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

Tony Blaise Bimenyimana

Abstract

This paper introduces the model free adaptive sliding mode control (MFASMC) system for multi-DC motors speed regulation problem. The control law is derived by incorporating sliding surface. Then, measurement accuracy enhanced through quadruple-frequency data processing method ensuring consistent sampling periods and accurate encoder readings for DC Motors. Furthermore, a nonlinear multi-agent system with fixed communication topology is implemented for improvement of regulation. The compact form dynamic linearization (CFDL) technique refines the control gains and adapts the system to varying conditions. Finally, simulation results are given to demonstrate effectiveness of the proposed design method.

KEYWORDS

Model-free adaptive sliding mode control, quadruple-frequency data processing method, compact form dynamic linearization, nonlinear multiagent system, multi-DC motors speed regulation.

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 ccv] Because of the distributed and flexible structure, the MASs based approaches have significant potential in addressing coordination challenges for complex systems [1 ap]. 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 [2 cc], multi-robot systems[3 cc] and industrial machinery.

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 [10 cc]. 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 [21 cc]-[22 cc], 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 [23 cc]-[26 cc] 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[21]-[22].

In [8 ap], 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[27 cc]. 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 cc] - [30 cc]

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[8 ap], 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 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.[34-40 ccv]
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. Preliminaries

The set of real numbers is denoted by $\mathbb{R}$. For a given matrix $A \in \mathbb{R}^{n\times n}$, $\|A\|$ represents the matrix norm. The notation $\text{diag}(\cdot)$ refers to a diagonal matrix, and I signifies the identity matrix of appropriate dimensions. In the context of multi-agent systems, graph theory serves as an effective tool to model interaction topologies. A brief introduction to directed graphs within algebraic graph theory is provided. Let $G = (V, E, A)$ be the weighted directed graph, where $V = \{1, 2, \dots, N\}$ represents the set of vertices, $E \subseteq V \times V$ denotes the set of edges, and A is the adjacency matrix. Here, V also indexes the agents. If agent j can receive a message from agent i, then $(i, j) \in E$, where j is the child of i, the neighborhood of agent i is given by $N_i = \{j \in V | (j, i) \in E\}$. The weighted adjacency matrix $A = (a_{ij})$ is defined such that $a_{ii} = 0$, $a_{ij} = 1$ if $(j, i) \in E$; otherwise, $a_{ij} = 0$. The laplacian matrix of G is defined as $L = D - A$, where $D = \text{diag}(d_1^{\text{in}}, d_2^{\text{in}}, \dots, d_N^{\text{in}})$ and $d_i^{\text{in}} = \sum_{j=1}^{N} a_{ij}$ is the in-degree of vertex i. The graph is said to be strongly connected if there exists a path between any pair of vertices.

In this research, the speed regulation problem of DC motors is frequently examined under the assumption that all motors demonstrate identical dynamic characteristics. However, heterogeneity remains a fundamental characteristic of systems incorporating multi-DC motors. Even when motors are of the same type and share similar structural features, the parameters can never be exactly the same. This inherent variability complicates coordinated speed regulation across heterogeneous motors. Consider the multi-DC motor system consisting of N motors, where the interaction topology is represented by G. Assume that each motor i follows the nonlinear dynamics:

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

where $y_i(k) \in \mathbb{R}$ represents the output (speed of the DC motor), $u_i(k) \in \mathbb{R}$ is the control input (the voltage), and $f_i(\cdot)$ is an unknown nonlinear function, respectively.

In this scenario, multiple agents aim to track the consensus trajectory $y_d(k)$, which is exclusively accessible to the subset of agents. This trsajectory is assumed to be generated by a virtual leader designated as vertex 0. To model this interaction, the construction of the direct graph $G' = (V \cup \{0\}, E', A')$ where V denotes the set of agents, E' represents the edge set defining connections from agents to the virtual leader, and A' constitute a weighted adjacency matrix detailing the above-mentioned connections.

The following assumptions for nonlinear dynamics are given to facilitate our analysis.

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

Assumption 2: The model $y_i(k+1) = f_i(y_i(k), u_i(k))$ is generalized lipschitz, 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 $\Delta y_i(k+1) = y_i(k+1) - y_i(k)$ and b is a positive constant.

Remark 1: The practical applicability of the aforementioned assumptions to nonlinear systems has been extensively discussed in [1] and assumption 1 establishes a foundational criterion for controller design while, assumption 2 implies that the rate of change in the output of an agent in response to changes in the control input is bounded. This constraint ensures that, from an energetic perspective, finite changes in control input energy correspond to bounded changes in output energy rates, a crucial consideration for system stability and performance.

B. Linearization Technique

Under Assumptions 1 and 2, the unknown agent dynamics (1) can be transformed into the following dynamic linearization model and then the distributed control law will be designed based on it:

Lemma 1: Consider follower agents with dynamics (1) satisfying Assumptions 1 and 2. If $||\Delta u_i(k)|| \neq 0$ holds, then agent (1) can be transformed into a compact form dynamic linearization(CFDL) data model:

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

where $|\phi_i(k)| \leq b$ with the variable $\phi_i(k)$ named PPD, which is bounded for any time instant k.

Remark 2: By virtual of CFDL technique, the unknown nonlinear agent dynamics (1) are now transformed into an incremental form data description (2) in every operation point with a bounded PPD. In Implementation, with good selected algorithms, PPD can be estimated by using the I/O data of the agents and the estimated value can also be proved to be bounded, which is shown in Theorem 1. Here, the CFDL data model is derived under the condition $||\Delta u_i(k)|| \neq 0$. In fact, if the case $||\Delta u_i(k)|| = 0$ happens at the some sampling time, a new CFDL data model can also be established after shifting $\gamma_i \varepsilon \mathbb{Z}^+$ time steps until $u_i(k+\gamma_i) \neq u_i(k)$ holds. In the next section, the data model (2) will be used to design the distributed control law and derive the stability analysis

Let denote the following distributed measurement output of $\xi_i(k)$ for ith agents as follow:

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

Assumption 3: The communication graph $\bar{G}$ is fixed and strogly connected, with at least one follower agent able to access the trajectory of the leader.

III. Main Results

A. Model Free Adaptive Controller Design

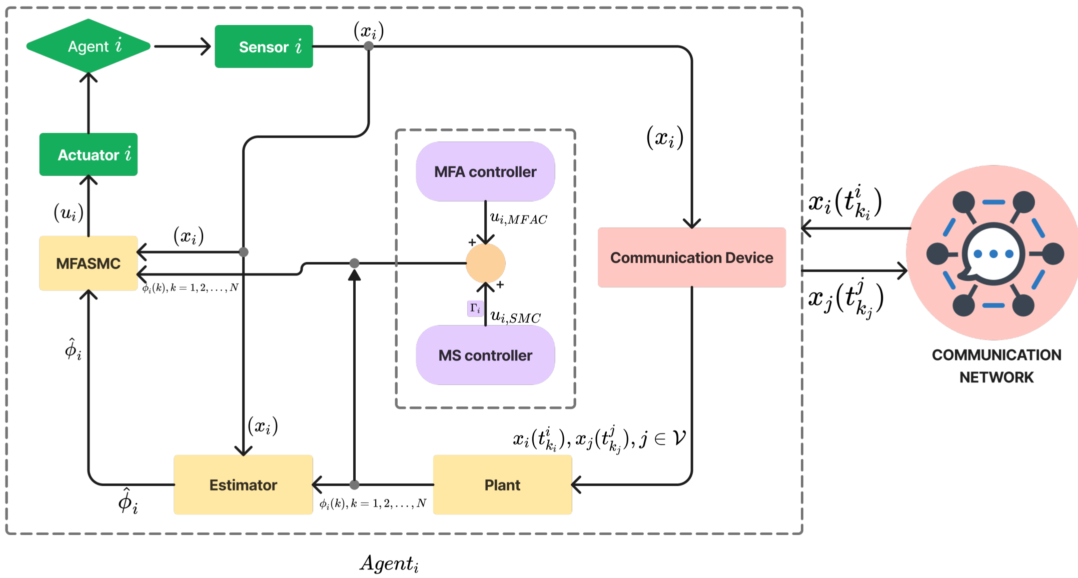

Fig. 1: Block diagram.

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:

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

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

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, guiding the behavior of system to ensure robust and accurate tracking of the desired trajectory.
The sliding mode surface is defined as:

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

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 robutness and 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

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.

Part 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 error 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:

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

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 $D = \text{diag}(d_1, d_2, d_3, \ldots, d_n)$. 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

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

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 Results

Consider the network comprising:

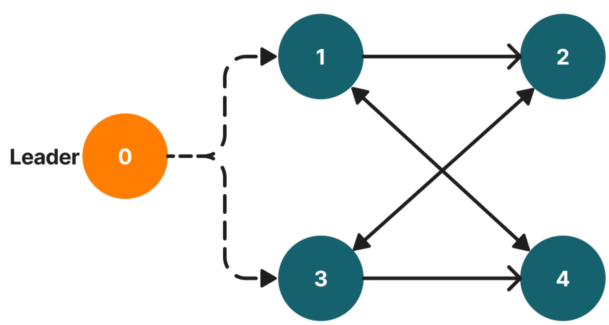

Fig. 2: Communication topology among agents.

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

A. Hardware Implementation

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.

B. Data Processing Method

The motor speed is measured in revolution per sencond (r/s) based on encoder measurements and the sampling interval t. The total number of encoder counts per revolution is calculated as $T = 4N_e Rr$, 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:

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

Each sampling interval triggers an interrupt where the controller samples the motor speed and updates control commands accordingly.

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 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 provided here 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.

C. Time Invariable Desired Trajectory

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 $T = 0.1$, $m = 350$, $\eta = 1$, $\mu = 1$. The MFA Controller parameters are given as $\rho = 1$, $\lambda = 50$ and $\alpha = 1$ with $\epsilon = 10^{-5}$.

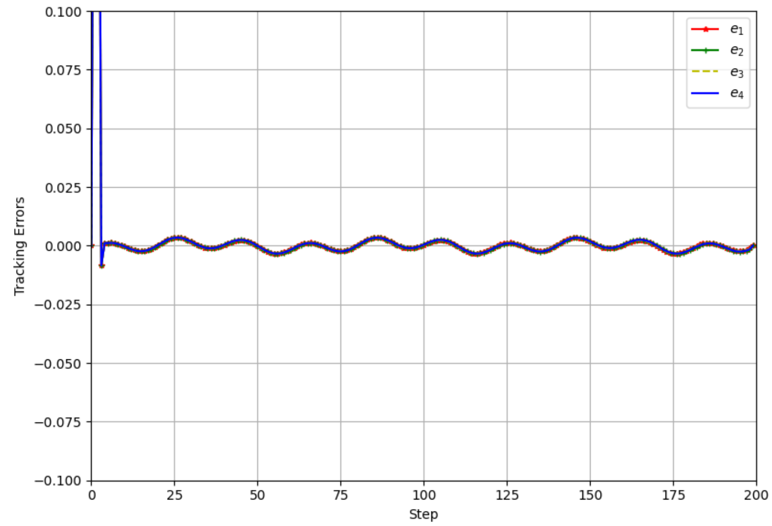

Fig. 3: Tracking errors for time varying desired trajectory.

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.

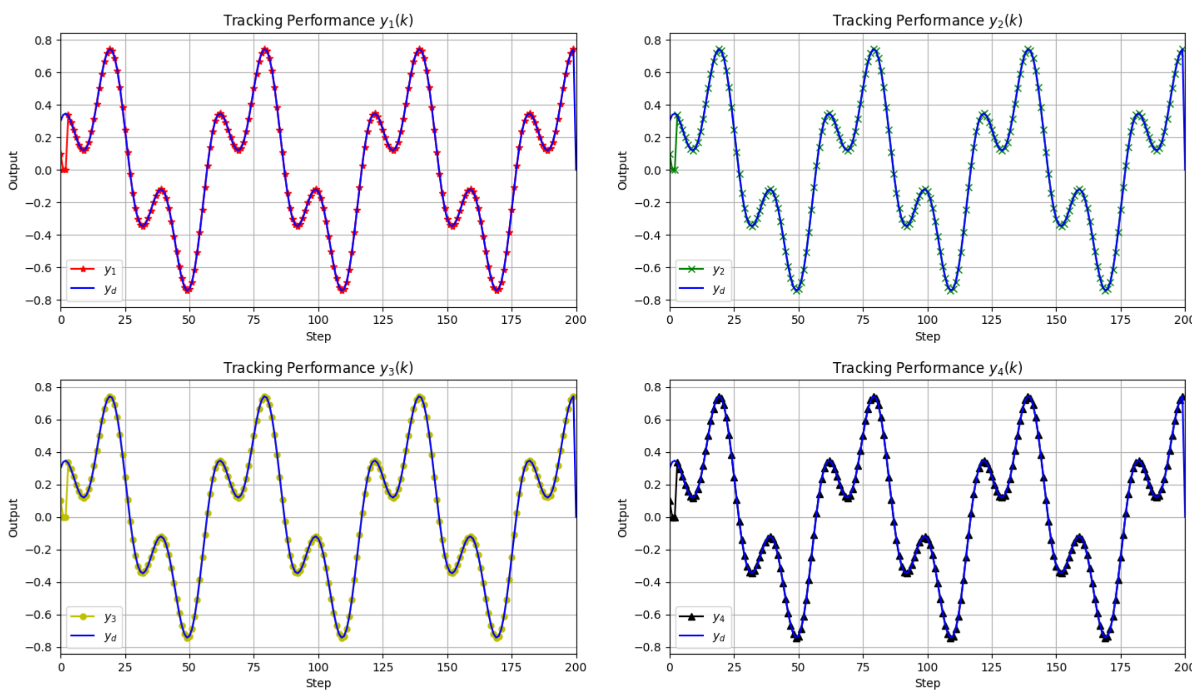

Fig. 4: Tracking performance of all agents for time-varying desired trajectory.

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.

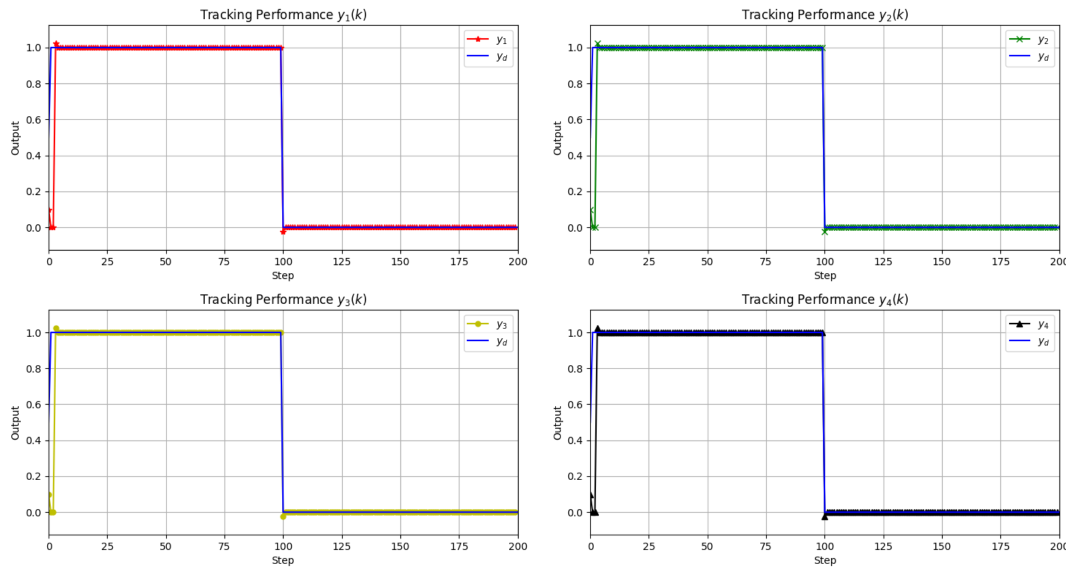

Fig. 5: Tracking performance of all agents for time-invariable desired trajectory.

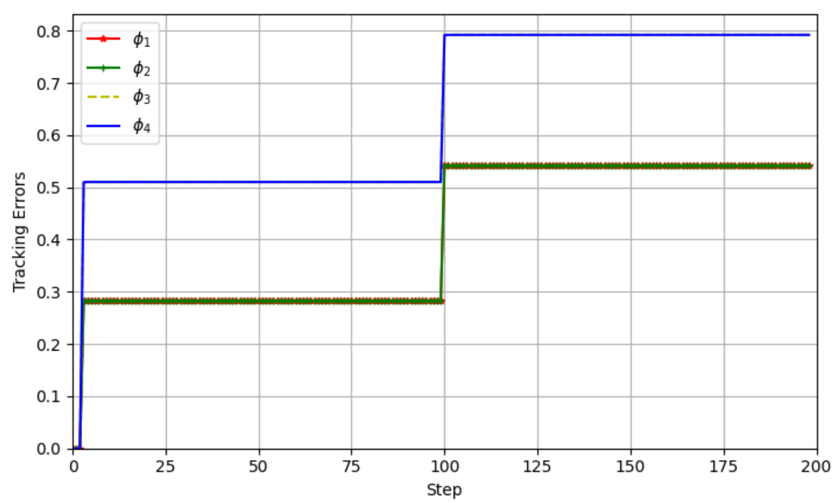

Fig. 6: PPD Estimation of all agents.

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

Overall, the simulation results suggest that the proposed control system is capable of tracking a constant desired trajectory for multiple agents. While there may be initial transient error, 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] 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.

[2] 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.

[3] 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.

[4] L. Moreau, "Stability of multiagent systems with time-dependent communication links," *IEEE Trans. Autom. Control*, vol. 50, no. 2, pp. 169-182, Feb. 2005.

[5] 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.

[6] 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.

[7] 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.

[8] 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.

[9] 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.

[10] 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.

[11] 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.

[12] 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.

[13] 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.

[14] 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.

[15] 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.

[16] 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.

[17] D. Meng, Y. Jia, J. Du, and J. Zhang, "On iterative learning algorithms for the formation control of nonlinear multi-agent systems," *Automatica*, vol. 50, no. 1, pp. 291–295, Jan. 2014.

[18] 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.

[19] 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.

[20] 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.

[21] 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.