zhao, Y. Ren, C. Mu, T. Zou, and K.-S. Hong, "Adaptive neural-network-based fault-tolerant control for a flexible string with composite disturbance observer and input constraints," *IEEE Trans. Cybern.*, vol. 52, no. 12, pp. 12843–12853, Dec. 2022.

[14] J. Peng, S. Ding, and R. Dubay, "Adaptive composite neural network disturbance observer-based dynamic surface control for electrically driven robotic manipulators," *Neural Comput. Appl.*, vol. 33, no. 11, pp. 6197–6211, Jun. 2021.

[15] X. Yang, W. Deng, and J. Yao, "Neural adaptive dynamic surface asymptotic tracking control of hydraulic manipulators with guaranteed transient performance," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Jan. 28, 2022, doi: 10.1109/TNNLS.2022.3141463.

[16] L. Cao, Z. Cheng, Y. Liu, and H. Li, "Event-based adaptive NN fixed-time cooperative formation for multiagent systems," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Oct. 10, 2022, doi: 10.1109/TNNLS.2022.3210269.

[17] L. Cao, Y. Pan, H. Liang, and T. Huang, "Observer-based dynamic event-triggered control for multiagent systems with time-varying delay," *IEEE Trans. Cybern.*, vol. 53, no. 5, pp. 3376–3387, May 2023.

[18] R. E. Precup, R. C. Roman, and A. Safaei, *Data-Driven Model-Free Controllers*. Boca Raton, FL, USA: CRC Press, 2022.

[19] Z. Hou and S. Jin, *Model Free Adaptive Control*. Boca Raton, FL, USA: CRC Press, 2013.

[20] Z. Hou, H. Gao, and F. Lewis, "Data-driven control and learning systems," *IEEE Trans. Ind. Electron.*, vol. 64, no. 5, pp. 4070–4075, May 2017.

[21] S. Formentin, K. van Heusden, and A. Karimi, ''A comparison of model-based and data-driven controller tuning,'' *Int. J. Adapt. Control Signal Process.*, vol. 28, no. 10, pp. 882–897, Oct. 2014.

[22] Z.-S. Hou and Z. Wang, ''From model-based control to data-driven control: Survey, classification and perspective,'' *Inf. Sci.*, vol. 235, pp. 3–35, Jun. 2013.

[23] S. Xiong and Z. Hou, ''Model-free adaptive control for unknown MIMO nonaffine nonlinear discrete-time systems with experimental validation,'' *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 4, pp. 1727–1739, Apr. 2022.

[24] Y. Weng and X. Gao, ''Adaptive sliding mode decoupling control with data-driven sliding surface for unknown MIMO nonlinear discrete systems,'' *Circuits, Syst., Signal Process.*, vol. 36, no. 3, pp. 969–997, Mar. 2017.

[25] Y. P. Weng and X. W. Gao, ''Data-driven sliding mode control of unknown MIMO nonlinear discrete-time systems with moving PID sliding surface,'' *J. Franklin Inst.*, vol. 354, no. 15, pp. 6463–6502, Oct. 2017.

[26] X. Wang, X. Li, J. Wang, X. Fang, and X. Zhu, ''Data-driven model-free adaptive sliding mode control for the multi degree-of-freedom robotic exoskeleton,'' *Inf. Sci.*, vol. 327, pp. 246–257, Jan. 2016.

[27] N. Ebrahimi, S. Ozgoli, and A. Ramezani, ''Data-driven sliding mode control: A new approach based on optimization,'' *Int. J. Control*, vol. 93, no. 8, pp. 1980–1988, Aug. 2020.

[28] D. Xu, X. Song, B. Jiang, W. Yang, and W. Yan, ''Data-driven sliding mode control for MIMO systems and its application on linear induction motors,'' *Int. J. Control, Autom. Syst.*, vol. 17, no. 7, pp. 1717–1725, Jul. 2019.

[29] H. Gao, G. Ma, Y. Lv, and Y. Guo, ''Forecasting-based data-driven model-free adaptive sliding mode attitude control of combined spacecraft,'' *Aerosp. Sci. Technol.*, vol. 86, pp. 364–374, Mar. 2019.

[30] D. Xu, Y. Shi, and Z. Ji, ''Model-free adaptive discrete-time integral sliding-mode-constrained-control for autonomous 4 WMV parking systems,'' *IEEE Trans. Ind. Electron.*, vol. 65, no. 1, pp. 834–843, Jan. 2018.

[31] K. Zhao, T. Yin, C. Zhang, J. He, X. Li, Y. Chen, R. Zhou, and A. Leng, ''Robust model-free nonsingular terminal sliding mode control for PMSM demagnetization fault,'' *IEEE Access*, vol. 7, pp. 15737–15748, 2019.

[32] R.-E. Precup, M.-B. Radac, R.-C. Roman, and E. M. Petriu, ''Model-free sliding mode control of nonlinear systems: Algorithms and experiments,'' *Inf. Sci.*, vol. 381, pp. 176–192, Mar. 2017.

[33] M. Liu, Z. Zhao, and L. Hao, ''Data-driven sliding mode control of shape memory alloy actuators with prescribed performance,'' *Smart Mater. Struct.*, vol. 30, no. 6, Jun. 2021, Art. no. 065012.

[34] M. Hou and Y. Wang, ''Data-driven adaptive terminal sliding mode control with prescribed performance,'' *Asian J. Control*, vol. 23, no. 2, pp. 774–785, Mar. 2021.

[35] W. Zhang, D. Xu, B. Jiang, and T. Pan, ''Prescribed performance based model-free adaptive sliding mode constrained control for a class of nonlinear systems,'' *Inf. Sci.*, vol. 544, pp. 97–116, Jan. 2021.

[36] D. Liu and G.-H. Yang, ''Performance-based data-driven model-free adaptive sliding mode control for a class of discrete-time nonlinear processes,'' *J. Process Control*, vol. 68, pp. 186–194, Aug. 2018.

[37] D. Liu and G.-H. Yang, ''Prescribed performance model-free adaptive integral sliding mode control for discrete-time nonlinear systems,'' *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 7, pp. 2222–2230, Jul. 2019.

[38] D. Liu and G.-H. Yang, ''Data-driven adaptive sliding mode control of nonlinear discrete-time systems with prescribed performance,'' *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 49, no. 12, pp. 2598–2604, Dec. 2019.

[39] L. Chen, H. Duan, S. Dian, and S. Hoang, ''Model free sliding mode control for inspection robot based on pigeon-inspired optimization,'' in *Proc. IEEE 15th Int. Conf. Control Autom. (ICCA)*, Edinburgh, U.K., Jul. 2019, pp. 537–542.

[40] B. Esmaeili, M. Salim, M. Baradarannia, and A. Farzamnia, ''Data-driven observer-based model-free adaptive discrete-time terminal sliding mode control of rigid robot manipulators,'' in *Proc. 7th Int. Conf. Robot. Mechatronics (ICRoM)*, Tehran, Iran, Nov. 2019, pp. 432–438.

[41] Y. Weng, N. Wang, and C. G. Soares, ''Data-driven sideslip observer-based adaptive sliding-mode path-following control of underactuated marine vessels,'' *Ocean Eng.*, vol. 197, Feb. 2020, Art. no. 106910.

[42] X. Huang, D. Zhai, and J. Dong, ''Adaptive integral sliding-mode control strategy of data-driven cyber-physical systems against a class of actuator attacks,'' *IET Control Theory Appl.*, vol. 12, no. 10, pp. 1440–1447, Jul. 2018.

[43] J. Park and I. W. Sandberg, ''Universal approximation using radial-basis-function networks,'' *Neural Comput.*, vol. 3, no. 2, pp. 246–257, Jun. 1991.

[44] J. J. Craig, *Introduction to Robotics: Mechanics and Control*. Pearson, NJ, USA: Pearson Education, 2005.

SINA NADERIAN was born in Hamedan, Iran, in 1992. He received the B.S. degree in electrical power engineering from Buali Sina University, Hamedan, in 2015, and the M.S. degree in control engineering from the Hamedan University of Technology, in 2017. He is currently pursuing the Ph.D. degree in control engineering with the Iran University of Science and Technology. From 2016 to 2019, he was involved in the control of microgrids and published some papers related to this title. In 2021, he was the Head of the Robotics Group, Hamedan, to implement a disinfection robot. His research interests include designing and implementing data-driven, reinforcement learning, sliding-mode, neural networks, and fractional controllers on power plants and robotic systems.

MOHAMMAD FARROKHI received the B.S. degree in electrical engineering from the K. N. Toosi University of Technology, Tehran, Iran, in 1985, and the M.S. and Ph.D. degrees in electrical engineering from Syracuse University, Syracuse, NY, USA, in 1989 and 1996, respectively. In 1996, he joined the Iran University of Science and Technology, where he is currently a Professor of electrical engineering. His research interests include automatic control, fuzzy systems, and neural networks.