Data-Driven Model Free Adaptive Sliding Mode Control for Multi DC-Motors Speed Regulation

Tony Blaise Bimenyimana

Abstract

This paper proposes the distributed data-driven sliding mode control approach to address the consensus problem of nonlinear multi-agent systems. Firstly, the equivalent data model for each agent is constructed using the compact-form dynamic linearization (CFDL) technique. Secondly, . Finally, the effectiveness of the proposed control approach is verified through experiments on multi-DC motors.

KEYWORDS

Model-free adaptive sliding mode control, compact form dynamic linearization, nonlinear multi-agent system, data-driven control.

I. INTRODUCTION

Motivated by the coordinated behaviors observed in nature, such as bird flocking or herd migration, the distributed control multi-agent systems(MASs) has been studied with interest in recent years. [1] Because of the distributed and flexible structure, the MASs based approaches have significant potential in addressing coordination challenges for complex systems [2]. The control of DC motors is essential in numerious technological applications, with single DC motor serving as the foundation for the systems in various fields, including vehicle coorporation [3], multi-robot systems and industrial machinery [4].

It is challenging to establish an accurate mathematical model for the speed regulation of multi-DC motor systems due to their nonlinear, time-varying, and multi-variable conditions [7]. Even if a relatively accurate mathematical models are developed, the associated control algorithms can become highly complex, so the model-based control approach is difficult to apply or extend to such systems. Nowdays, data-driven methods have been used to control, decision making, planing, fault diagnosis, predicting, etc. In [15]- [16], a compact form dynamic linarization (CFDL) data-driven modeling method is proposed for nonlinear systems. So far, the CFDL data-driven modeling method has proven to be highly useful in various domains [17]-[20] with different characteristics such as simplicity and particularly, a small amount of calculation, ease of implementation, and strong robustness, making it highly effective for addressing unknown nonlinear time-varying systems.

In [21], a model-free adaptive control (MFAC) approach is presented for MASs, which uses input and output data to achieve consensus tracking trajectory.

Alternatively, the sliding mode control (SMC) adapts dynamically the system operation based on the current state of the system such as the error and its derivative, ensuring the system follows a predetermined sliding surface [22]. Because the sliding surface can be designed and has the purpose to deal with the object parameters and disturbances, the SMC provides the fast response, insensitivity to parameter changes and disturbances, and simplicity in implementation. The above advantages make SMC widely used in control systems and one of main topic to focus in the ongoing research. [28]- [30]

Inspired by the abovementioned considerations, this paper studies a novel model-free adaptive sliding mode control for speed regulation of multi-DC motors. The proposed control method consists to eliminates the need for precise system models by

dynamically adjusting the control law [21], where is particularly useful in systems where obtaining a detailed model is impractical or complex such as the tradition PID control method. Compared with other existing literature on multi-DC motors systems, the results of this paper have following distinct features.

1) The proposed method integrates encoder counts based on the well known constant elapsed time (CET) method [31] for accurate speed control in nonlinear, time-varying multi-DC motor systems. This innovation greatly enhances resolution and reliability, setting it apart from traditional methods.
2) The implementation of a fixed communication topology for agents interconnections, which makes the controller more applicable and flexible in real implementation.
3) Utilizes the pseudo partial derivative (PPD) estimation mechanism within the MFASMC scheme, which provides strong robustness

The following sections will outline the remaining content of this paper: section II provides preliminaries and problem formulation, Section III The main results, Section IV presents simulation results and performance analysis, demonstrating the efficacy of the proposed method under various operating conditions. At the end, Section V concludes the paper, summarizing the key findings potential points for future research.

II. Preliminaries and problem formulation

A. Directed Graph Theory

The directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$ is employed to describe the information exchange between agents. Here, $\mathcal{A} = [a_{ij}] \in \mathbb{R}^{N\times N}$ represents the adjacency matrix, $\mathcal{V} = \{v_1, v_2, \ldots, v_N\}$ is the set of vertices, and $\mathcal{E} = [(v_j, v_i)|v_i \in \mathcal{V}] \subseteq \mathcal{V} \times \mathcal{V}$ is the set of edges. Moreover, $\mathcal{N}(i) = \{j \in \mathcal{V}|(i,j) \in \mathcal{E}\}$ denotes the neighbor set of agent i, where $a_{ij} \neq 0$. No self-loop is allowed in this article, which means $(i,i) \notin \mathcal{E}$ for any $i \in \mathcal{V}$, $a_{ii} = 0$. Furthermore, the degree matrix $K = diag(k_1, \ldots, k_N)$. If $k_i > 0$, agent i can directly obtain the information from the leader. The Laplacian matrix L is defined as $L = (\mathcal{D} - \mathcal{A})$, here $\mathcal{D} = diag(d_1, \ldots, d_N)$ and $d_i = \sum_{j=1}^{N} a_{ij}$ denotes the in-degree matrix. Moreover, the graph is strongly connected if the path exists between every pair of vertices.

B. Problem Formulation

Consider the nonlinear multi-agent systems composed of N agents:

$$y_i(k+1) = f_i(y_i(k), u_i(k)), \quad i = 1, 2, \ldots, N \tag{1}$$

where $u_i(k) \in \mathbb{R}$ and $y_i(k) \in \mathbb{R}$ represent the system input and output signals of agent i, respectively. $f_i(\cdot)$ signifies an unknown nonlinear function.

Assumption 1: The partial derivative of $f_i(\cdot)$ with respect to $u_i(k)$ is continuous.

Assumption 2: The system (1) satisfies the generalized Lipschitz condition, meaning that if $\Delta u_i(k) = u_i(k) - u_i(k-1) \neq 0$ then $|\Delta y_i(k+1)| \leq b|\Delta u_i(k)|$ holds for any k, where $\Delta y_i(k+1) = y_i(k+1) - y_i(k)$.

Remark 1: The assumptions above are general. As the common condition for model-free control methods, Assumption 1 is crucial in ensuring effective system performance. Assumption 2 implies that the system input rate constrains the system output rate, which is satisfied in many practical systems.

Assumption 3: The network of followers forms a strongly connected graph, ensuring each follower can directly receive information from at least one leader.

Lemma 1 [8]: Consider the nonlinear multi-agent system (1) satisfying above two assumptions. If $|\Delta u_i(k)| \neq 0$ holds, then the system can be transformed into the CFDL data model as follows:

$$\Delta y_i(k+1) = \phi_i(k)\Delta u_i(k) \tag{2}$$

where $\phi_i(k)$ is called pseudo partial derivative (PPD), satisfying $|\phi_i(k)| \leq b$.

The distributed measurement error of $\xi_i(k)$ for N agents is established as:

$$\xi_i(k) = \sum_{j \in N_i} a_{ij}(y_j(k) - y_i(k)) + d_i(y_d(k) - y_i(k)) \tag{3}$$

if the agent i can receive data from the leader, then $d_i = 1$; otherwise, $d_i = 0$. Additionally, $y_d(k)$ represents the reference trajectory.

Remark 2: The CFDL technique requires no prior knowledge about the system dynamic model. Moreover, the dynamic behavior of time-varying PPD can be highly complex and more challenging to verify. Therefore a data-driven method for studying numerical characteristics is the preferred solution.

III. Main Results

A. Model Free Adaptive Controller Design

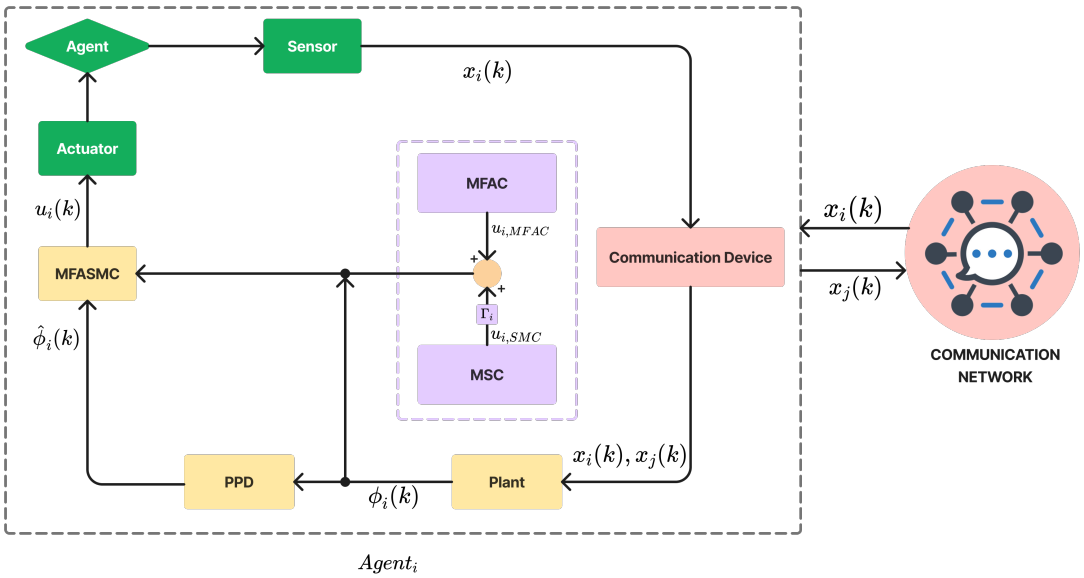

Fig. 1: Block diagram.

Consider the following PPD criterion function, which is used to evaluate the performance of the parameter $\phi_i(k)$:

$$J(\phi_i(k)) = |\Delta y_i(k) - \phi_i(k)\Delta u_i(k-1)|^2 + \mu|\phi_i(k) - \hat{\phi}_i(k-1)| \tag{4}$$

Assumption 4 : The PPD $\phi_i(k) > \varsigma$, $i = 1, 2, 3, \ldots, N$ holds for all k, where ς is an rondomly small positive constant without loss of generality, assume that $\phi_i(k) > \varsigma$.

Differentiating equation (4) with respect to PPD parameter $\phi_i(k)$ and make it equal to zero, the following update rule is proposed for the distributed MFAC algorithm:

$$\hat{\phi}_i(k) = \hat{\phi}_i(k-1) + \frac{\eta \Delta u_i(k-1)(\Delta y_i(k) - \hat{\phi}_i(k-1)\Delta u_i(k-1))}{\mu + \Delta u_i(k-1)} \tag{5}$$

$$\hat{\phi}_i(k) = \hat{\phi}_i(1), \text{ if} |\hat{\phi}_i(k)| \leq \epsilon \ or \ sign(\hat{\phi}_i(k)) \neq sign(\hat{\phi}_i(1)) \tag{6}$$

Here, η is the learning rate that controls the step size of the update, $\mu > 0$ is a weight factor. $\hat{\phi}_i(1)$ is the initial value of $\hat{\phi}_i(k)$ and $\hat{\phi}_i(k)$ is the estimated value of $\phi_i(k)$.

Remark 3: In the parameter estimation law (5), $y_i(k)$ is used to estimate $\hat{\phi}_i(k)$. The abovementioned scheme ensures the convergence of (5). Additionally, the reset algorithm (6) is introduced to enhance the ability of the parameter estimation algorithm to effectively track time-varying parameters.

To design the MFAC alogrithm, the performance function $J(u_i(k))$ is set as:

$$J(u_i(k)) = |\xi_i(k+1)|^2 + \lambda |u_i(k) - u_i(k-1)|^2 \tag{7}$$

Subtituting (2) and (3) into (7), then differentiating (7) with respect to $u_i(k)$, and make it equal to zero, gives:

$$u_{i,\text{MFAC}}(k) = u_{i,\text{MFAC}}(k-1) + \frac{\rho \phi_i(k)}{\lambda + |\phi_i(k)|^2} \xi_i(k) \tag{8}$$

where $\rho \in$ (0,1) is a step-size constant, which is added to make (8) general. Using the parameter estimation algorithm (5) and the control law algorithm (8), the MFAC scheme is constructed.

B. Sliding Mode Controller Design

To design the SMC for system, the sliding mode surface is first defined as:

$$S_i(k+1) = S_i(k) + e_i(k+1) + \alpha e_i(k) \tag{9}$$

Taking 3 into 9, with reaching law $S_i(k+1) = 0$, the equivalent control law is then designed as follow:

where α is a positive constant, and to ensure that the system trajectory is driven toward and remains on the sliding surface. The reaching law dictates how quickly the system state converges to the sliding surface and is given by:

$$\Delta S_i(k+1) = -\varepsilon T sign(k) \tag{10}$$

In that equation, ε is a small positive constant that controls the rate of the convergence, T is the sampling period, and $sign(k)$ indicates the direction in which the system should move to reach the sliding surface.

By combining the sliding surface definition and the reaching law, derivating the control law that ensures the desired tracking performance while maintaining robustness.

The final sliding mode control input $u_{i,\text{SMC}}(k)$ is designed:

$$u_{i,\text{SMC}}(k) = u_{i,\text{SMC}}(k) + \frac{y_d(k+1) - y(k) + \alpha e_i(k) + \varepsilon T sign(k)}{\phi_i(k)} \tag{11}$$

To enhance the adaptibility of the control system, the MFASMC approach is employed. the conrol input of the system is defined as:

$$u_i(k) = u_{i,\text{MFAC}}(k) + \Gamma_i u_{i,\text{SMC}}(k) \tag{12}$$

where the parameter Γ is a gain factor that adjusts the contribution of the sliding mode control in the control effort and tunes the convergence rate.

C. Stability Analysis

Theorem 1: For the MASs (1) that satifies Assumptions 1 and 2, assuming $|\xi_{i,0}| < \min\{a_{i,0}, b_{i,0}\}\rho_{i,0}$

The stability analysis is conducted in two primary steps. The first step focuses on estabilishing the bounds, the second step ensures that the error remains within acceptable limits over time, leading to a stable system.

Step 1: Establishment of error bounds

The estimated error between the estimated and actual values of the system parameters, denote as $\tilde{\phi}_i(k) = \hat{\phi}_i(k) - \phi_i(k)$, starting from the foundational equation derived from the compact dynamic linearization model in equation (2) along with the PPD estimation equation (5). To ensure the theorem validity, the detailed proof that demonstrates the correctness of this error bound is provided.

$$\tilde{\phi}_i(k) = \hat{\phi}_i(k+1) + \frac{\eta \Delta u_i(k-1)}{\mu + |\Delta u_i(k-1)|^2}((\Delta y_i(k) - \hat{\phi}_i(k-1)\Delta u_i(k-1))) - \phi_i(k) \tag{13}$$

Then by simplifying the previous equation, the result is:

The control coefficient $\beta_i(k)$ is expressed from the previous equation, plays a critical role in adjusting the control input for each agent at time step k. The following equation is expressed:

$$\tilde{\phi}_i(k) = \tilde{\phi}_i(k-1) + \beta_i(k)(\Delta y_i(k) - \hat{\phi}_i(k-1)\Delta u_i(k-1) - \phi_i(k) - \phi_i(k-1)) \tag{14}$$

$$\tilde{\phi}_i(k) = (1 - \frac{\eta(\Delta u_i(k-1))^2}{\mu + |\Delta u_i(k-1)|^2})\tilde{\phi}_i(k-1) - \Delta\phi_i(k) \tag{15}$$

To demonstrate the boundedness of the error, by taking the absolute value of both sides of the error (15). This is a crucial step, as it allows us to establish an inequality that provides an upper bound on the error term.

Taking the absolute value on both sides and applying the triangle inequality to the right-hand side, it follows that:

$$|\tilde{\phi}_i(k)| \leq \left|1 - \frac{\eta(\Delta u_i(k-1))^2}{\mu + |\Delta u_i(k-1)|^2}\right| |\tilde{\phi}_i(k-1)| + |\Delta\phi_i(k)| \tag{16}$$

Defining:

$$\alpha(k-1) = \frac{\eta(\Delta u_i(k-1))^2}{\mu + |\Delta u_i(k-1)|^2} \tag{17}$$

So (16) becomes:

$$|\tilde{\phi}_i(k)| \leq |1 - \alpha(k-1)||\tilde{\phi}_i(k-1)| + |\Delta\phi_i(k)| \tag{18}$$

Since $|\phi_i(k)| \leq b$, considering assumption 4, we can obtain $|\phi_i(k-1) - \phi_i(k)|$, and then:

$$|\tilde{\phi}_i(k)| \leq |1-q_1||\tilde{\phi}_i(k-1)| + b \tag{19}$$

Back to the initial condition at $k=0$ and summing the resulting geometric series:

$$\sum_{j=0}^{k-1}(1-q_1)^j = \frac{1-(1-q_1)^k}{q_1}$$

Thus:

$$|\tilde{\phi}_i(k)| \leq (1-q_1)^k|\tilde{\phi}_i(0)| + \frac{b}{q_1}(1-(1-q_1)^k) \tag{20}$$

As $k \to \infty$, the term $(1-q_1)^k$ tends to zero, simplifying the bound to:

$$|\tilde{\phi}_i(k)| \leq \frac{b}{q_1} \tag{21}$$

Remark 4: The inequality in (21) shows that $\tilde{\phi}_i(k)$ is bounded by $\frac{b}{q_1}$. This result implies that the error $\tilde{\phi}_i(k)$ will remain within this bound, even as k approaches infinity. Thus, the system error behavior is effectively controlled and constrained.

Step 2:

The expression for $\xi_i(k)$ is formulated as follows:

$$\xi_i(k) = \sum_{j\in N_i}(e_i(k) - e_j(k)) + d_i e_i(k) \tag{22}$$

In this equation, $\xi_i(k)$ represents the distributed measurement of the ith system at time k, taking into account the deviations from the neighbors j in the set N_i and an additional term $d_i e_i(k)$ that depends on the specific characteristics of the ith system. Define the collective stack vectors as follows:

$$\begin{aligned}
y(k) &= [y_1(k) \quad y_2(k) \quad \dots \quad y_n(k)]^T \\
e(k) &= [e_1(k) \quad e_2(k) \quad \dots \quad e_n(k)]^T \\
\xi(k) &= [\xi_1(k) \quad \xi_2(k) \quad \dots \quad \xi_n(k)]^T \\
u(k) &= [u_1(k) \quad u_2(k) \quad \dots \quad u_n(k)]^T
\end{aligned}$$

Using the above-mentioned definitions, the measurement output $\xi_i(k)$ can be rewritten as as follows:

$$\xi(k) = (L+D)e(k) \tag{23}$$

Where L represents the interaction matrix that describes how each system interacts with its neighbors, while D is a diagonal matrix defined by $D = \text{diag}(d_1, d_2, d_3, \dots, d_n)$. By noting the definition in (23), the controller (8) is rewritten as:

$$u(k) = u(k-1) + \rho H_1(k)(L+D)e(k) \tag{24}$$

where

$$H_1(k) = \rho\text{diag}\left(\frac{\hat{\phi}_1(k)}{\lambda + |\hat{\phi}_1(k)|^2}, \frac{\hat{\phi}_2(k)}{\lambda + |\hat{\phi}_2(k)|^2}, \dots, \frac{\hat{\phi}_n(k)}{\lambda + |\hat{\phi}_n(k)|^2}\right).$$

In similar way, CFDL model (2) is also transformed into the following collective form:

$$y(k+1) = y(k) + H_\phi(k)\Delta u(k) \tag{25}$$

where

$$\Delta u(k) = u(k) - u(k-1)$$

$$H_\phi(k) = \text{diag}(\phi_1(k), \phi_2(k), \phi_3(k), \ldots, \phi_n(k))$$

We can substitute (24) and (25) to get

$$e(k+1) = (I - \rho \sum(k)(L+D))e(k) \tag{26}$$

where $\sum(k) = H_\phi(k)H_1(k) = \text{diag}(\mathcal{V}_1(k), \mathcal{V}_2(k), \mathcal{V}_3(k), \ldots, \mathcal{V}_n(k))$, and each $\mathcal{V}_i(k)$ is given by:

$$\mathcal{V}_i(k) = \frac{\hat{\phi}_i(k)}{\lambda + |\hat{\phi}_i(k)|^2}, \quad i = 1, 2, \ldots, n$$

Thus :

$$\Theta(k) = \text{diag}(\mathcal{V}_1(k), \mathcal{V}_2(k), \mathcal{V}_3(k), \ldots, \mathcal{V}_n(k))(L+D)$$

To ensure convergence of the tracking error, the following condition is imposed:

$$\|I - \rho\Theta(k)\| < 1 \tag{27}$$

This condition guarantees that the tracking error $e(k)$ will approach zero as $k \to \infty$. Consequently:

$$\lim_{k\to\infty} \|e(k+1)\| = 0$$

Remark 5: This condition ensures that the tracking error decreases over time and eventually converges to zero as k approaches infinity. The implication is under the proposed control strategy, the system will successfully align with the desireed performance, eliminating any discrepancies in tracking. This convergence demonstrates the robustness and effectiveness of the control method in achieving accurate and stable performance.

IV. Simulation Example

Consider the network comprising:

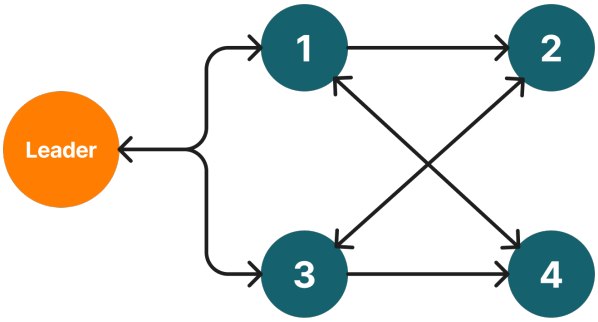

Fig. 2: Communication topology among agents.

This section describes the methodology and hardware used for speed regulation, including the calculation of motor speed using the quadruple-frequency data processing method.

The DC brushed motors have a rated voltage of 12V, an unloaded speed of 293 ± 21 RPM and a rated current of 0.36 A. The gear ratio of 20 means that the output speed of the motor is 1/20 of the rotor speed, resulting in higher torque with a higher gear ratio. The Hall encoders used have 13 pulses per revolution, meaning each full rotation generates 13 pulse signals.

To enhance measurement accuracy, employing the quadruple-frequency data processing method. This technique quadruples the effective resolution of the encoder by processing the output pulse signals at four times the frequency, thus increasing measurement precision by a factor of four.

The motor speed is measured in revolution per second (r/s) based on encoder measurements and the sampling interval t. The total number of encoder counts per revolution is calculated as $T = 4N_eRr$, where N_e is the encoder line count equal to 13, R_r is the reduction ratio equal to 20 and the factor of 4 accounts for quadrature encoding. The number of rotations, N_r is determined using $N_r = \frac{m}{T}$, with m representing the total encoder count.

The speed of the motor is then derived as:

$$v = \frac{N_r}{t} = \frac{m}{Tt} \tag{28}$$

The quadruple-frequency method is crucial for maximizing encoder measurement precision, resulting in more accurate speed control for the motor system.

Four follower DC motors and the models for each DC motor are governed by:

$$\begin{aligned}
\text{DC Motor 1}: &\quad y_1(k+1) = \tfrac{mu_1(k)}{0.1T} \\
\text{DC Motor 2}: &\quad y_2(k+1) = \tfrac{mu_2(k)}{0.1T} \\
\text{DC Motor 3}: &\quad y_3(k+1) = \tfrac{mu_3(k)}{0.3T} \\
\text{DC Motor 4}: &\quad y_4(k+1) = \tfrac{mu_4(k)}{0.3T}
\end{aligned}$$

It is evident that the agents considered are heterogeneous, as the dynamics differ from one another. In this scenario, the dynamics are assumed to be unknown and are only used to generate the I/O data for the MASs.

As illustrated in Fig. 2, the virtual leader is designated as vertex 0. It can be observed that only agents 1 and 3 can receive information from the leader, forming a strongly connected communication graph. Assume that the information exchange among agents is directed and fixed. The Laplacian matrix of the graph is given as follows:

$$L = \begin{bmatrix} 1 & 0 & 0 & -1 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 1 & 0 \\ -1 & 0 & -1 & 2 \end{bmatrix}$$

with $D = \text{diag}(1, 0, 1, 0)$, we consider the following two different desired trajectories.

The expression for $y_d(k)$ is:

$$y_d(k) = 0.5 \sin\left(\frac{k\pi}{30}\right) + 0.3 \cos\left(\frac{k\pi}{10}\right)$$

as k in the range $0 \leq k \leq 200$.

The initial parameters are chosen as $u_i(1) = 0.1$, $y_i(1) = 0.1$ and $\phi_i(0) = 1$ for all agents in this simulation, $\Gamma_1 = \Gamma_2 = 0.15$ and $\Gamma_3 = \Gamma_4 = 0.45$, with $T = 0.1$, $m = 350$, $\eta = 1$, $\mu = 1$, other parameters are given as $\rho = 1$, $\lambda = 50$ and $\alpha = 1$ with $\epsilon = 10^{-5}$.

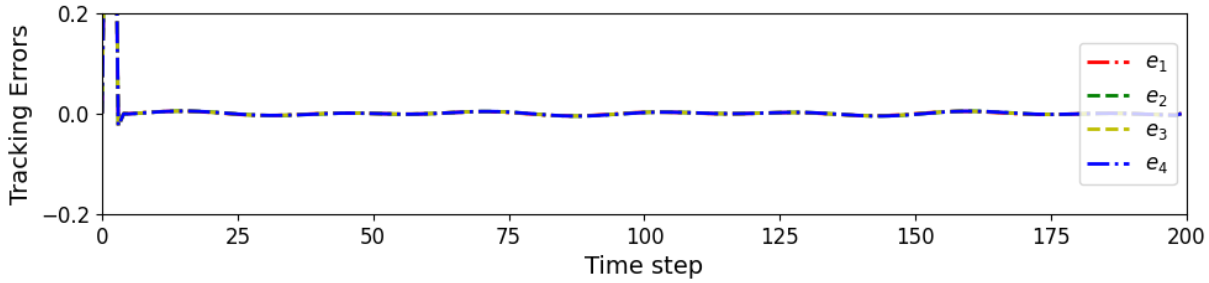

Fig. 3: Tracking errors for time varying desired trajectory.

As shown in Fig. 3, the tracking errors between the actual and desired trajectories for agents e_1, e_2, e_3, and e_4 are relatively small and converge to zero over time. However, the individual agents exhibit varying levels of tracking error, suggesting that their unique dynamics or initial conditions may influence the performance.

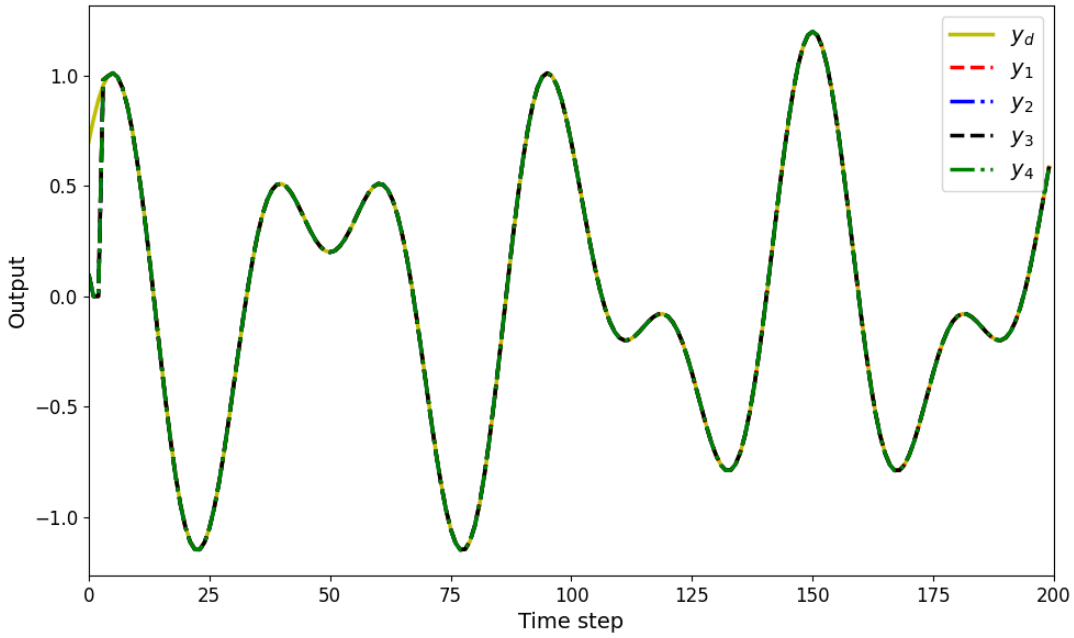

Fig. 4: Tracking performance of all agents for time-varying desired trajectory.

Fig. 4 presents a detailed analysis of the tracking performance for all agents. All agents successfully track the time-varying desired trajectory, confirming the effectiveness of the proposed control system. While minor variations in individual trajectories are evident, each agent generally adheres to the desired path. Factors such as agent dynamics, communication delays, and environmental disturbances could potentially influence the tracking performance.

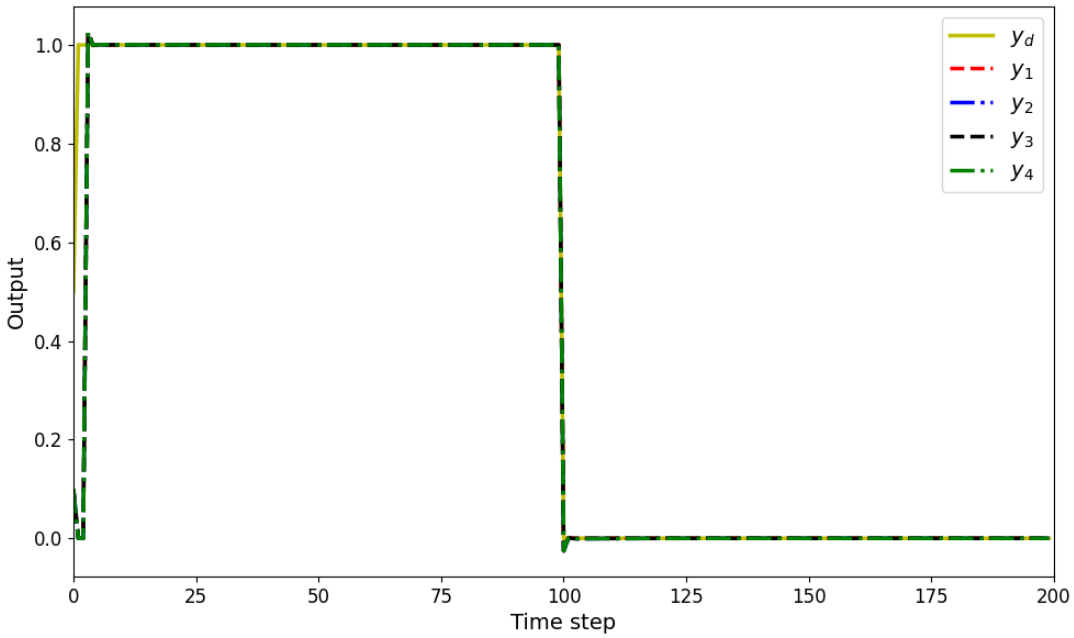

Fig. 5: Tracking performance of all agents for time-invariable desired trajectory.

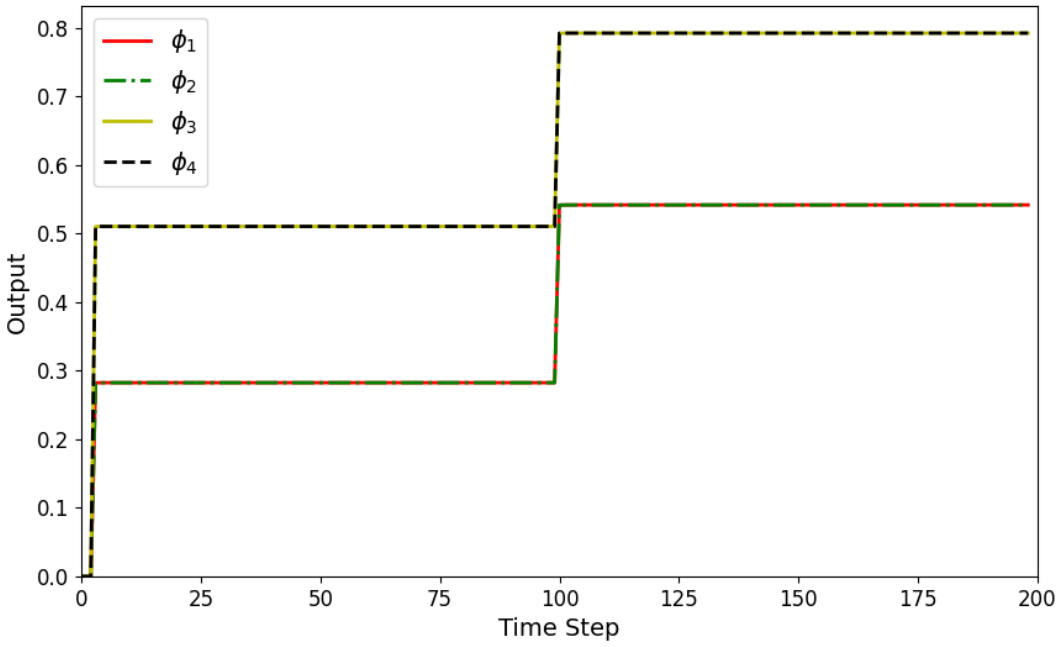

Fig. 6: PPD Estimation of all agents.

Fig. 5 demonstrates that all agents successfully track the time-invariable desired trajectory. Meanwhile, Fig. 6 shows the PPD estimation for all agents, highlighting the accuracy of the adaptive estimation process within the control framework.

To verify the proposed consensus tracking control methodology, the experimental validation is conducted using a multi-DC motor system, as illustrated in Fig. 8. The system consists of three DC motors equipped with Hall encoders and reduction gears, an STM32F407 main control chip, two motor drive modules, and an LCD display module. The microcontroller unit (MCU) STM32F407ZGT6 is used for high-resolution pulse width modulation (PWM) output generation to achieve precise motor speed control. The timer module is utilized for this purpose.

In addition, the controller code is written in C language using STM32CubeIDE, while STM32CubeMX is used for pin configuration. The main purpose of the experiment is to ensure that the three motors accurately track the reference trajectory:

$$y_d(k) = 0.5\sin(0.07\pi(k)) + 0.7\cos(0.04\pi(k))$$

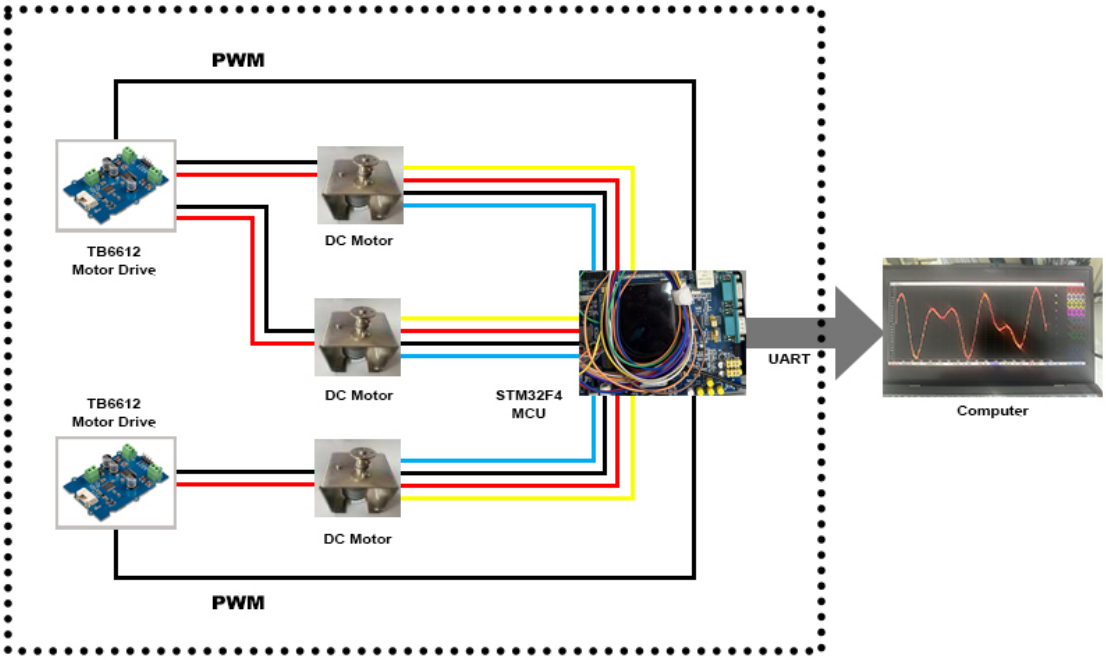

Fig. 7: System connection diagram of multi-DC motors consensus tracking control.

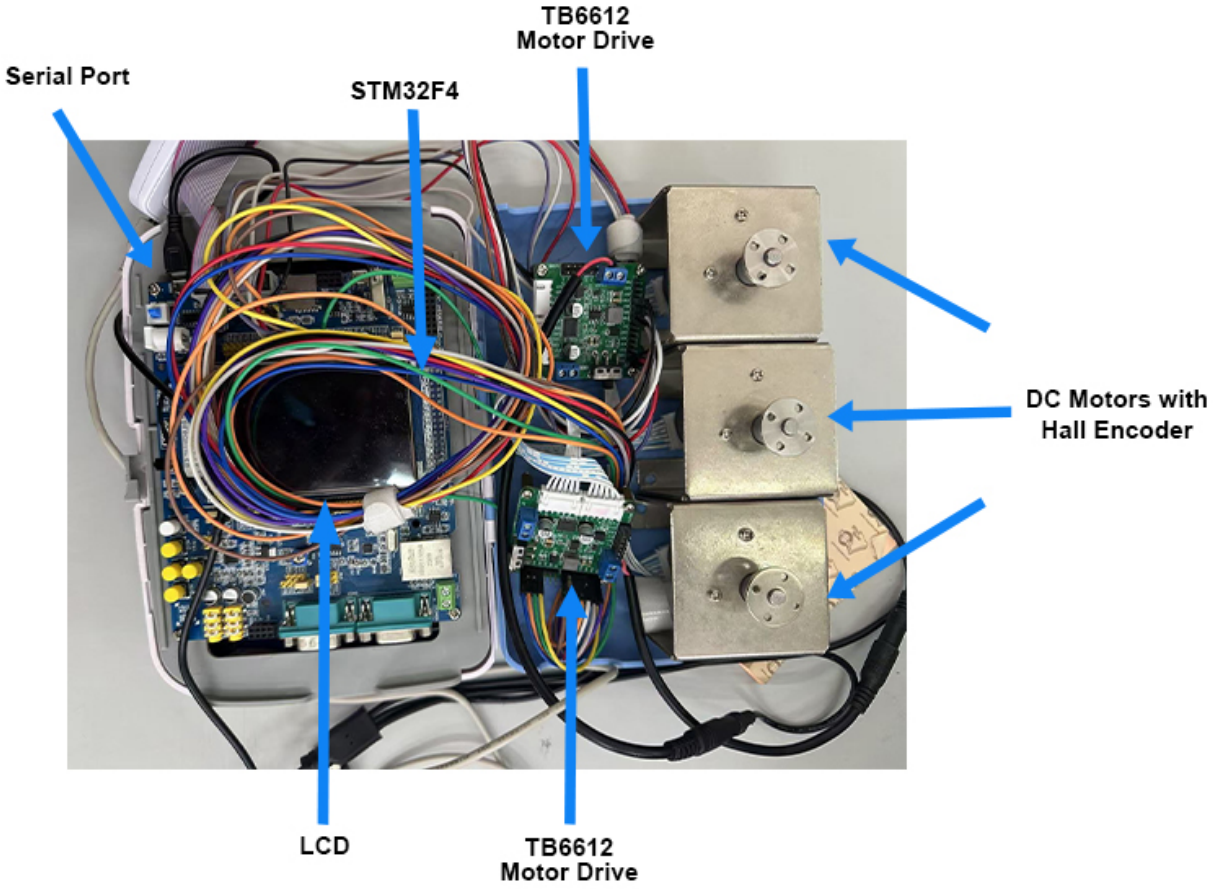

Fig. 8: Multi-DC motors system.

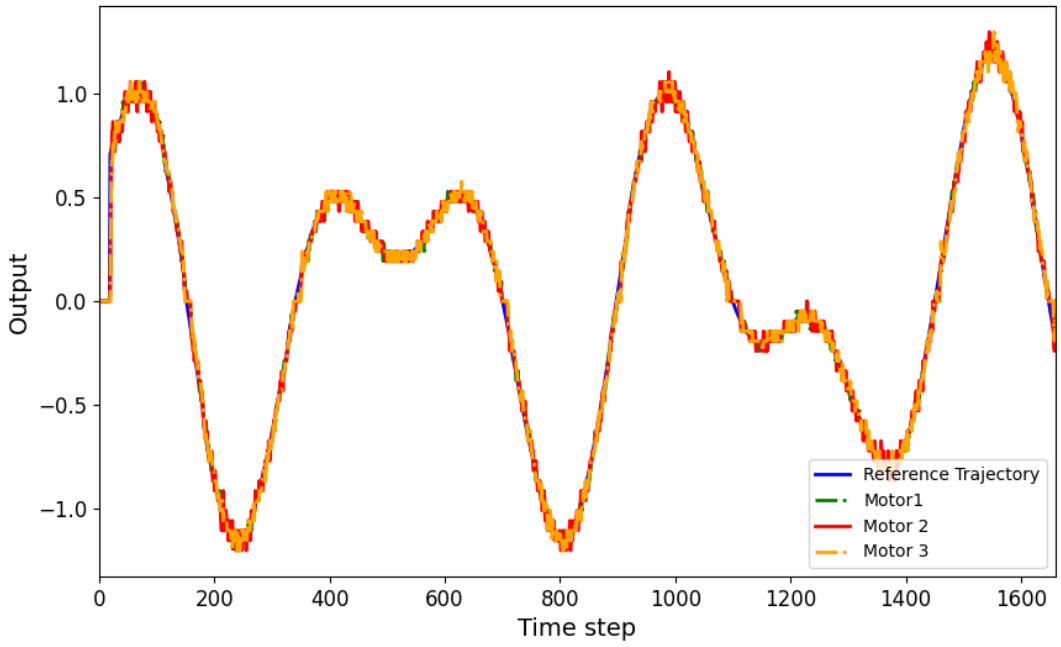

Fig. 9: Tracking performance of 3 DC motors for time-invariable desired trajectory.

Fig.9 shows the tracking performance of multi-DC motors. As presented, all DC motors successfully track the desired trajectory, demonstrating the effectiveness of the proposed control method. Where the reference trajectory is presented by blue.
Overall, the simulation results suggest that the proposed control system is capable of tracking a constant desired trajectory for multiple agents. While initial transient errors may accur, the system eventually reaches a steady-state condition with minimal tracking error. The variations in tracking performance among the agents highlight the potential influence of individual characteristics and external factors.

V. Conclusion

In this paper, a novel model-free adaptive sliding mode control was presented for multi-DC motors speed regulation. The proposed method achieves robust performance and adaptability in the presence of varying conditions. Utilizing a fixed topology represented by Laplacian matrice, the control strategy effectively manages the interconnections among multiple DC motors. The advantages and effectiveness of the proposed approach are demonstrated through detailed simulation results. Future work will focus on implementing the proposed approach in practical scenerios under hardware constraints.

References

[1] Lewis, F.L.; Zhang, H.; Hengster-Movric, K.; Das, A. Cooperative Control of Multi-Agent Systems: Optimal and Adaptive Control Design; Springer Science and Business Media: Berlin/Heidelberg, Germany, 2014.

[2] Bu, XH; Hou, ZS and Zhang, HW. Data-Driven Multiagent Systems Consensus Tracking Using Model Free Adaptive Control. *IEEE Transactions on Neural Networks and Learning Systems*, May 2018, 29(5): 1514-1524.

[3] Ren, W. Consensus strategies for cooperative control of vehicle formations. IET Control Theory Appl. 2007, 1, 505–512.

[4] Dong, S.; Wang, C.; Wen, S.; Gang, F. A synchronization approach to trajectory tracking of multiple mobile robots while maintaining time-varying formations. IEEE Trans. Robot. 2009, 25, 1074–1086.

[5] Z. Hou and S. Jin, "A novel data-driven control approach for a class of SISO nonlinear systems," *IEEE Transactions on Control Systems Technology*, vol. 19, no. 6, pp. 1549-1558, 2011.

[6] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215-233, 2007.

[7] M. Sampei, T. Tamura, T. Kobayashi, and N. Shibui, "Arbitrary path tracking control of articulated vehicles using nonlinear control theory," IEEE Trans. Control Syst. Technol., vol. 3, no. 1, pp. 125–131, Mar. 1995.

[8] W. Ren, R. W. Beard, and E. M. Atkins, "Information consensus in multivehicle cooperative control," *IEEE Control Systems*, vol. 27, no. 2, pp. 71-82, 2007.

[9] D. Xu, B. Jiang, and P. Shi, "Adaptive observer based data-driven control for nonlinear discrete-time processes," *IEEE Transactions on Automation Science and Engineering*, vol. 11, no. 4, pp. 1037-1045, 2014.

[10] R. Chi, Z. Hou, S. Jin, D. Wang, and C.-J. Chien, "Enhanced data-driven optimal terminal ILC using current iteration control knowledge," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 11, pp. 2939-2948, 2015.

[11] Ren, Y.; Hou, Z. Robust model-free adaptive iterative learning formation for unknown heterogeneous non-linear multi-agent systems. IET Control Theory Appl. 2020, 14, 654–663.

[12] C. L. P. Chen, G.-X. Wen, Y.-J. Liu, and F.-Y. Wang, "Adaptive consensus control for a class of nonlinear multiagent time-delay systems using neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 6, pp. 1217-1226, 2014.

[13] Zhou, N (Zhou, Ning); Deng, WX (Deng, Wenxiang); Yang, XW (Yang, Xiaowei); Yao, JY (Yao, Jianyong), "Continuous adaptive integral recursive terminal sliding mode control for DC motors," *International Journal of Control*, vol. 96, no. 9, pp. 2190-2200, June 2022.

[14] X. Liu, J. Lam, W. Yu, and G. Chen, "Finite-time consensus of multiagent systems with a switching protocol," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 4, pp. 853–862, Apr. 2016.

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

[16] Z. Hou and S. Jin, "A novel data-driven control approach for a class of discrete-time nonlinear systems," IEEE Trans. Control Syst. Technol., vol. 19, no. 6, pp. 1549–1558, Nov. 2011.

[17] D. Xu, B. Jiang, and F. Liu, "An improved data driven MDEL free adaptive constrained control for a solid oxide fuel cell," IET Control Theory Appl., vol. 10, no. 12, pp. 1412–1419, 2016.

[18] D. Xu, B. Jiang, and P. Shi, "A novel model free adaptive control design for multivariable industrial processes," IEEE Trans. Ind. Electron., vol. 61, no. 11, pp. 6391–6398, Nov. 2014.

[19] D. Xu, B. Jiang, and P. Shi, "Adaptive observer based data-driven control for nonlinear discrete-time processes," IEEE Trans. Autom. Sci. Eng., vol. 11, no. 4, pp. 1037–1045, Oct. 2014.

[20] Z. Pang, G. Liu, D. Zhou, and D. Sun, "Data-based predictive control for networked nonlinear systems with network-induced delay and packet dropout,"IEEE Trans. Ind. Electron., vol. 63, no. 2, pp. 1249–1257, Feb. 2016.

[21] H. Zhang and F. L. Lewis, "Adaptive cooperative tracking control of higher-order nonlinear systems with unknown dynamics," Automatica, vol. 48, no. 7, pp. 1432-1439, 2012.

[22] J. Liu, S. Vazquez, L. Wu, A. Marque, H. Gao, and L. G. Franquelo et al., "Extended state observer based sliding mode control for three-phase power converters," IEEE Trans. Ind. Electron., vol. 64, no. 1, pp. 22–31, Jan. 2017.

[23] Z.-G. Hou, L. Cheng, and M. Tan, "Decentralized robust adaptive control for the multiagent system consensus problem using neural networks," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 39, no. 3, pp. 636–647, Jun. 2009.

[24] Z. Hou and W. Huang, "The model-free learning adaptive control of a class of SISO nonlinear systems," *Proceedings of the American Control Conference*, Albuquerque, NM, USA, Jun. 1997, pp. 343–344.

[25] H. Su, G. Chen, X. Wang, and Z. Lin, "Adaptive second-order consensus of networked mobile agents with nonlinear dynamics," *Automatica*, vol. 47, no. 2, pp. 368–375, Feb. 2011.

[26] D. Xu, B. Jiang, and P. Shi, "Adaptive observer based data-driven control for nonlinear discrete-time processes," *IEEE Transactions on Automation Science and Engineering*, vol. 11, no. 4, pp. 1037–1045, Oct. 2014.

[27] H. Zhang and F. L. Lewis, "Adaptive cooperative tracking control of higher-order nonlinear systems with unknown dynamics," *Automatica*, vol. 48, no. 7, pp. 1432–1439, Jul. 2012.

[28] Z. Wu, X. Wang, and X. Zhao, "Backstepping terminal sliding mode control of DFIG for maximal wind energy captured," Int. J. Innovative Comput. Inf. Control, vol. 12, no. 5, pp. 1565–1579, 2016.

[29] X. Yan and C. Edwards, "Adaptive sliding-mode-observer-based fault reconstruction for nonlinear systems with parametric uncertainties," IEEE Trans. Ind. Electron., vol. 55, no. 11, pp. 4029–4036, Nov. 2008.

[30] J. Liu, W. Luo, X. Yang, and L. Wu, "Robust model-based fault diagnosis for PEM fuel cell air-feed system," IEEE Trans. Ind. Electron., vol. 63, no. 5, pp. 3261–3270, May 2016.

[31] A. Anuchin, A. Dianov and F. Briz, "Synchronous Constant Elapsed Time Speed Estimation Using Incremental Encoders," in IEEE/ASME Transactions on Mechatronics, vol. 24, no. 4, pp. 1893-1901, Aug. 2019, doi: 10.1109/TMECH.2019.2928950.

[32] S. Qin and T. Badgwell, "A survey of industrial model predictive control technology," Control Eng. Pract., vol. 11, pp. 733–764, 2003.

[33] A. Sharafian, V. Bagheri, and W. Zhang, "RBF neural network sliding mode consensus of multi-agent systems with unknown dynamical model of leader-follower agents," *International Journal of Control, Automation and Systems*, vol. 16, no. 2, pp. 749–758, 2018.

[34] X. Ma, F. Sun, H. Li, and B. He, "Neural-network-based integral sliding-mode tracking control of second-order multi-agent systems with unmatched disturbances and completely unknown dynamics," *International Journal of Control, Automation and Systems*, vol. 15, no. 4, pp. 1925–1935, 2017.

[35] R. Rahmani, H. Toshani, and S. Mobayen, "Consensus tracking of multi-agent systems using constrained neural-optimiser-based sliding mode control," *International Journal of Systems Science*, vol. 51, no. 14, pp. 2653–2674, 2020.

[36] Z. Peng, G. Wen, A. Rahmani, and Y. Yongguang, "Distributed consensus-based formation control for multiple nonholonomic mobile robots with a specified reference trajectory," *International Journal of Systems Science*, vol. 46, no. 8, pp. 1447–1457, 2015.