

# A novel dynamic prescribed performance fuzzy-neural backstepping control for PMSM under step load\*

Xuechun Hu<sup>a</sup>, Yu Xia<sup>b,\*</sup>, Zsófia Lendek<sup>c</sup>, Jinde Cao<sup>d</sup> and Radu-Emil Precup<sup>e,f</sup>

<sup>a</sup>School of Mechanical Engineering, Guizhou University, Guizhou 550025, China

<sup>b</sup>State Key Laboratory of Mechanical System and Vibration, Shanghai Jiao Tong University, Shanghai 200240, China

<sup>c</sup>Department of Automation, Technical University of Cluj-Napoca, Memorandumului 28, 400114, Cluj-Napoca, Romania

<sup>d</sup>School of Mathematics Southeast University, Nanjing 211189, China

<sup>e</sup>Department of Automation and Applied Informatics, Politehnica University of Timisoara, 300223 Timisoara, Romania

<sup>f</sup>Center for Fundamental and Advanced Technical Research, Romanian Academy - Timisoara Branch, 300223 Timisoara, Romania

---

## ARTICLE INFO

*Keywords:*

PMSM

Prescribed performance function

Fuzzy neural backstepping control

Speed function

---

## ABSTRACT

In order to meet the performance requirements of permanent magnet synchronous motor (PMSM) systems with time-varying model parameters and input constraints under step load, this paper proposes a dynamic prescribed performance fuzzy-neural backstepping control approach. Firstly, a novel finite-time asymmetric dynamic prescribed performance function (FADPPF) is proposed to tackle the issues of exceeding predefined error, control singularity, and system instability that arise in the traditional prescribed performance function under load changes. To address model accuracy degradation and control quality deterioration caused by nonlinear time-varying parameters and input constraints in the PMSM system, a backstepping controller is designed by combining the speed function (SF), fuzzy neural network (FNN), and the proposed FADPPF. The FNN approximates nonlinear uncertain functions in the system model; the SF, as an error amplification mechanism, works together with FADPPF to ensure the transient and steady-state performance of the system. The stability of the devised control strategy is proved using Lyapunov analysis. Finally, simulation results demonstrate the dynamic self-adjusting ability and effectiveness of FADPPF under step load. In addition, the feasibility and superiority of the proposed control scheme are validated.

---

## 1. Introduction

Permanent Magnet Synchronous Motors (PMSMs) are characterized by their strong performance, high efficiency, and simple structure, playing a pivotal role in fields such as electric vehicles and robotics. However, the inherent nonlinearities of PMSM systems, coupled with multi-source disturbances under complex operating conditions, pose two challenges to the collaborative optimization of control system robustness and dynamic performance. On the one hand, parameter variations caused by motor temperature rise and abrupt changes in load torque necessitate controllers with strong anti-disturbance compensation capabilities. On the other hand, the interplay between control input saturation and model uncertainties makes the quantitative assurance of dynamic response quality a critical research focus. Adaptive fuzzy/neural network control [1], [2], [3], [4], [5] or active disturbance rejection control [6], [7] or sliding mode control [8], [9] has proven effective in enhancing system disturbance rejection capabilities. To address input saturation, auxiliary systems can be designed to suppress output oscillations or instability caused by saturated inputs [10], [11], or alternatively, control inputs can be directly formulated within unsaturated ranges through saturation-aware designs [12], [13]. To tackle the challenges arising from coupled effects of parameter perturbations, uncertain disturbances, and input saturation, improved control strategies have been developed, including backstepping control methods [14], [15], [16], non-recursive adaptive control approaches [17], [18], and sliding mode control techniques [19], [20]. These stability-oriented control schemes have effectively addressed parametric/model uncertainties, load torque disturbances, and input saturation constraints. Nevertheless, their schemes predominantly

---

\* This work was supported in part by the Guizhou Province Graduate Research Fund, under Grant YJSCXJH[2020]050, in part by the Romanian Ministry of Research, Innovation and Digitization, CNCS/CCCDI - UEFISCDI, under Grant ERANET-ENUAC-e-MATS, within PNCDI IV, and in part by the National Natural Science Foundation of China under Grant 52175022.

\*corresponding author

✉ hu567800@126.com (X. Hu); rainyx@sjtu.edu.cn (Y. Xia); zsofia.lendek@aut.utcluj.ro (Z. Lendek); jdcao@seu.edu.cn (J. Cao); radu.precup@aut.upt.ro (R. Precup)

focus on ensuring asymptotic convergence (e.g.,  $\lim_{t \rightarrow \infty} e(t) = 0$  or  $\|e(t)\| \leq \bar{e}$ ) and state boundedness ( $\|x(t)\| \leq \bar{x}$ ), while failing to impose quantitative constraints on transient performance metrics such as overshoot and convergence rate. This limitation may result in insufficient dynamic performance in practical applications, such as degraded trajectory tracking accuracy during high-speed operations of industrial robots.

Prescribed Performance Control (PPC) achieves quantitative constraints on tracking errors by constructing time-varying performance boundary functions, providing a breakthrough solution to the aforementioned problems [21], [22]. Its core lies in strictly restricting the error trajectory within a preset time-varying envelope, thereby explicitly regulating the transient response process. In recent years, the integration of PPC with adaptive fuzzy/neural control [23], [24], sliding mode control [25], and adaptive backstepping control [26], [27], [28] has demonstrated significant advantages in scenarios such as aircraft attitude control and vehicle path tracking. In response to the differentiated requirements for transient performance across various dynamic systems and engineering application scenarios, a variety of performance prescribed functions (PPFs) with different derivative forms and characteristics have been successively proposed, including homogeneous fractional-type [21], polynomial-type [29], exponential-type [22], fractional power-type [30], hyperbolic tangent function-type [31], sinusoidal-type [32], and cosine-type [33]. Existing methods employ symmetric constraint boundaries ( $|e(t)| < \rho_l(t) = \rho_u(t)$ ) or proportional constraint boundaries ( $\rho_l(t) = k\rho_u(t), k \in (0, 1)$ ) for enveloping control of error trajectories. However, research results indicate that tracking errors still exhibit unexpected overshoots. To address this, Bu et al. [34] proposed tunnel-type PPF (TPPF), which dynamically reconstructs asymmetric constraint boundaries through an event-triggered mechanism. Unlike static designs with fixed proportional coefficients  $k$  in asymmetric constraint strategies [27], [29], [31], the TPPF constructs an initial co-sided asymmetric constraint band satisfying  $\text{sgn}(\rho_u(0)) = \text{sgn}(\rho_l(0))$  based on the sign characteristics of the initial error  $e(0)$ . Under the TPPF constraints, overshoot behavior can be perfectly suppressed. However, the zero-overshoot characteristic of TPPF comes at the cost of sacrificing dynamic response speed, thereby limiting the convergence rate of errors.

Moreover, the PPFs involved in the aforementioned research lack self-adjustment capability. When input saturation is present, strategies that impose such "rigid" PPF constraints on error trajectories may cause control singularity. Input saturation is not a new research problem. For instance, previous studies such as [14], [15], [16], [17], [18], [19], [20] have effectively addressed instability caused by input saturation. However, these strategies are not directly applicable to control singularity phenomena induced by input saturation in the PPC. The core challenge in resolving control singularity in the PPC lies in preventing errors from crossing constraint boundaries under input saturation. When the reference trajectory changes rapidly, input saturation may occur, leading to boundary-crossing errors. To address this, Wang and Hu [35] introduced the rate of reference trajectory variation into PPF design, proposing a predictive performance control strategy where constraint boundaries adapt to the reference trajectory's slope. However, under system disturbances, this strategy may still fail to prevent boundary-crossing errors due to its inability to rapidly detect disturbance variations. To overcome this limitation, Bu et al. [36] developed a self-adjusting PPF strategy based on the derivative of tracking errors. Nevertheless, this approach requires assumptions that both disturbances and tracking errors are continuously differentiable. Furthermore, Ji et al. [37] designed a saturation-tolerant prescribed performance function (SPPF) that employs the control input signal as an independent variable to construct an auxiliary system for real-time expansion or recovery of error constraint boundaries, thereby effectively preventing tracking errors from crossing the constraint envelope. However, the SPPF may exhibit unnecessary constraint envelope expansion during self-adjustment, leading to a slower error convergence rate. Bu et al. [38] proposed a dynamic adjustment strategy for the PPF by designing an auxiliary system with the error signal as the independent variable. Nevertheless, such auxiliary systems, which generate self-adjustment signals through multiple integrations, may introduce lag in envelope self-adjustment, resulting in boundary-crossing errors.

Inspired by the above methods, this paper proposes a dynamic prescribed performance fuzzy-neural backstepping control method for PMSM systems subject to parameter perturbations, uncertain disturbances, and input saturation. The method integrates a finite-time asymmetric dynamic prescribed performance function (FADPPF), speed function (SF), and fuzzy neural network (FNN) within the framework of backstepping technique. The FADPPF is designed to address control singularity phenomena induced by input saturation and insufficient dynamic disturbance adaptation. The SF amplifies the tracking error to improve system sensitivity to the tracking error. The FNN deals with the parameter and model uncertainty. The main contributions of this paper can be summarized as follows:

- (1) To address the control singularity phenomenon and boundary adjustment lag issues of traditional PPFs under input saturation and dynamic disturbances, the FADPPF is proposed. This PPF consists of the basic constraint and the self-adjusting additional constraint.

- (a) Distinct from [21], [22], [26], [29], [31], [32], [33] and [34], the PPFs in these studies lack self-adjustment capability. Except for [34], existing schemes struggle to effectively suppress transient overshoot. Although [34] avoids overshoot, it significantly sacrifices error convergence speed. The proposed scheme strictly limits overshoot amplitude while maintaining satisfactory error convergence speed.
  - (b) Compared with the adjustment scheme in [35], the FADPPF eliminates the requirement of derivatives of disturbances and tracking errors. Compared with the schemes in [36], [37] and [38], the FADPPF requires no auxiliary system, enables rapid self-adjustment, and eliminates envelope adaptation lag.
  - (c) Unlike the synchronized boundary adjustment in [35], [36], [37] and [38], the proposed FADPPF allows independent adjustment of upper and lower constraint boundaries. This asymmetric design enhances the capability to constrain error trajectories, achieving superior transient performance.
- (2) Compared with the backstepping controllers with PPF in [26], [27], [28], this paper incorporates the SF into the prescribed performance backstepping control, thereby accelerating the error convergence speed and achieving better transient performance and steady-state accuracy.

The remainder of this paper is organized as follows. Section 2 introduces the mathematical model of surface-mounted PMSM and preliminaries. Section 3 designs the FADPPF. Section 4 presents the design and stability analysis of the dynamic prescribed performance fuzzy-neural backstepping controller. The simulation results are presented and analyzed in Section 5. Finally, Section 6 draws the conclusions.

## 2. Mathematical model and preliminaries

### 2.1. PMSM model

The cross-sectional diagram of the stator and rotor of the surface mounted PMSM is shown in Fig. 1, where the coordinate axes of the a-b-c three-phase stationary coordinate system are perpendicular to the three-phase axes A-X, B-Y, C-Z of the stator winding, respectively. The mathematical model of surface mount PMSM in d-q rotating coordinate system is [39]

$$\begin{cases} \frac{d\theta_e}{dt} = \omega_e = n_p\omega, \\ \frac{d\omega}{dt} = \frac{1}{J} \left( \frac{3}{2}n_p\psi i_q - b\omega - T_L \right), \\ \frac{di_q}{dt} = \frac{1}{L} (u_q - R_0 i_q - n_p\omega (L i_d + \psi)), \\ \frac{di_d}{dt} = \frac{1}{L} (u_d - R_0 i_d + n_p L \omega i_q), \end{cases} \quad (1)$$

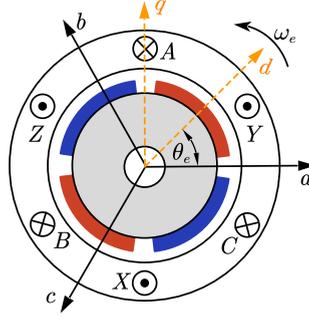
where  $u_d$ ,  $u_q$ ,  $i_d$ , and  $i_q$  represent the voltage and current on the d-axis and q-axis, respectively.  $R_0$ ,  $n_p$ ,  $\theta_e$ ,  $\omega_e$ ,  $\omega$ ,  $\psi$ ,  $L$ ,  $J$ ,  $T_L$ ,  $b$ , and  $t$  are the stator resistance, number of motor pole pairs, electrical angle, electrical angular velocity, mechanical angular velocity, permanent magnet flux linkage, inductance, moment of inertia, load torque, damping coefficient and time, respectively. The physical properties of the stator winding and magnet depend nonlinearly on the temperature; therefore,  $R_0$ ,  $\psi$  and  $L$  vary nonlinearly with temperature [40, 41].  $u_d$  and  $u_q$  are limited by the voltage of the DC link, that is, there are positive constants  $\bar{u}_d$  and  $\bar{u}_q$ , such that the inputs must satisfy  $|u_d| \leq \bar{u}_d$  and  $|u_q| \leq \bar{u}_q$ .

**Lemma 1.** (See [42]) If the variable  $u$  satisfies  $|u| \leq \bar{u}$ , then based on the hyperbolic tangent function  $g(v) = \bar{u} \tanh(v/\bar{u})$  and the mean value theorem,  $u$  can be written as  $u = g'(\delta v)v + d(v)$ , where  $0 < g'(\delta v) = \frac{\partial g(\delta v)}{\partial(\delta v)} = \frac{4}{(e^{\delta v/\bar{u}} + e^{-\delta v/\bar{u}})^2} \leq 1$ ,  $\delta$  is a constant belonging to  $(0, 1)$ ,  $d(v) = \text{sat}(v) - g(v)$ . Let  $G^m = g'(\delta v)$ . There exists a positive constant  $\underline{G}$  that satisfies  $0 < \underline{G} < G^m \leq 1$ .

According to Lemma 1, the control input  $u_x$  ( $x \in \{d, q\}$ ) can be written as [17]:

$$u_x = G_x^m v_x + d(v_x) = v_x + f(v_x), \quad (2)$$

where  $v_x$  is an auxiliary control input to be designed,  $G_x^m = g'(\delta_x v_x)$ ,  $\delta_x \in (0, 1)$  and  $f(v_x) = \bar{u}_x \tanh(v_x/\bar{u}_x) + d(v_x) - v_x$ . There is a positive constant  $\underline{G}_x$  that satisfies  $0 < \underline{G}_x < G_x^m \leq 1$ . Let  $D_x = \bar{u}_x - \bar{u}_x \tanh(1)$  and there exists  $|d(v_x)| \leq D_x$ .



**Fig. 1.** The cross-sectional diagram of the stator and rotor of the surface mounted PMSM.

## 2.2. Fuzzy neural network

The parameters such as  $R_0$ ,  $L$  and  $\psi$  in the PMSM model will deviate from their nominal values under the influence of load, temperature, and other factors. In order to ensure the performance of the PMSM control system, FNN is used to approximate nonlinear uncertainties in the PMSM mathematical model.

A smooth continuous function  $Y_j$  can be approximated by FNN [43]

$$\hat{Y}_j(x) = \varpi_j^T \gamma_j(x), \quad (3)$$

where  $\varpi_j = [\varpi_j^1, \dots, \varpi_j^n, \dots, \varpi_j^N]^T$ ,  $\gamma_j(x) = [\gamma_j^1(x), \dots, \gamma_j^n(x), \dots, \gamma_j^N(x)]^T$ , and

$$\gamma_j^n(x) = \prod_{i=1}^I \beta_j^{in}(x) / \sum_{n=1}^N \prod_{i=1}^I \beta_j^{in}(x), \quad (4)$$

where  $i = 1, \dots, I$ ,  $j = 1, \dots, J$  and  $n = 1, \dots, N$ ,  $\beta_j^{in}(x)$  represents the  $n$ th membership function (MF) of the  $i$ th input signal  $x_j^i$  of  $Y_j$ , as

$$\beta_j^{in}(x) = \exp \left[ - \left( \frac{x_j^i - \varepsilon_j^{in}}{\alpha_j^{in}} \right)^2 \right], \quad (5)$$

where  $\alpha_j^{in}$  and  $\varepsilon_j^{in}$  represent the width and center of  $\beta_j^{in}(x)$ ,  $[x_j^1, \dots, x_j^i, \dots, x_j^I] = X_j$  is the input signal for FNN to approximate  $Y_j$ .

**Lemma 2.** (See [44]) For any smooth continuous function  $Y(x)$  on the compact set  $O_x$ , there exists an optimal weight vector  $\varpi^*$  that satisfies

$$|Y(x) - \varpi^* \gamma(x)| \leq \tau, \quad (6)$$

where  $x$  is the argument of the function,  $\gamma(x) = [\gamma^1(x), \dots, \gamma^n(x), \dots, \gamma^N(x)]^T$  is a normalized membership vector calculated through fuzzy rules and  $\tau$  is an arbitrarily small positive constant.

The parameters of the FNN, including the number of membership functions, along with the centers and widths of membership degree functions, directly influence both approximation accuracy and computational load. The premise variables of FNN are selected as measurable states strongly correlated with nonlinear characteristics (e.g., stator current  $i_q$ , rotor angular velocity  $\omega$ , and error term  $e_1$ ), forming direct mapping of permanent magnet synchronous motor (PMSM) nonlinearities. Gaussian membership functions are employed for smooth approximation, with membership function centers uniformly distributed over the full variable range. The width parameters (set as 1/4-1/2 of center spacing) ensure 20%-50% overlap ratio to maintain state transition smoothness. The centers and widths of the membership functions are determined by comprehensively considering the computational burden and approximation accuracy.

### 3. Design of the FADPPF

In order to solve the problems of control singularity and system instability caused by load changes and input constraints, and effectively constrain the performance of the system under transient and steady-state conditions, a real-time self-adjusting FADPPF based on the system tracking error is proposed in this section.

In the prescribed performance control, the tracking error  $e(t)$  of the system is expected to meet

$$-\underline{p}(t) < e(t) < \bar{p}(t), \quad (7)$$

where  $e(t) = y(t) - y_r(t)$ ,  $y_r(t)$  represents the reference trajectory of  $y(t)$ ,  $\bar{p}(t)$  and  $\underline{p}(t)$  are the upper and lower constraint functions to be designed, also known as PPF.  $\bar{p}(t)$  and  $\underline{p}(t)$  jointly define the allowable range of  $e(t)$  to ensure that the convergence speed and steady-state error of  $e(t)$  meet the expected requirements.

In classic prescribed performance control,  $\bar{p}(t)$  and  $\underline{p}(t)$  are usually the same, that is  $\bar{p}(t) = \underline{p}(t) = p(t)$  and  $-\underline{p}(t) < e(t) < \bar{p}(t)$ . A function that satisfies the classical definition of PPF [22] is a monotone decreasing nonlinear function. And the pseudo-exponential type PPF is written as

$$p(t) = \begin{cases} \frac{\lambda_0 - \lambda_\infty}{1 + f_0 t^2} + \lambda_\infty, & t < t_0, \\ \lambda_\infty, & t \geq t_0, \end{cases} \quad (8)$$

where  $\lambda_0$ ,  $\lambda_\infty$ ,  $f_0$  and  $t_0$  are design constants.  $\lambda_0$  and  $\lambda_\infty$  represent the starting and ending values of  $p(t)$ , respectively. The size of  $f_0$  affects the slope of  $p(t)$ .  $t_0$  represents the time it takes for  $p(t)$  to decrease from  $\lambda_0$  to  $\lambda_\infty$ . The distance between the  $e(t)$  and PPF will affect the size of the control input  $u(t)$ . To ensure  $-p(0) < e(0) < p(0)$ ,  $\lambda_0$  usually takes the limit value of the system state. But when  $|e(0)|$  is farther away from  $\lambda_0$ , the smaller  $|u(0)|$ , the slower the convergence rate of  $e(t)$ . When  $|e(t)|$  is large, it is generally desirable for  $|u(t)|$  to be as large as possible in order for  $e(t)$  to converge to the desired interval in a shorter time. However, the slope influencing factors of existing PPFs are constant and cannot be adjusted in real-time based on  $e(t)$  to obtain larger control signals. Therefore, the slope adjustment function  $f_0(t)$  is proposed to modify the slope influence factor  $f_0$  in Eq. (8):

$$f_0(t) = \begin{cases} \frac{\lambda_1^2}{e(t)^2} + \lambda_2, & |e(t)| > \lambda_\infty, \\ \frac{\lambda_1^2}{\lambda_\infty^2} + \lambda_2, & \text{otherwise.} \end{cases} \quad (9)$$

In addition, the function described by Eq. (8) is not continuous at  $t = t_0$ , which causes the virtual control input containing PPF in subsequent controller design to be non-differentiable at certain points, leading to controller design difficulties. To address this problem, we propose the following  $p(t)$  whose slope can be adjusted in real time according to  $e(t)$  and is continuously smooth:

$$p(t) = \begin{cases} \left(\frac{t_0 - t}{t_0}\right)^{a_1} \frac{\lambda_0 - \lambda_\infty}{1 + f_0(t)t^2} + \lambda_\infty, & t < t_0, \\ \lambda_\infty, & t \geq t_0, \end{cases} \quad (10)$$

where  $\lambda_1$  and  $\lambda_2$  are constants to be determined,  $a_1$  is the amplitude adjustment coefficient to be determined.

The  $e(0)$  can be clearly divided into two possibilities:  $e(0) > 0$  and  $e(0) \leq 0$ . Therefore, combining the curve constraint formed by  $p(t)$  and the constant constraint, different  $\bar{p}(t)$  and  $\underline{p}(t)$  can be obtained by initially selecting the parameter  $a_0$  to effectively avoid unwanted tracking error overshoot. The PPF can be expressed as

$$\begin{cases} \bar{p}(t) = a_0 p(t) + (1 - a_0) \bar{\lambda}_\infty, \\ \underline{p}(t) = a_0 \underline{\lambda}_\infty + (1 - a_0) p(t), \end{cases} \quad (11)$$

where  $\bar{\lambda}_\infty$  and  $\underline{\lambda}_\infty$  are positive constants to be determined, representing constant constraints for  $e(0) \leq 0$  and  $e(0) > 0$ , respectively. And  $a_0 = \begin{cases} 1, & e(0) > 0, \\ 0, & \text{otherwise.} \end{cases}$

Further considering the influence of disturbance mutation and input constraint on  $e(t)$ , the self-adjusting term is designed to ensure that  $e(t)$  remains within the limits of  $\bar{p}(t)$  and  $\underline{p}(t)$  even in the event of disturbance mutation or

input saturation, in order to avoid control singularity and system instability. The self-adjusting term corresponding to the curve constraint is  $p_{\Delta 1}(t)$ , and the self-adjusting term corresponding to the constant constraint is  $p_{\Delta 2}(t)$ . Based on  $p(t)$ , the time contraction function  $f_i(t)$ ,  $i = 1, 2$  and amplitude amplification function  $f_{\Delta i}(t)$  are introduced to design  $p_{\Delta i}(t)$  as:

$$p_{\Delta 1}(t) = \begin{cases} \left(\frac{t_0 - f_1(t)t}{t_0}\right)^{a_1} \frac{(\lambda_0 - \lambda_\infty)f_{\Delta 1}(t)}{1 + f_0(t)(f_1(t)t)^2}, & t < t_0, \\ (1 - f_1(t))^{a_1} \frac{(\lambda_0 - \lambda_\infty)f_{\Delta 1}(t)}{1 + f_0(t)(f_1(t)t_0)^2}, & t \geq t_0, \end{cases} \quad (12)$$

$$p_{\Delta 2}(t) = (1 - f_2(t))^{a_1} \frac{(\lambda_0 - a_0 \lambda_\infty - (1 - a_0) \bar{\lambda}_\infty) f_{\Delta 2}(t)}{1 + f_0(t)(f_2(t)t_0)^2}. \quad (13)$$

When the control input is saturated and  $e(t)$  exceeds the designed safety threshold, the disturbance response part corresponding to the curve constraint makes the value of  $p_{\Delta 1}(t)$  equal to  $f_{\Delta 1}(t)$  times the value of  $p(t)$  at a certain point in the past through the time contraction function  $f_1(t)$  and the amplitude amplification function  $f_{\Delta 1}(t)$ . Similarly, the disturbance response part corresponding to the constant constraint makes the value of  $p_{\Delta 2}(t)$  equal to  $f_{\Delta 2}(t)$  times the value of  $p(t)$  at a certain point in the past through the time contraction function  $f_2(t)$  and the amplitude amplification function  $f_{\Delta 2}(t)$ .  $f_i(t)$  and  $f_{\Delta i}(t)$  are functions designed according to desired error convergence speed. Meanwhile, the amplitude amplification function  $f_{\Delta i}(t)$  should be designed to ensure that  $p_{\Delta i}(t)$  is continuous and differentiable. In this paper, we use

$$J_1 = \lambda_3 p(t) / |e(t)|, \quad (14)$$

$$J_2 = \lambda_3 \left[ (1 - a_0) \bar{\lambda}_\infty + a_0 \lambda_\infty \right] / |e(t)|, \quad (15)$$

$$f_i(t) = (J_i)^{a_i+1}, \quad (16)$$

$$f_{\Delta i}(t) = \tanh \left[ \lambda_4 (J_i^{-1} - 1) \right] \tanh \left\{ \lambda_3 (J_i^{-1} - 1) / \left[ \lambda_5 (1 - \lambda_3) \right] \right\}, \quad (17)$$

where  $a_2, a_3, \lambda_3, \lambda_4$  and  $\lambda_5$  are constants to be determined.

The trigger conditions make use of  $\bar{p}(t)$  and  $\underline{p}(t)$  are designed as  $J_a = \lambda_3 p(t) / e(t)$ ,  $J_b = \lambda_3 \bar{\lambda}_\infty / e(t)$  and  $J_c = \lambda_3 \lambda_\infty / e(t)$ . Then, the self-tuning selection factors  $\bar{a}$  and  $\underline{a}$  of the upper and lower constraint functions are used to determine whether to trigger self adjustment. The self-tuning trigger conditions are designed as:

(1) When  $e(0) > 0$  and  $0 < J_a \leq 1$ ,  $\bar{a} = 1$ . When  $e(0) \leq 0$  and  $0 < J_b \leq 1$ ,  $\bar{a} = 1$ . In other cases,  $\bar{a} = 0$ .

(2) When  $e(0) > 0$  and  $0 < -J_c \leq 1$ ,  $\underline{a} = 1$ . When  $e(0) \leq 0$  and  $0 < -J_a \leq 1$ ,  $\underline{a} = 1$ . In other cases,  $\underline{a} = 0$ .

The self-tuning trigger conditions of the upper and lower constraint functions are independent and do not affect each other.

The constant constraint formed by  $\bar{\lambda}_\infty$  or  $\lambda_\infty$  and the curve constraint formed by  $p(t)$  are collectively referred to as basic constraint. The nonlinear constraints generated by  $p_{\Delta i}(t)$ ,  $i = 1, 2$  are called self-adjusting additional constraint. The FADPPF consists of basic constraint and self-adjusting additional constraint, expressed as:

$$\begin{cases} \bar{p}(t) = a_0 (p(t) + \bar{a} p_{\Delta 1}(t)) + (1 - a_0) (\bar{\lambda}_\infty + \bar{a} p_{\Delta 2}(t)), \\ \underline{p}(t) = a_0 (\lambda_\infty + \underline{a} p_{\Delta 2}(t)) + (1 - a_0) (p(t) + \underline{a} p_{\Delta 1}(t)). \end{cases} \quad (18)$$

If self-adjustment is triggered,  $\bar{a} = 1$  or  $\underline{a} = 1$ ,  $\bar{p}(t)$  or  $\underline{p}(t)$  is not belong to the monotonic function.

The parameter characteristics of the FADPPF and their control performance impacts are summarized as follows:

- (1) Initial boundary parameter ( $\lambda_0$ ): Determines the starting point and maximum self-adjustment amplitude of the PPF. Excessively high values can reduce response speed, while too low values risk control singularity.
- (2) Steady-state boundary parameters ( $\lambda_\infty, \bar{\lambda}_\infty, \lambda_\infty$ ): Overly large values degrade steady-state accuracy, while too small settings require stronger control inputs and may exacerbate chattering.

- (3) Self-adjustment trigger threshold factor ( $\lambda_3$ ): Critical for the PPF's dynamic performance.  $\lambda_3 \geq 1$  causes adjustment hysteresis (increasing constraint violation risk), while  $\lambda_3 < 0.8$  produces overly conservative adjustments (impairing transient response speed). Recommended range:  $0.8 < \lambda_3 < 1$ .
- (4) Slope influence factor ( $\lambda_1, \lambda_2, \lambda_4, \lambda_5, a_1, a_2, a_3, t_0$ ): Collectively determine error convergence rate. Their synergy governs convergence speed and transient response quality.

**Remark 1.** Using Eq. (10) as an example to explain the function of parameter  $a_1$ . In Eq. (10),  $\left(\frac{t_0-t}{t_0}\right)^{a_1}$  ensures that the function is continuous at  $t = t_0$ , where the larger  $a_1$  is, the faster  $\left(\frac{t_0-t}{t_0}\right)^{a_1}$  decreases as  $t$  increases, approaching 0 from 1 more rapidly. If the value of  $a_1$  is large, combined with the influence of  $f_0(t)$  on the slope of  $p(t)$ , unexpected input saturation and triggering unnecessary function self adjustment may occur. Therefore, it is not recommended to take the value of  $a_1$  as more than  $\frac{4\lambda_0}{e(0)}$ . Moreover, in controller design, the differentiation of  $p(t)$  is involved. To reduce the computational complexity of the controller, it is recommended to take an integer for  $a_1$ . In summary, we recommend  $a_1$  to be an integer within  $\left[1, 4\left\lfloor\frac{\lambda_0}{e(0)}\right\rfloor\right]$ , where  $\lfloor \cdot \rfloor$  represents the floor function.

**Remark 2.** The upper and lower boundaries described by the existing asymmetric PPFs are both nonlinear functions, and the constrained boundary formed is like a funnel, and there may still be unwanted overshoot. The basic constraint boundary of FADPPF proposed in this paper is like a semi-funnel. For example, when  $e(0)$  is negative, the lower boundary is a monotone rising nonlinear curve, and the upper boundary is the constant value, which is the upper limit of the steady-state interval. The time when  $e(t)$  first enters the steady-state interval is the time to reach the expected accuracy, and there will be no unexpected overshoot behavior.

**Remark 3.** The FADPPF realizes dynamic self-adjustment based on the idea of event triggering. When the error is within the boundary of the basic constraint, the self-adjusting additional constraint part is 0. When the error exceeds the designed threshold  $J_a, J_b$ , or  $J_c$ , the constraint boundary is triggered to dynamically self adjust according to the error situation, and this self-adjustment is only to adjust the single boundary corresponding to the exceeded threshold, rather than adjusting the upper and lower boundaries simultaneously as shown in the existing PPFs.

**Remark 4.** In practice, to avoid division by zero when  $e(t) = 0$ , the constant  $a_4$  is introduced, and  $0 \ll a_4 < \lambda_3 < 1$ .  $J_a, J_b$  and  $J_c$  are modified to

$$J_a = \begin{cases} \lambda_3 p(t) / (a_4 \lambda_\infty), & |e(t)| < a_4 \lambda_\infty, \\ \lambda_3 p(t) / e(t), & \text{otherwise,} \end{cases} \quad (19)$$

$$J_b = \begin{cases} \lambda_3 / a_4, & e(t) < a_4 \bar{\lambda}_\infty, \\ \lambda_3 \bar{\lambda}_\infty / e(t), & \text{otherwise,} \end{cases} \quad (20)$$

$$J_c = \begin{cases} -\lambda_3 / a_4, & e(t) > -a_4 \underline{\lambda}_\infty, \\ \lambda_3 \underline{\lambda}_\infty / e(t), & \text{otherwise.} \end{cases} \quad (21)$$

## 4. Controller design and stability analysis

Considering the nonlinear time-varying parameters and constrained control inputs in the PMSM, a dynamic prescribed performance fuzzy-neural backstepping controller is constructed. The controller is based on the proposed FADPPF, combined with the error amplification mechanism SF, to ensure the transient and steady-state performance of the PMSM. The unknown functions in the PMSM model and the complex differential functions in the backstepping process are estimated by FNN. The control scheme is shown in Fig. 2.

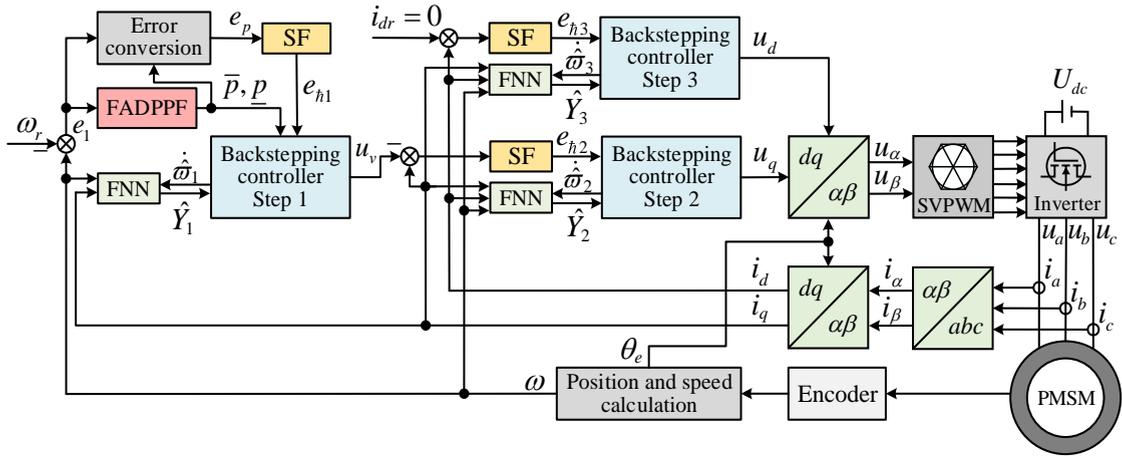


Fig. 2. Diagram of the control scheme.

#### 4.1. Controller design

The errors of the PMSM system are defined as

$$e_1 = \omega - \omega_r, \quad (22)$$

$$e_2 = i_q - u_v, \quad (23)$$

$$e_3 = i_d, \quad (24)$$

wherein  $\omega_r$  represents the reference trajectory of  $\omega$  and  $u_v$  is the virtual control input. To ensure that the tracking error  $e_1$  remains within the range defined by FADPPF,  $e_1$  is transformed into a new variable  $e_p$  as [33, 45]:

$$e_p = \frac{e_1}{(\bar{p}(t) - e_1)(\underline{p}(t) + e_1)}. \quad (25)$$

To improve the convergence speed of the tracking error, the SF is introduced into the controller design. When  $t \geq 0$ , the SF  $\hat{h}(t)$  is continuous and differentiable, as [46]

$$\hat{h}(t) = \begin{cases} k_2^4 \delta(t) / \left[ (1 - k_1)(k_2 - t)^4 + k_1 k_2^4 \delta(t) \right], & 0 \leq t < k_2, \\ 1/k_1, & t \geq k_2, \end{cases} \quad (26)$$

where  $k_1 \in (0, 1]$  and  $k_2 \in (0, \infty)$  are design parameters.  $\delta(t)$  is a smoothing function, which requires  $\delta(0) = 1$  and  $\dot{\delta}(t) \geq 0$ .  $k_1$  requires balance between dynamic adjustment demands and stability - dynamic regulation capability diminishes when  $k_1$  approaches 1.  $k_2$  demands careful trade-off between response speed and overshoot suppression, as excessive  $k_2$  values may prolong transient processes. In this paper,  $\delta(t) = t^2 + 1$ .

Based on the SF, the acceleration errors  $e_{hj}$  are considered

$$e_{hj} = \begin{cases} \hat{h}(t) e_p, & j = 1, \\ \hat{h}(t) e_j, & j = 2, 3. \end{cases} \quad (27)$$

The specific design process of the dynamic prescribed performance fuzzy-neural backstepping controller is divided into the following three steps.

**Step 1:** The time derivative of  $e_{h1}$  is

$$\dot{e}_{h1} = \dot{\hat{h}}(t) e_p + \hat{h}(t) [p_a (Y_1 + e_2 + u_v - \dot{\omega}_r) + p_b], \quad (28)$$

where  $p_a = \frac{\bar{p}(t)\underline{p}(t)+e_1^2}{(\bar{p}(t)-e_1)^2(\underline{p}(t)+e_1)^2}$ ,  $p_b = -\frac{(\bar{p}(t)\dot{\underline{p}}(t)+\dot{\bar{p}}(t)\underline{p}(t))e_1+(\dot{\bar{p}}(t)-\dot{\underline{p}}(t))e_1^2}{(\bar{p}(t)-e_1)^2(\underline{p}(t)+e_1)^2}$ ,  $Y_1 = \frac{[(1.5n_p\psi-J)i_q-b\omega-T_L]}{J}$ . Since  $J$ ,  $\psi$  and  $T_L$  are uncertain and time-varying,  $Y_1$  is approximated by a FNN. According to (3) and (6),  $X_1 = [\omega, i_q]$ ,  $Y_1$  can be written as

$$Y_1 \leq \varpi_1^{*T} \gamma_1 + \tau_1, \quad (29)$$

where  $\tau_1$  is the estimation error. Calculate  $\gamma_1$  according to (4) and (5). The estimated value  $\hat{\varpi}_1$  of  $\varpi_1^*$  is calculated by the adaptive law designed later.

Design the first Lyapunov function  $V_1$  as

$$V_1 = e_{h1}^2/2. \quad (30)$$

The derivative of  $V_1$  is

$$\dot{V}_1 = \dot{h}(t) e_p e_{h1} + \dot{h}(t) e_{h1} [p_a (Y_1 + e_2 + u_v - \dot{\omega}_r) + p_b]. \quad (31)$$

The virtual control input  $u_v$  and the adaptive law of  $\hat{\varpi}_1$  are constructed as

$$u_v = \dot{\omega}_r - c_1 e_p - \hat{\varpi}_1^T \gamma_1 - 0.5 \dot{h}(t) p_a e_{h1} - \frac{\dot{h}(t) e_p}{p_a \dot{h}(t)} - \frac{p_b}{p_a}, \quad (32)$$

$$\dot{\hat{\varpi}}_1 = v_1 (\dot{h}(t) p_a e_{h1} \gamma_1 - b_1 \hat{\varpi}_1), \quad (33)$$

where  $c_1$ ,  $v_1$  and  $b_1$  are positive parameters to be determined.

Substituting (29) and (32) into (31) gives

$$\begin{aligned} \dot{V}_1 &= p_a \dot{h}(t) e_{h1} (Y_1 - \hat{\varpi}_1^T \gamma_1 + e_2 - c_1 e_p - \dot{h}(t) p_a e_{h1}/2) \\ &\leq p_a e_{h1} e_{h2} - c_1 p_a e_{h1}^2 + p_a \dot{h}(t) e_{h1} \tilde{\varpi}_1^T \gamma_1 + \chi_1, \end{aligned} \quad (34)$$

where  $\tilde{\varpi}_1 = \varpi_1^* - \hat{\varpi}_1$  and  $\chi_1 = \tau_1^2/2$ .

**Step 2:** The time derivative of  $e_{h2}$  is

$$\dot{e}_{h2} = \dot{h}(t) e_2 + \dot{h}(t) [u_q - R_0 i_q - n_p \omega (L i_d + \psi)] / L - \dot{h}(t) \dot{u}_v. \quad (35)$$

Let  $L_t = \frac{1}{L}$ ,  $\bar{L}_t = \frac{1}{L_{\min}}$ ,  $\underline{L}_t = \frac{1}{L_{\max}}$ ,  $L_{\min}$  and  $L_{\max}$  represent the minimum and maximum values of  $L$  within the operating temperature range of the PMSM. Define the second Lyapunov function  $V_2$  as

$$V_2 = V_1 + e_{h2}^2/2. \quad (36)$$

Taking the derivative of  $V_2$ :

$$\dot{V}_2 = \dot{V}_1 + \dot{h}(t) e_{h2} (Y_2 - p_a e_p - \dot{h}(t) e_{h2} + L_t u_q), \quad (37)$$

where  $Y_2 = -L_t R_0 i_q - L_t n_p \omega (L i_d + \psi) + \left(\frac{\dot{h}^2(t)}{2} + \frac{\dot{h}(t)}{h(t)}\right) e_2 - \dot{u}_v + p_a e_p$ .

Using the FNN to estimate  $Y_2$ ,  $X_2 = [\omega, i_d, i_q, e_1, e_2]$ ,  $Y_2$  can be written as

$$Y_2 \leq \varpi_2^{*T} \gamma_2 + \tau_2, \quad (38)$$

where  $\tau_2$  is the estimation error. Calculate  $\gamma_2$  according to (4) and (5). The estimated value  $\hat{\varpi}_2$  of  $\varpi_2^*$  is calculated by the adaptive law designed later.

Considering the limitation of q-axis control input  $u_q$ ,  $u_q$  is designed as

$$u_q = \begin{cases} \bar{u}_q \text{sign}(v_q), & |v_q| \geq \bar{u}_q, \\ v_q, & |v_q| < \bar{u}_q, \end{cases} \quad (39)$$

$$v_q = -\hat{\theta}_2^T \gamma_2 - c_2 e_2 - 0.5 \dot{h}(t) e_{h2}, \quad (40)$$

where  $c_2$  is the design parameter,  $\hat{\theta}_2$  is an auxiliary variable and  $\hat{\theta}_2 = \hat{\varpi}_2 / (\underline{L}_t \underline{G}_q)$ . The adaptive law of  $\hat{\theta}_2$  is designed as

$$\dot{\hat{\theta}}_2 = v_2 (\dot{h}(t) e_{h2} \gamma_2 - b_2 \hat{\theta}_2), \quad (41)$$

where  $v_2$  and  $b_2$  are positive parameters to be determined.

Substituting Eqs. (38)-(40) and (2) into (37),  $\dot{V}_2$  can be updated as

$$\begin{aligned} \dot{V}_2 &= \dot{V}_1 + \dot{h}(t) e_{h2} (Y_2 - p_a e_p - \dot{h}(t) e_{h2} + L_t v_q) + L_t \dot{h}(t) e_{h2} d (v_q) \\ &\leq -c_1 p_a e_{h1}^2 - c_2 L_t e_{h2}^2 + p_a \dot{h}(t) e_{h1} \tilde{\varpi}_1^T \gamma_1 + \dot{h}(t) e_{h2} \tilde{\varpi}_2^T \gamma_2 + \chi_2, \end{aligned} \quad (42)$$

where  $\chi_2 = \chi_1 + \tau_2^2/2 + L_t f^2(v_q)/2$ ,  $\tilde{\varpi}_2 = \varpi_2^* - \hat{\varpi}_2$ ,  $\theta_2^* = \varpi_2^* / (\underline{L}_t \underline{G}_q)$  and  $\tilde{\theta}_2 = \theta_2^* - \hat{\theta}_2$ .

**Step 3:** The time derivative of  $e_{h3}$  is

$$\dot{e}_{h3} = \dot{h}(t) e_3 + \dot{h}(t) (L_t u_d + L_t n_p L \omega i_q - L_t R_0 i_d). \quad (43)$$

Design the third Lyapunov function  $V_3$  as

$$V_3 = V_2 + e_{h3}^2/2. \quad (44)$$

Taking the derivative of  $V_3$  yields

$$\dot{V}_3 = \dot{V}_2 + \dot{h}(t) e_{h3} (Y_3 - \dot{h}(t) e_{h3}/2 + L_t u_d), \quad (45)$$

where  $Y_3 = L_t n_p L \omega i_q - L_t R_0 i_d + \left(\frac{\dot{h}^2(t)}{2} + \frac{\dot{h}(t)}{h(t)}\right) e_3$ . And according to FNN,  $X_3 = [\omega, i_d, i_q, e_3]$ ,  $Y_3$  is approximated as

$$Y_3 \leq \varpi_3^* \gamma_3 + \tau_3, \quad (46)$$

where  $\tau_3$  is the estimation error. Calculate  $\gamma_3$  according to (4) and (5). The estimated value  $\hat{\varpi}_3$  of  $\varpi_3^*$  is calculated by the adaptive law designed later.

Construct the d-axis control input  $u_d$  and adaptive law of  $\hat{\theta}_3$  as

$$u_d = \begin{cases} \bar{u}_d \text{sign}(v_d), & |v_d| \geq \bar{u}_d, \\ v_d, & |v_d| < \bar{u}_d, \end{cases} \quad (47)$$

$$v_d = -\hat{\theta}_3^T \gamma_3 - c_3 e_3 - 0.5 \dot{h}(t) e_{h3}, \quad (48)$$

$$\dot{\hat{\theta}}_3 = v_3 (\dot{h}(t) e_{h3} \gamma_3 - b_3 \hat{\theta}_3), \quad (49)$$

where  $c_3$ ,  $v_3$  and  $b_3$  are positive parameters to be determined, and  $\hat{\theta}_3 = \hat{\varpi}_3 / (\underline{L}_t \underline{G}_d)$ .

$\dot{V}_3$  can be updated to

$$\begin{aligned} \dot{V}_3 &= \dot{V}_2 + \dot{h}(t) e_{h3} (Y_3 - 0.5 \dot{h}(t) e_{h3} + L_t v_d) + L_t \dot{h}(t) e_{h3} d (v_d) \\ &\leq -c_1 p_a e_{h1}^2 - c_2 L_t e_{h2}^2 - c_3 L_t e_{h3}^2 + p_a \dot{h}(t) e_{h1} \tilde{\varpi}_1^T \gamma_1 + \dot{h}(t) e_{h2} \tilde{\varpi}_2^T \gamma_2 + \dot{h}(t) e_{h3} \tilde{\varpi}_3^T \gamma_3 + \chi_3, \end{aligned} \quad (50)$$

where  $\chi_3 = \chi_2 + \tau_3^2/2 + L_t f^2(v_d)/2$ ,  $\tilde{\varpi}_3 = \varpi_3^* - \hat{\varpi}_3$ ,  $\theta_3^* = \varpi_3^* / (\underline{L}_t \underline{G}_d)$  and  $\tilde{\theta}_3 = \theta_3^* - \hat{\theta}_3$ .

Based on the above three steps, control inputs (39) and (47), along with the adaptive laws (33), (41), and (49), are derived. The detailed block diagram of the control framework is shown in Fig. 3.



Define the compact set  $\mathcal{G}_0 = \{X \mid V(X) \leq \chi_V\}$ . All signals of closed-loop system are bounded,  $(e_{h_j}, \tilde{\omega}_j)^T \in \mathcal{G}_0$ ,  $j = 1, 2, 3$ . Furthermore, since  $e_{h_1}(t)$  is bounded, combining (25) and (27), it can be inferred that  $e_p(t)$  is bounded. According to the properties of the barrier function, if  $e_p(t)$  is bounded and the initial state  $\omega$  satisfies  $-\underline{p}(0) < e_1(0) < \bar{p}(0)$ , then there must be two time-varying functions  $F_u(t)$  and  $F_d(t)$  that satisfy  $-\underline{p}(t) < F_d(t) \leq e_1(t) \leq F_u(t) < \bar{p}(t)$ .

Parameter perturbations and external disturbances may drive the error trajectory close to the constraint boundary. If the setting of  $\lambda_3$  ensures  $v_x(t) \leq \bar{u}_x$ ,  $x \in \{q, d\}$ , then  $|f(v_x)| = |d(v_x)| \leq D_x$ . If  $\lambda_3$  allows  $v_x$  to exceed  $\bar{u}_x$ ,  $|f(v_x)|$  may temporarily surpass  $D_x$ . However, due to the self-adjustment triggering conditions of FADPPF,  $|f(v_x)|$  remains bounded. Let  $D_x^v$  denote the bound of  $v_x$  when self-adjustment is triggered, then  $|f(v_x)| \leq \max\{D_x^v, D_x\}$ . Combined with Eq. (53), if parameter perturbations, unmodeled dynamics, or disturbances (e.g., torsional vibrations, sensor noise) exceed the approximation capability of the FNN,  $\chi$  may surpass  $kV$ . Additionally, the  $\lambda_3$  closer to 1 leads to larger  $|f(v_x)|$ , further increasing  $\chi$ . Therefore, the selection of FADPPF parameters (e.g.,  $\lambda_3$ ) and FNN membership function parameters must balance transient/steady-state performance and robustness against perturbations, unmodeled dynamics, and disturbance magnitudes/frequencies in specific applications.

## 5. Simulation results

In this section, three simulations are provided to demonstrate the effectiveness and superiority of the proposed dynamic prescribed performance fuzzy-neural backstepping control scheme. First, to visually show the advantages of FADPPF, Case 1 compares the simulation results under different PPFs without considering SF. Then, Case 2 provides a comparison of simulation results for control schemes with and without SF. Finally, to further demonstrate the superior performance of the PMSM under the combined action of SF and FADPPF, Case 3 provides simulation results under different PMSM parameters.

To make the description more concise, FAD, S, A, FP, and DE are used as subscripts or superscripts to denote simulation results based on FADPPF, SPPF [37], APPF [27], FPPPF [30], and DEPPF [38], respectively. For example, the tracking error obtained from FADPPF-based simulation is expressed as  $e_{1-FAD}$ .

**Case 1: A comparison of prescribed performance fuzzy-neural backstepping control schemes based on FADPPF, SPPF, APPF, FPPPF, and DEPPF without SF is presented.**

The relevant parameters of the PMSM model are  $L = 2.95\text{mH}$ ,  $J = 0.04457\text{kg}\cdot\text{m}^2$ ,  $b = 0.005\text{N}\cdot\text{ms}/\text{rad}$ ,  $n_p = 5$ ,  $\psi = 91.45\text{mWb}$ ,  $R_0 = 0.59\Omega$ . The voltage of the DC-link is  $U_{dc} = 200\text{V}$ .  $\bar{u}_q$  and  $\bar{u}_d$  are determined as  $\bar{u}_q = 0.99U_{dc}/\sqrt{3}\text{V}$  and  $\bar{u}_d = 0.1U_{dc}/\sqrt{3}\text{V}$ . The reference trajectory of  $\omega$  is  $\omega_r = 10 \sin(2t) - 6 \cos(2t) + 25$ .

In FNN, the centers and widths of MFs for  $\omega$ ,  $i_q$ ,  $i_d$ ,  $e_1$ ,  $e_2$  and  $e_3$  are  $[-80 : 1.6 : 80]$ ,  $[-13 : 0.26 : 13]$ ,  $[-5 : 0.1 : 5]$ ,  $[-160 : 3.2 : 160]$ ,  $[-26 : 0.52 : 26]$ ,  $[-10 : 0.2 : 10]$ , 0.6, 0.1, 0.03, 1, 0.2 and 0.06, respectively. The initial values of  $\omega$ ,  $i_q$ ,  $i_d$  and FNN weights  $\omega_j$ ,  $j = 1, 2, 3$  are all 0. The parameters of the controller are  $c_j = 2$ ,  $v_j = 6$  and  $b_j = 0.001$ ,  $j = 1, 2, 3$ . The parameters of FADPPF are designed as  $\lambda_0 = 25$ ,  $\lambda_\infty = 0.6$ ,  $\bar{\lambda}_\infty = 0.3$ ,  $\underline{\lambda}_\infty = 0.5$ ,  $t_0 = 0.5$ ,  $a_1 = 2$ ,  $a_2 = 1$ ,  $a_3 = 1$ ,  $\lambda_1 = 1$ ,  $\lambda_2 = 10$ ,  $\lambda_3 = 0.9$ ,  $\lambda_4 = 1.5$  and  $\lambda_5 = 0.4$ .

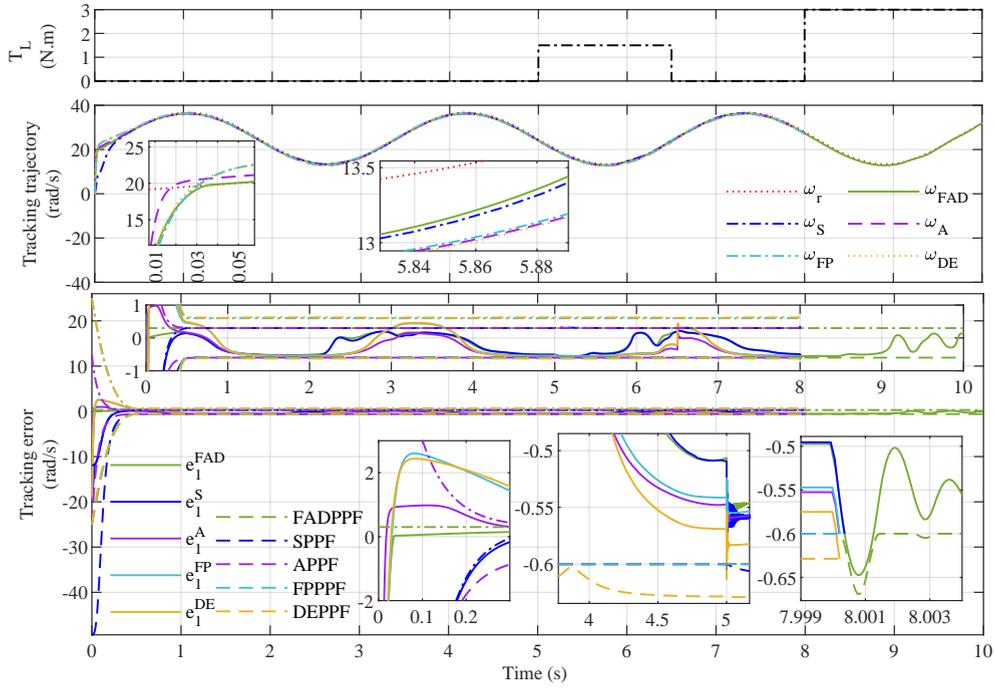
The APPF in [27] is  $\mu_1 = (\mu_{1,0} - \mu_{1,\infty}) \exp(-a_1 t) + \mu_{1,\infty}$ , and  $-\delta_{1,\min} \mu_1 < e_1(t) < \delta_{1,\max} \mu_1$ . The parameters of APPF are selected as  $\mu_{1,0} = 25$ ,  $\mu_{1,\infty} = 0.6$ ,  $a_1 = 15$ ,  $\delta_{1,\min} = 1$  and  $\delta_{1,\max} = 0.5$ .

The FPPPF in [30] is  $\rho(t) = (\rho_0^{a_1} - a_1 a_2 t)^{\frac{1}{a_1}} + \rho_\infty$  if  $0 \leq t < t_f$ , and  $\rho(t) = \rho_\infty$  otherwise,  $-\rho(t) < e_1(t) < \rho(t)$ . The parameters of FPPPF are selected as  $\rho_0 = 24.4$ ,  $\rho_\infty = 0.6$ ,  $t_f = 0.5$ ,  $a_1 = 0.4$  and  $a_2 = \rho_0^{a_1} / (a_1 t_f)$ .

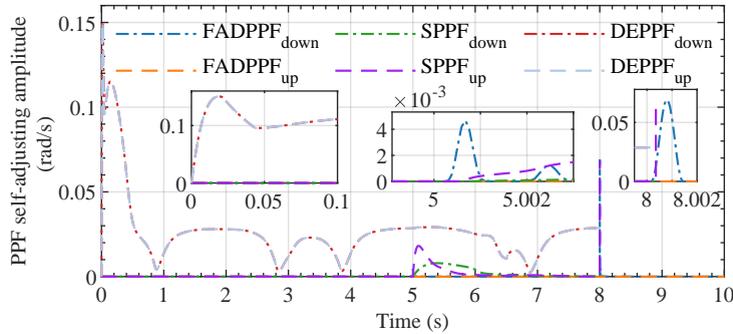
The SPPF from [37] is  $e_l = \rho_{t_f} \text{sign}(e_1(0)) + [\delta_l - \text{sign}(e_1(0))] \rho_t$  and  $e_u = -\rho_{t_f} \text{sign}(e_1(0)) + [\delta_u + \text{sign}(e_1(0))] \rho_t$ , wherein  $\rho_t = (\rho_0 - \rho_{t_f}) \exp(-\varpi) + \rho_{t_f}$  if  $0 \leq t \leq t_f$ , and  $\rho_t = \rho_{t_f}$  otherwise, wherein  $\varpi = kt_f t / (t_f - t)$ . And  $-E_l < e_1(t) < E_u$ , wherein  $E_l = e_l + \gamma_l$  and  $E_u = e_u + \gamma_u$ . The parameters of SPPF are selected as  $\rho_0 = 25$ ,  $\rho_{t_f} = 0.6$ ,  $k = 500$ ,  $t_f = 0.5$ ,  $\bar{\eta}_l = 5$ ,  $\bar{\eta}_u = 5$ ,  $\delta_l = 1$  and  $\delta_u = 0.5$ . The transformation errors  $z_1$ ,  $z_2$  and  $z_3$  in [22] include auxiliary system state variables, which are different from  $e_p$ ,  $e_2$  and  $e_3$  in this paper.  $v_{q-FT}$  and  $v_{d-FT}$  based on SPPF are:  $v_{q-FT} = -\hat{\theta}_{2-FT}^T \gamma_2 - c_2 z_2 - z_2/2 - \text{sign}(\eta_{2,u} - \eta_{2,l}) \Gamma_2 (\eta_{2,u} - \eta_{2,l}) / \Gamma_t$ ,  $v_{d-FT} = -\hat{\theta}_{3-FT}^T \gamma_3 - c_3 z_3 - z_3/2 - \text{sign}(\eta_{3,u} - \eta_{3,l}) \Gamma_3 (\eta_{3,u} - \eta_{3,l}) / \Gamma_t$ , wherein  $\eta_{j,u}$  and  $\eta_{j,l}$  ( $j = 1, 2, 3$ ) are the state variables of the auxiliary system. The parameters of the auxiliary system are chosen as  $\Gamma_1 = 1$ ,  $\Gamma_2 = 5$  and  $\Gamma_3 = 5$ .

The DEPPF in [38] is  $\mu(t) = \frac{\mu_0 - \mu_\infty}{(1 - e^{-a_1 t})^2} (e^{-a_1 t} - e^{-a_1 t_f})^2 + \mu_\infty$  if  $t \leq t_f$ , and  $\mu(t) = \mu_\infty$  otherwise. The auxiliary system for dynamic adjustment of DEPPF is  $\dot{x}(t) = -a_2 x(t) + |e(t)|$  if  $\delta < \Lambda$ , and  $\dot{x}(t) = -a_2 x(t)$  otherwise, wherein

$x(0) = 0$ ,  $\delta = \min \{|\mu(t) - e(t)|, |\mu(t) + e(t)|\}$ , and  $\Lambda = \frac{\mu(t)}{a_3}$ . Then  $-\mu(t) - x(t) < e_1(t) < \mu(t) + x(t)$ . The parameters of FPPPF are selected as  $\mu_0 = 25$ ,  $\mu_\infty = 0.6$ ,  $t_f = 0.5$ ,  $a_1 = 1.5$ ,  $a_2 = 20$  and  $a_3 = 1.1$ .

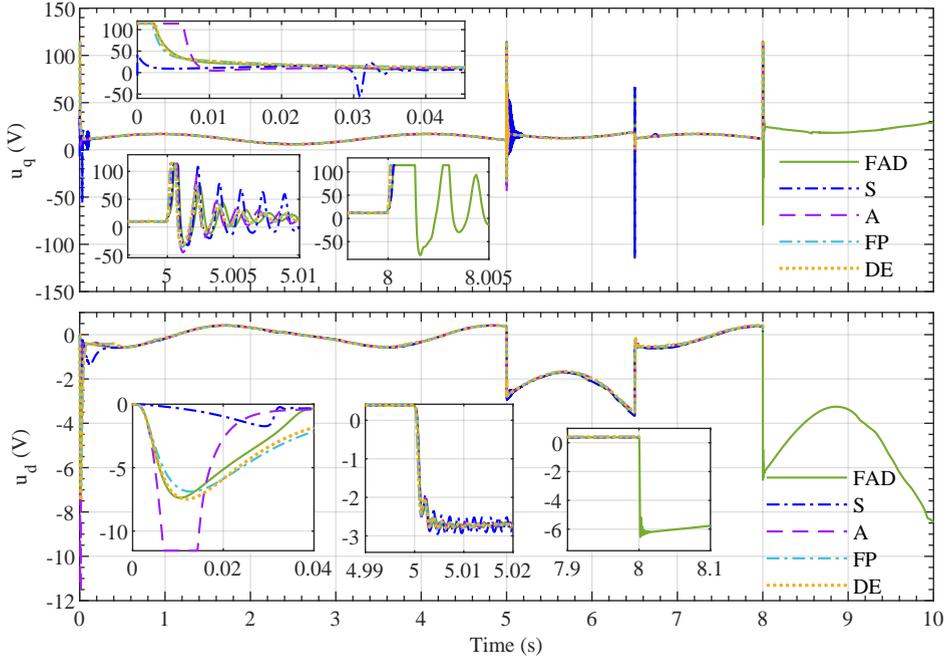


**Fig. 4.** The load torque trajectory, tracking trajectories, tracking errors and corresponding FADPPFs of Case 1.



**Fig. 5.** The PPF self-adjustment amplitude.

Fig. 4 shows the load torque  $T_L(t)$ , tracking trajectories, tracking errors and corresponding PPFs. Fig. 5 is the self-adjusting amplitudes of FADPPF and SPPF. Fig. 6 shows the control signals. Although the five basic constraint functions are set with identical initial and asymptotic values, there is a significant difference in the values of the five PPFs at  $t = 0s$ , and the slopes of each PPF as time progresses are different. Therefore,  $\omega_{FAD}$ ,  $\omega_S$ ,  $\omega_A$ ,  $\omega_{FP}$  and  $\omega_{DE}$  have different transient behaviors in  $[0.5, 5)s$ . As shown in Fig. 4, the APPF, FPPPF, and DEPPF exhibit larger overshoot tolerance, and significant error overshoot occurs under their constraints. The overshoot of  $\omega_{FAD}$  is approximately 0.03 rad/s,  $\omega_S$  shows negligible overshoot (0 rad/s), while  $\omega_A$ ,  $\omega_{FP}$ , and  $\omega_{DE}$  demonstrate overshoots of approximately 1 rad/s, 2.6 rad/s, and 2.45 rad/s, respectively. The overshoot of  $\omega_A$  is about 33 times that under FADPPF-based control. The overshoot of  $\omega_{FP}$  is about 86 times that under FADPPF-based control. The overshoot of  $\omega_{DE}$  is about 81 times that under FADPPF-based control.



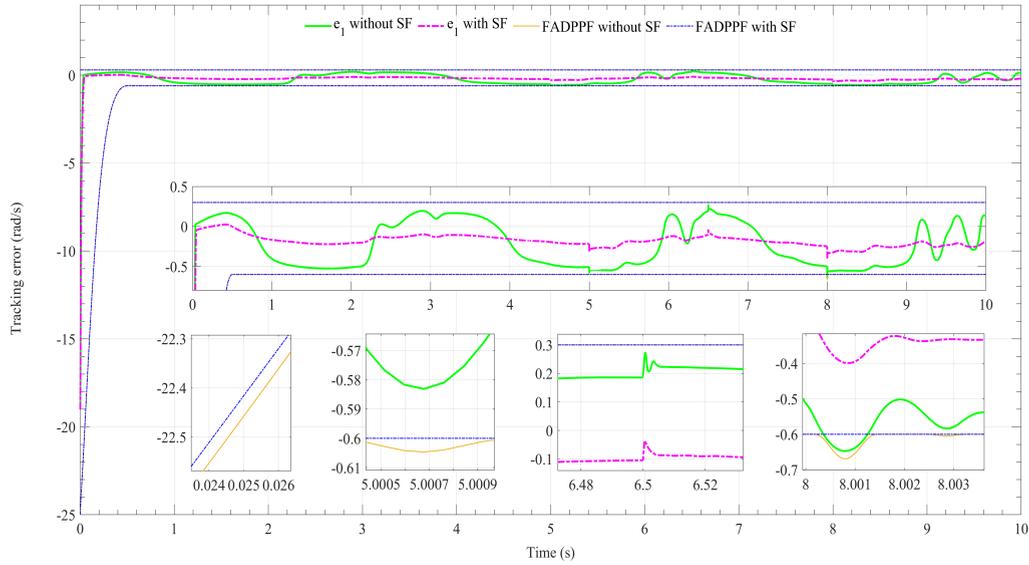
**Fig. 6.** The control signal trajectories of Case 1.

During  $[0.03, 0.3]$  s, the trajectory of  $e_1^S$  closely follows the upper envelope of SPPF, and the design of the upper envelope of SPPF forces a decrease in the transient performance of the system.  $\omega_S$  has the longest rise time under SPPF-based control, which is approximately 9 times that under FADPPF-based control and 19 times that under APPF-based control. In  $t \in [0.5, 5]$  s, the tracking errors under different PPF controls all meet the constraints, and  $e_1 \in [-0.6, 0.3]$  rad/s.

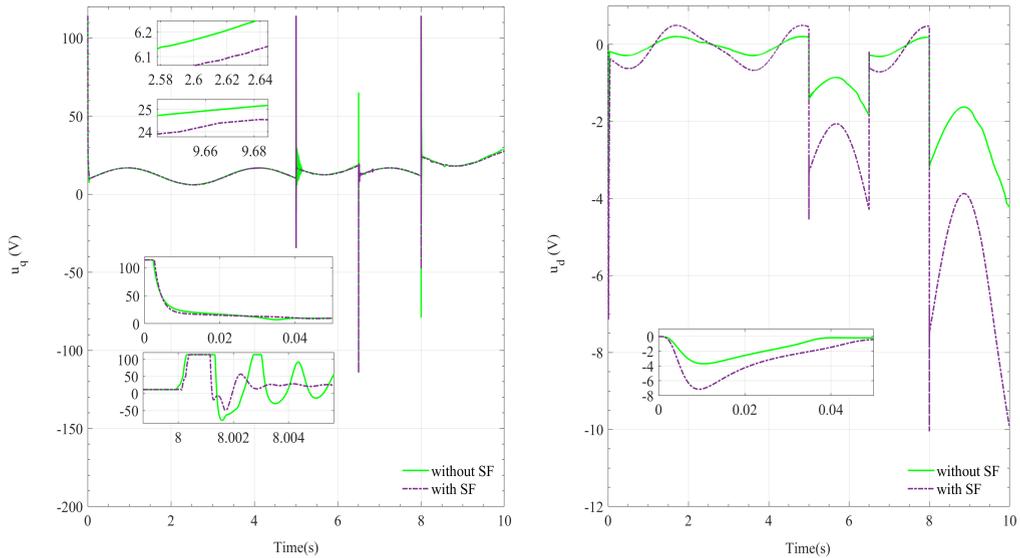
$T_L(t)$  undergoes a step change at  $t = 5$  s. In the FADPPF-based simulation, due to  $0 < \lambda_1 p(t)/e(t) \leq 1$ , the self-adjustment of  $\bar{p}(t)$  is triggered. The lower envelope of FADPPF undergoes self adjustment, while the upper envelope remains unchanged. In  $t \in [5, 6.5]$  s,  $\max(|e_1^{FAD}(t)|) = 0.583113$  rad/s and  $\max(|p_{\Delta 1}(t)|) = 0.004677$  at  $t = 5.00068$  s. The asymmetry of FADPPF is not only reflected in the basic constraint, but also in the self-adjusting additional constraint. When using the SPPF,  $\max(|e_1^S(t)|) = 0.583877$  rad/s at  $t = 5.00062$  s. At this point, the self-adjustment amplitude of SPPF is almost 0. As shown in Fig. 5, the maximum self-adjustment amplitude of the upper envelope is 0.0181, which occurs at  $t = 5.09098$  s, and the maximum self-adjustment amplitude of the lower envelope is 0.00792, which occurs at  $t = 5.39076$  s, exposing the problem of lagged self-adjustment and reduced the convergence speed of tracking error.

At  $t = 8$  s,  $T_L(t)$  changes from 0 to 3. Due to the lack of self-adjustment capability in the APPF and FPPPF,  $e_1^A(t)$  asymptotically approaches the APPF boundary at  $t = 8.00019$  s, triggering system instability, while  $e_1^{FP}(t)$  converges infinitely close to the FPPPF constraint at  $t = 8.0002$  s, also leading to instability. For SPPF, the delayed self-adjustment causes  $e_1^S(t)$  to exceed the range limited by SPPF at  $t = 8.00019$  s, resulting in system instability. As indicated by Fig. 7, under the self-adjustment mechanism of DEPPF, the dynamic additional term of DEPPF is always greater than 0 throughout the simulation. The constraint envelope is relaxed even without the need for dynamic adjustment. But the envelope self-adjustment exhibits a delayed response to the load step change from 0 to 3. Consequently, due to the lagged self-adjustment of DEPPF,  $e_1^{DE}(t)$  violates the DEPPF-restricted range at  $t = 8.0002$  s, inducing control singularity. In the simulation based on FADPPF, when  $t \in [8, 10]$  s, the step load causes  $\max(|e_{1-FAD}(t)|) = 0.64734$  rad/s at  $t = 8.00076$  s. The self-adjustment of  $\bar{p}(t)$  is triggered,  $\max(|p_{\Delta 1}(t)|) = 0.0690698$  occurs at  $t = 8.00076$  s.  $e_{1-FAD}(t)$  consistently meets the constraints, and its transient and steady-state performance meets the expected requirements.

**Case 2:** The control schemes with and without SF are compared. The parameters of SF are  $k_1 = 0.2$  and  $k_2 = 2$ . Other simulation parameters are the same as in Case 1.



**Fig. 7.** The tracking errors and corresponding FADPPFs of Case 2.

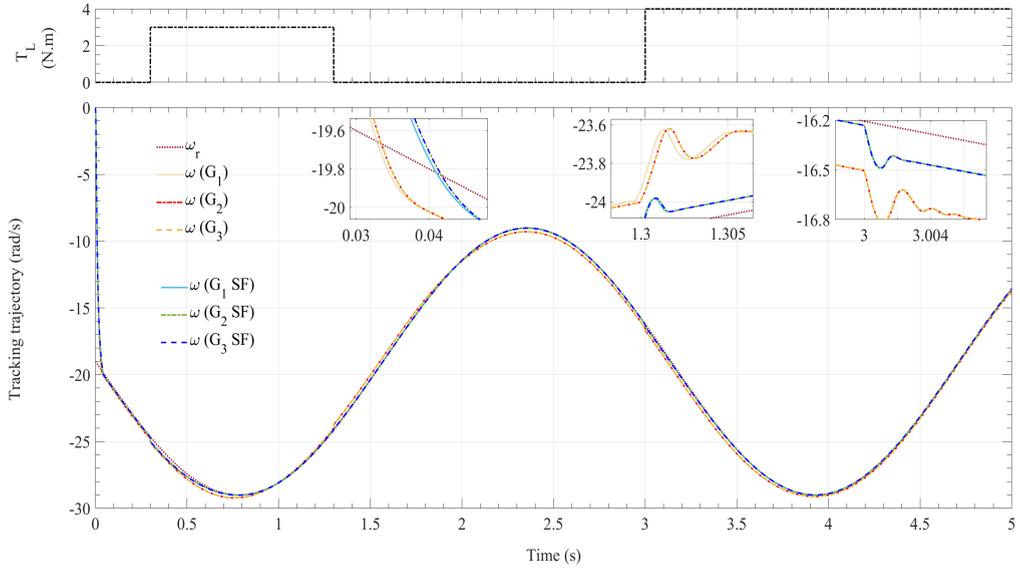


**Fig. 8.** The control signal trajectories of Case 2.

Fig. 7 shows the tracking errors and the corresponding FADPPFs with and without SF. Fig. 8 shows the control signals trajectories of Case 2. From Fig. 7, it is evident that the FADPPF exhibits a semi-funnel shape with the opening facing downwards, and the slope of the opening will self-adjust according to  $e_1(t)$ . Within the specified time  $t_0 = 0.5s$ ,  $e_1(t)$  reaches the expected steady-state requirement. At  $t = 5s$ ,  $t = 6.5s$  and  $t = 8s$ , the load undergoes sudden changes. In the control scheme without SF, the self-adjustments of  $p(t)$  are triggered at  $t = 5s$  and  $t = 8s$ . Based on  $p_{\Delta 1}(t)$ , the FADPPF of control scheme without SF self-adjusts quickly, exhibiting non-monotonic characteristics. Under the control scheme containing SF, the tracking error converges to an interval smaller than the expected steady-state error

interval, so self-adjustment is not triggered when  $T_L(t)$  suddenly changes. In both cases, the tracking errors are kept within the constraints formed by FADPPF. However, under the control scheme containing SF, the PMSM system has smaller overshoot, and exhibits faster error convergence, smaller steady-state error and stronger disturbance rejection.

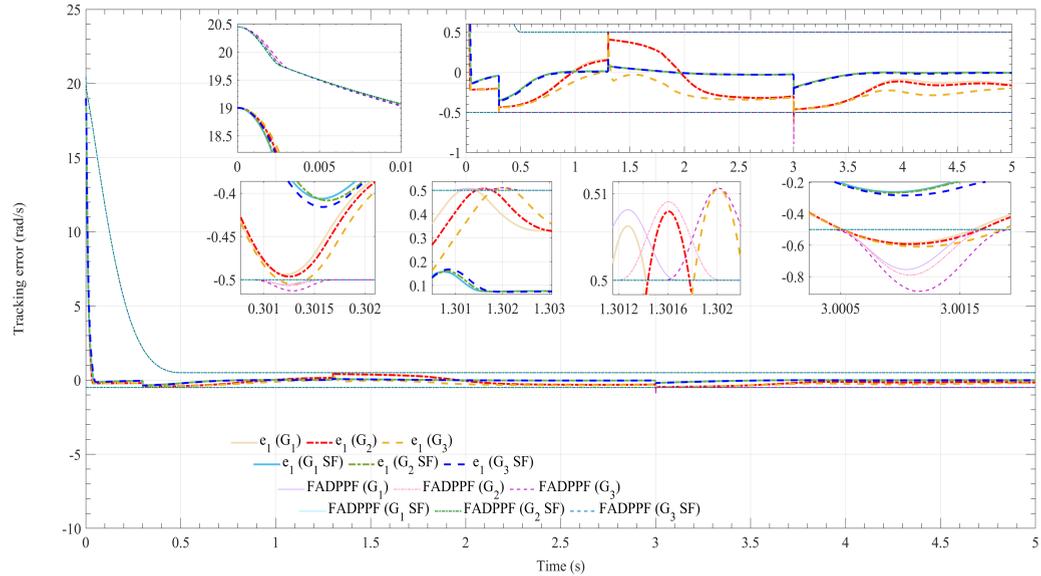
**Case 3:** Case 3 presents a simulation comparison of the PMSM under different parameters. According to [40] and [41], considering the influence of temperature on  $R_0$ ,  $\psi$  and  $L$ , the following three groups of motor parameters are selected for comparison: the first group,  $R_0 = 0.5659\Omega$ ,  $\psi = 92.6\text{mWb}$  and  $L = 2.945\text{mH}$ . Group 2,  $R_0 = 0.59\Omega$ ,  $\psi = 91.45\text{mWb}$  and  $L = 2.95\text{mH}$ . The third group,  $R_0 = 0.7026\Omega$ ,  $\psi = 84.3\text{mWb}$  and  $L = 2.97\text{mH}$ . To further verify the switching mechanism and self adjustment mechanism of FADPPF, the reference trajectory in Case 3 is set as  $\omega_r = -10 \sin(2t) - 19$ , and the parameters of FADPPF are:  $\lambda_0 = 20$ ,  $\lambda_\infty = 0.5$ ,  $a_3 = 1.3$ ,  $\lambda_4 = 1.3$  and  $\lambda_5 = 1.5$ . The other simulation parameters are the same as in Case 2.



**Fig. 9.** The load torque trajectory and tracking trajectories of Case 3.

Fig. 9 shows  $T_L(t)$  and tracking trajectories. Fig. 10 shows the tracking error and corresponding FADPPF. Among them,  $G_i, i = 1, 2, 3$  in the legend indicate that the result is based on the  $i$ th motor parameters, while SF in the legend indicates that the result is based on the controller containing the SF. When  $e_1(0) > 0$ , FADPPF is a half-funnel shape with an upward opening, the upper and lower constraint envelopes are asymmetrical, as shown in Fig. 10. Different  $e_1(0)$  results in different values of  $a_0$  in FADPPF, so the opening direction of FADPPF in Case 3 is different from that in Case 1-2. And  $0 < \lambda_3 p(0) / e(0) \leq 1$ , the self-adjustment of  $\bar{p}(t)$  is triggered at  $t = 0\text{s}$ ,  $p_{\Delta 1}(0) > 0$ . Thus the starting point of  $\bar{p}(t)$  of all FADPPFs in Case 3 is greater than the designed  $\lambda_0 = 20$  in Fig. 10. At the same time, the slopes of each  $\bar{p}(t)$  are time-varying and different from each other. This is because the real-time slope of FADPPF is related to the tracking error, and the greater  $e_1(t)$ , the greater the slope when  $t \leq t_0$ . Therefore, even if the design parameters of FADPPF are the same, the slope of each  $\bar{p}(t)$  is different for different operating conditions, and the transient constraint envelope can be adaptively adjusted according to the actual working scenario. This proves its superior applicability in practical engineering implementation.

The load torque  $T_L(t)$  undergoes step change at  $t = 0.3\text{s}$ ,  $t = 1.3\text{s}$  and  $t = 3\text{s}$ , the  $p_{\Delta 1}(t)$  or  $p_{\Delta 2}(t)$  of FADPPF quickly self-adjusts in the three simulations without SF. Even if the parameters of PMSM model are different,  $e_1(t)$  is constantly within the bounds formed by FADPPF, and its transient response and steady-state error meet the predetermined requirements. When using SF and FADPPF, tracking errors converges to a smaller range. For example, in  $t \in [4, 5]\text{s}$ , under the joint action of SF and PPF, the tracking errors of three groups of PMSM model parameters are all within  $[-0.02, 0]\text{rad/s}$ , while in the simulation without SF, the tracking errors are within  $[-0.3, -0.08]\text{rad/s}$ .



**Fig. 10.** The tracking errors and corresponding FADPPFs of Case 3.

## 6. Conclusion

A new FADPPF is proposed to address the issues of control singularity and system instability in existing prescribed performance control strategies. Based on the FADPPF, a dynamic prescribed performance fuzzy-neural backstepping control approach is proposed for PMSM. The simulation results show that FADPPF can perform self-adjustment quickly based on the real-time tracking error, effectively avoiding control singularity and system instability. In addition, in the case of constrained inputs, time-varying model parameters, and step load, the closed-loop PMSM system has superior transient and steady state performance. **Future work will focus on integrating PPC with energy-optimal control for electric vehicles to ensure transient performance of PMSM-driven electric vehicles while minimizing energy consumption, thereby harmonizing energy efficiency, thermal management, and dynamic responsiveness across diverse driving cycles.**

## References

- [1] Q. C. Yang, F. K. Zhang, Q. H. Sun, and C. Wang. Dynamic learning from adaptive neural control for full-state constrained strict-feedback nonlinear systems. *Neural Networks*, 170:596–609, 2024.
- [2] Z. C. Wang, X. Wang, and N. Pang. Dynamic event-triggered controller design for nonlinear systems: Reinforcement learning strategy. *Neural Networks*, 163:341–353, 2023.
- [3] R. E. Precup, A. T. Nguyen, and S. Blazic. A survey on fuzzy control for mechatronics applications. *Int. J. Syst. Sci.*, 55(4):771–813, 2024.
- [4] A. Bounemour, M. Chemachema, A. Zahaf, and S. Bououden. Adaptive fuzzy fault-tolerant control using nussbaum gain for a class of siso nonlinear systems with unknown directions. In *Proc. 4th Int. Conf. Electr. Eng. Control Appl. (ICEECA 2019)*, volume 682 of *Lect. Notes Electr. Eng.*, Singapore, 2021. Springer.
- [5] A. Bounemour and M. Chemachema. Adaptive fuzzy fault-tolerant control using nussbaum-type function with state-dependent actuator failures. *Neural Comput. Appl.*, 33:191–208, 2021.
- [6] P. Chen, H. Gan, Y. Liu, and Y. Luo. Active disturbance rejection fractional-order independent control of time delay systems with application to air floating motion platform. *IEEE Trans. Instrum. Meas.*, 73:1–10, 2024.
- [7] Y. Qin, D. Zhao, W. Zhang, and Y. Deng. Improved active disturbance rejection control and parameter setting for position tracking of active suspension electro-hydraulic servo actuator. *IEEE/ASME Trans. Mechatron.*, 2024.
- [8] B. Li, T. Guan, X. Guan, K. Zhang, and K.-F. C. Yiu. Optimal fixed-time sliding mode control for spacecraft constrained reorientation. *IEEE Trans. Autom. Control*, 69(4):2676–2683, 2024.
- [9] J. Xia and H. Ouyang. Chattering free sliding-mode controller design for underactuated tower cranes with uncertain disturbance. *IEEE Trans. Ind. Electron.*, 71(5):4963–4975, 2024.
- [10] G. Li, J. Yu, and X. Chen. Adaptive fuzzy neural network command filtered impedance control of constrained robotic manipulators with disturbance observer. *IEEE Trans. Neural Netw. Learn. Syst.*, 34(8):5171–5180, 2023.

- [11] X. Hu, G. Zhu, Y. Ma, Z. Li, R. Malekian, and M. Á. Sotelo. Event-triggered adaptive fuzzy setpoint regulation of surface vessels with unmeasured velocities under thruster saturation constraints. *IEEE Trans. Intell. Transp. Syst.*, 23(8):13463–13472, 2022.
- [12] J. Sun, J. Yi, and Z. Pu. Fixed-time adaptive fuzzy control for uncertain nonstrict-feedback systems with time-varying constraints and input saturations. *IEEE Trans. Fuzzy Syst.*, 30(4):1114–1128, 2022.
- [13] N. Han, X. Ren, C. Zhang, and D. Zheng. Prescribed performance control with input indicator for robot system based on spectral normalized neural networks. *Neurocomputing*, 492:201–210, 2022.
- [14] S. Kang, P. X. Liu, and H. Wang. Fixed-time adaptive fuzzy command filtering control for a class of uncertain nonlinear systems with input saturation and dead zone. *Nonlinear Dyn.*, 110:2401–2414, 2022.
- [15] H. Gong, M. J. Er, Y. Liu, and C. Ma. Three-dimensional optimal trajectory tracking control of underactuated auvs with uncertain dynamics and input saturation. *Ocean Eng.*, 298:116757, 2024.
- [16] R. Xu, G. Tang, D. Xie, L. Han, and H. Huang. Neural network for 3d trajectory tracking control of a cmg-actuated underwater vehicle with input saturation. *ISA Trans.*, 123:152–167, 2022.
- [17] O. Bey and M. Chemachema. Finite-time event-triggered output-feedback adaptive decentralized echo-state network fault-tolerant control for interconnected pure-feedback nonlinear systems with input saturation and external disturbances: A fuzzy control-error approach. *Inf. Sci.*, 669:120557, 2024.
- [18] A. Bounemour and M. Chemachema. Finite-time output-feedback fault tolerant adaptive fuzzy control framework for a class of mimo saturated nonlinear systems. *Int. J. Syst. Sci.*, 56(4):733–752, 2024.
- [19] G. Zhang, C. M. Chew, Y. Xu, and M. Fu. Switching dynamic event-triggered sliding mode based trajectory tracking control for asvs with nonlinear dead-zone and saturation inputs. *IEEE Trans. Intell. Transp. Syst.*, 26(3):4019–4031, 2025.
- [20] X. Song, Z. Fan, S. Lu, Y. Yan, and B. Yue. Predefined-time sliding mode attitude control for liquid-filled spacecraft with large amplitude sloshing. *Eur. J. Control*, 77:100970, 2024.
- [21] A. Ilchmann, E. P. Ryan, and C. J. Sangwin. Tracking with prescribed transient behaviour. *ESAIM Control Optim. Calculus Variat.*, 7:471–493, 2002.
- [22] C. P. Bechlioulis and G. A. Rovithakis. Robust adaptive control of feedback linearizable mimo nonlinear systems with prescribed performance. *IEEE Trans. Autom. Control*, 53(9):2090–2099, 2008.
- [23] L. Zhang, W. W. Che, C. Deng, and Z. G. Wu. Prescribed performance control for multiagent systems via fuzzy adaptive event-triggered strategy. *IEEE Trans. Fuzzy Syst.*, 30(12):5078–5090, 2022.
- [24] Y. Xia, K. Xiao, and Z. Geng. Event-based adaptive fuzzy control for stochastic nonlinear systems with prescribed performance. *Chaos, Solitons Fractals*, 180(114501), 2024.
- [25] P. Yang and Y. Su. Proximate fixed-time prescribed performance tracking control of uncertain robot manipulators. *IEEE ASME Trans. Mechatron.*, 27(5):3275–3285, 2022.
- [26] Y. Xia, K. Xiao, Y. Yao, Z. Geng, and Z. Lendek. Fixed-time fuzzy vibration reduction for stochastic mems gyroscopes with low communication resources. *IEEE Trans. Fuzzy Syst.*, 32(8):4220–4233, 2024.
- [27] Y. Li, X. Shao, and S. Tong. Adaptive fuzzy prescribed performance control of nontriangular structure nonlinear systems. *IEEE Trans. Fuzzy Syst.*, 28(10):2416–2426, 2020.
- [28] Y. Wang, Y. Liu, Y. Wang, and T. Chai. Neural output feedback control of automobile steer-by-wire system with predefined performance and composite learning. *IEEE Trans. Veh. Technol.*, 72(5):5906–5921, 2023.
- [29] J. Tao, T. Zhang, and Q. Liu. Novel finite-time adaptive neural control of flexible spacecraft with actuator constraints and prescribed attitude tracking performance. *Acta Astronaut.*, 179:646–658, 2021.
- [30] Y. Xia, C. Liu, Y. Tuo, and J. Li. Command filter-based event-triggered control for stochastic mems gyroscopes with finite-time prescribed performance. *ISA Trans.*, 148:212–223, 2024.
- [31] X. Bu, X. Wu, F. Zhu, J. Huang, Z. Ma, and R. Zhang. Novel prescribed performance neural control of a flexible air-breathing hypersonic vehicle with unknown initial errors. *ISA Trans.*, 59:149–159, 2015.
- [32] H. Xie, Y. Jing, J.-X. Zhang, and G. M. Dimirovski. Low-complexity tracking control of unknown strict-feedback systems with quantitative performance guarantees. *IEEE Trans. Cybern.*, 54(10):5887–5900, 2024.
- [33] Y. Xia, Z. Lendek, Z. Geng, J. Li, and J. Wang. Adaptive fuzzy control for stochastic nonlinear systems with nonmonotonic prescribed performance and unknown control directions. *IEEE Trans. Syst., Man, Cybern. Syst.*, 2024.
- [34] X. Bu, Q. Qi, and B. Jiang. A simplified finite-time fuzzy neural controller with prescribed performance applied to waverider aircraft. *IEEE Trans. Fuzzy Syst.*, 30(7):2529–2537, 2022.
- [35] Y. Wang and J. Hu. Improved prescribed performance control for air-breathing hypersonic vehicles with unknown deadzone input nonlinearity. *ISA Trans.*, 79:95–107, 2018.
- [36] X. Bu, M. Lv, H. Lei, and J. Cao. Fuzzy neural pseudo control with prescribed performance for waverider vehicles: A fragility-avoidance approach. *IEEE Trans. Cybern.*, 53(8):4986–4999, 2023.
- [37] R. Ji, D. Li, and S. S. Ge. Saturation-tolerant prescribed control for nonlinear time-delay systems. *IEEE Trans. Fuzzy Syst.*, 31(8):2495–2508, 2023.
- [38] X. Bu, R. Luo, M. Lv, and H. Lei. Adaptive fuzzy safety control of hypersonic flight vehicles pursuing adaptable prescribed behaviors: A sensing and adjustment mechanism. *IEEE Trans. Fuzzy Syst.*, 32(12):7050–7062, 2024.
- [39] X. Yang, J. Yu, Q.G. Wang, L. Zhao, H. Yu, and C. Lin. Adaptive fuzzy finite-time command filtered tracking control for permanent magnet synchronous motors. *Neurocomputing*, 337:110–119, 2019.
- [40] G. Feng, C. Lai, J. Tjong, and N. C. Kar. Noninvasive kalman filter based permanent magnet temperature estimation for permanent magnet synchronous machines. *IEEE Trans. Power Electron.*, 33(12):10673–10682, 2018.
- [41] H. S. Jung and D. Y. Oh. Estimation of stator and magnet temperatures of ipmsm from active and reactive energies at medium and high speeds. *IEEE Trans. Transport. Electrific.*, 9(2):2983–2993, 2023.

- [42] G. Cui, J. Yu, and Q. G. Wang. Finite-time adaptive fuzzy control for mimo nonlinear systems with input saturation via improved command-filtered backstepping. *IEEE Trans. Syst., Man, Cybern. Syst.*, 52(2):980–989, 2022.
- [43] X. Bu, B. Jiang, and H. Lei. Low-complexity fuzzy neural control of constrained waverider vehicles via fragility-free prescribed performance approach. *IEEE Trans. Fuzzy Syst.*, 31(7):2127–2139, 2023.
- [44] N. Zhang, J. W. Xia, J. H. Park, J. Zhang, and H. Shen. Improved disturbance observer-based fixed-time adaptive neural network consensus tracking for nonlinear multi-agent systems. *Neural Networks*, 162:490–501, 2023.
- [45] K. Zhao and Y. Song. Neuroadaptive robotic control under time-varying asymmetric motion constraints: A feasibility-condition-free approach. *IEEE Trans. Cybern.*, 50(1):15–24, 2020.
- [46] H. Liang, Y. Zhang, T. Huang, and H. Ma. Prescribed performance cooperative control for multiagent systems with input quantization. *IEEE Trans. Cybern.*, 50(5):1810–1819, 2020.