IEEE Robotics and Automation Letters (RAL) paper presented at
2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Neural-FxSMC: A Robust Adaptive Neural Fixed-time Sliding Mode Control for Quadrotors with Unknown Uncertainties

Subhash Chand Yogi[‡], Laxmidhar Behera[†,§], *Senior Member, IEEE* and Twinkle Tripathy[†]

***Abstract*—This paper presents *Neural-FxSMC*, a robust and precise control scheme for quadrotors to counter unknown dynamics, uncertainties, and external disturbances. *Neural-FxSMC*, (*i*) addresses fixed-time convergence of the tracking error, control singularity, and chattering issues simultaneously, which is not possible with the existing Fixed time Sliding Mode Control (FxSMC), and (*ii*) relaxes the a priori bound assumption over the uncertainties that are often considered as a constant or a state-dependent upper bound. The fixed-time convergence of tracking error is guaranteed by establishing fixed-time convergence of the Non-singular Fast Terminal Sliding Surface (NFTSS), contrary to the existing works where the NFTSS convergence depends on initial conditions. The Chattering is suppressed via Radial Basis Function Network (RBFN) based uncertainties estimation. Finally, using the Lyapunov theory, we prove the fixed-time convergence and boundedness of *Neural-FxSMC* weights. We comprehensively evaluate *Neural-FxSMC* in challenging scenarios such as unknown payload and turbulent wind. Our *Neural-FxSMC*, apart from handling unknown dynamics and uncertainties, also offers direct gravity compensation without using quadrotor mass and gravity.**

***Index Terms*—Robust/Adaptive Control; Aerial Systems: Applications, Fixed-time Convergence, Intelligent Control.**

I. Introduction

SLIDING mode control (SMC) [1]–[3] is a popular quadrotor control scheme due to its strong robustness to matched disturbances. Advanced SMC with finite/fixed-time tracking convergence [2]–[4] infers enhanced control precision and robustness and overcomes peaking effects in asymptotic convergent SMC as discussed in [5]. A fixed-time SMC (FxSMC) [4] provides an additional advantage of estimating convergence/settling time without using initial states. This is particularly useful since initial states are not always known in real-time, and their larger values may lead to unbounded convergence time as in finite-time SMC [2], [3]. In addition, FxSMC has faster convergence than both finite-time SMC [2] and asymptotic convergence SMC [1]. This contributes to enhancing tracking accuracy as each trajectory setpoint converges within a tighter error band before the arrival of the next setpoint. The FxSMC can benefit quadrotor mission planning, enhancing tracking accuracy, task optimization and execution by pre-flight convergence time estimation.

This work was supported by IHFC-IIT Delhi under the project IITM/IHFC-IITDELHI/LB/370 (IIT Mandi India).

[‡] is with IIT Mandi iHub and HCI Foundation, IIT Mandi, India 175005, Email: subhash@ihubiitmandi.ac.in while the work has been carried out at IIT Kanpur.
[†] is with Dept. of Electrical ENgineering, IIT Kanpur, India 208016, Email: ttripathy@iitk.ac.in.
[§] is with School of Computing & Electrical Engineering, IIT Mandi, India. Email: director@iitmandi.ac.in

Video demonstration: https://youtu.be/0dQYSH0gM_g

Despite several FxSMC schemes, the control singularity and chattering issues are still present when both sliding manifold and tracking error are required to converge in a fixed time. Recent FxSMC [10]–[15] address the convergence speed only, i.e., faster convergence of sliding manifold but suffers from controller singularity. The singularity may generate high control input and should be handled carefully. Although singularity is tackled via piecewise function in [10]–[14], it results in asymptotic or theoretically infinite time convergence near sliding surface, which is not a truly fixed-time approach. In contrast, [16] infers fixed-time convergence of sliding manifold without singularity; however, it has asymptotic tracking error convergence. Authors in [17] achieve non-singularity via integral FxSMC via bilimit homogeneity; however, the same can not be used for convergence time estimation. Direct singularity-free [18], [19] and FxSMC [20] do not use any piecewise function to tackle singularity. However, robustness is compromised, and estimated convergence time is ineffective due to its dependency on states that are not known in advance.

Recently, authors in [21] address the control singularity with fixed-time convergence and offer robustness via pure switching action near equilibrium but with high chattering. The chattering poses a critical issue for highly nonlinear and open-loop unstable systems, e.g., quadrotors. It can trigger unmodelled dynamics, leading to excessive power fluctuations in the motors in steady state, thus affecting the motor performance and degrading battery life. Moreover, it can also hamper vision-based localization (Visual SLAM and VIO), which is quite sensitive to high-frequency vibrations. Prolonged vibrations can trigger state estimation divergence, leading to quadrotor crashes, regardless of controller effectiveness. Although chattering can be reduced with higher order SMC and Boundary Layer Approximation (BLA) [22], they deteriorate the tracking performance near sliding surface.

Furthermore, other critical issues include the lack of exact system dynamics, uncertainties, and external disturbances. These are often treated as lumped uncertainties. The adaptive approaches or disturbance observer [23] assume uncertainties bounded by either constant value [6], [24] or state-dependent bound with constant coefficients [1], [7], [20]. These constants are updated to trigger the switching action to counter uncertainties appropriately. More recently, [25] proposes an extreme learning machine for upper bound estimation. However, the upper bound estimation may still lead to overestimating the control effort, leading to chattering. While authors in [26] propose a switching function approximation scheme for robotic manipulator to tune the switching gain of conventional SMC via a neural network, however, only asymptotic convergence

is guaranteed. Other neural network-based methods are employed to estimate unmodelled dynamics/uncertainties in [13], [19], [27]. The controller uses this estimate to cancel out the uncertainties, whereas switching gain is set by trial and error.

Given the above discussion, handling control singularity, chattering, and fixed-time convergence of the tracking error simultaneously, without assuming any bound on uncertainties while dealing with completely unknown quadrotor dynamics, is a state-of-the-art control problem. This paper is the first to address this objective by making the following contributions:

1) Fast Fixed-time Reaching Law: We propose a new fixed-time reaching law that exhibits faster convergence than recent fixed-time methods [14], [21].

2) Fixed-time Convergence of both Tracking Error and Sliding Surface: We achieve this by guaranteeing fixed-time convergence of Non-singular Fast Terminal Sliding Surface (NFTSS) [29], which had finite-time convergence only. We obtain a simple convergence time expression for NFTSS.

3) Neural-FxSMC development: We propose a control framework namely *Neural-FxSMC* for quadrotors with completely unknown dynamics and uncertainties without assuming these a priori bounded unlike [1], [6], [7], [20], [24].

4) Simultaneously Addressing Singularity and Chattering: *Neural-FxSMC* handles these issues at once. Singularity is tackled by NFTSS, and chattering is suppressed by adjusting the switching gain based on an RBFN-based uncertainty estimate. This contrasts to [13], [19], [27], which eliminates the uncertainty estimate from the controller expression.

5) Rigorous Experimentation: The proposed scheme is comprehensively evaluated under various uncertainties, i.e., unknown payload transport and high wind turbulence.

Next, Sec. II discusses the proposed control scheme and *Neural-FxSMC* followed by *Neural-FxSMC* development for quadrotor in Sec. III. Simulations in Sec. IV are followed by rigorous experiments in Sec. V and conclusion in Sec. VI.

II. Proposed Control Scheme

The quadrotor dynamics resembles a second-order dynamic system [27]. Hence, we start with second-order dynamics with disturbance d and states $y, \dot{y} \in \mathbb{R}$ as follows:

$$\ddot{y} = f_y + g_y u + d \tag{1}$$

where $f_y, g_y \in \mathbb{R}$. The formulation of sliding mode controller consists of a sliding surface and reaching law. We first introduce a fixed-time convergent sliding surface and a novel fast reaching law, and then *Neural-FxSMC* is developed. We also compare with recent methods while showing how it avoids the pitfalls of singularity [14] and chattering [21].

A. Proposed Sliding Surface

For a second order system (1), a Non-singular Fast Terminal Sliding Surface (NFTSS) [29] can be written as follows:

$$s = y + k_1 |y|^{r_1} sign(y) + k_2 |\dot{y}|^{r_2} sign(\dot{y}) \tag{2}$$

where $1 < r_2 < r_1 < 2$, and $k_1, k_2 > 0$. We establish fixed-time convergence of y in contrast to [29], where convergence time depends on initial conditions and has a complex expression.

When sliding surface is reached i.e. $s = 0$, we rewrite (2):

$$k_2 |\dot{y}|^{r_2} sign(\dot{y}) = -(|y| + k_1 |y|^{r_1}) sign(y)$$

This follows $sign(\dot{y}) = -sign(y)$ and $k_2 |\dot{y}|^{r_2} = (|y| + k_1 |y|^{r_1})$ or $|\dot{y}| = \left[\frac{1}{k_2}(|y| + k_1 |y|^{r_1})\right]^{\frac{1}{r_2}}$. Expression of $\dot{y}$ is $\dot{y} = |\dot{y}| sign(\dot{y})$. Using both of the last equalities, the dynamics of y becomes,

$$\dot{y} = -\left(\frac{1}{k_2}(|y| + k_1 |y|^{r_1})\right)^{\frac{1}{r_2}} sign(y) \tag{3}$$

Consider a Lyapunov candidate $V = \frac{1}{2} y^2$ to prove the fixed-time convergence. Its time derivative $\dot{V} = y\dot{y}$ becomes,

$$\begin{aligned} \dot{V} &= -\left(\frac{1}{k_2}|y|^{r_2+1} + \frac{k_1}{k_2}|y|^{r_1+r_2}\right)^{\frac{1}{r_2}} \\ &= -\left(\frac{1}{k_2} 2^{\frac{r_2+1}{2}} V^{\frac{r_2+1}{2}} + \frac{k_1}{k_2} 2^{\frac{r_1+r_2}{2}} V^{\frac{r_1+r_2}{2}}\right)^{\frac{1}{r_2}} \end{aligned} \tag{4}$$

If $1 < r_2 < r_1 < 2$, then $\frac{r_2+1}{2r_2} < 1$ and $\frac{r_1+r_2}{2r_2} > 1$. Now, we prove the fixed-time convergence in two steps as follows. *First*, if $|V| \geq 1$ then $\dot{V} \leq -\eta_2 V^{\frac{r_1+r_2}{2r_2}}$ for $\eta_2 = (\frac{1}{k_2} 2^{\frac{r_1+r_2}{2}})^{\frac{1}{r_2}}$. Thus $y(t_0)$ will reach to $|y| = 1$ within the time t_1 as $t_1 \leq \frac{2r_2}{\eta_2(r_1 - r_2)}$. *Second*, $|V| \leq 1$ implies $\dot{V} \leq -\eta_1 V^{\frac{r_2+1}{2r_2}}$ for $\eta_1 = (\frac{k_1}{k_2} 2^{\frac{r_2+1}{2}})^{\frac{1}{r_2}}$. Hence $|y| = 1$ will reach to $y = 0$ within time $t_2 \leq \frac{2r_2}{\eta_2(r_2 - 1)}$. We can infer that the tracking error y converges to zero from an initial state $y(t_0)$ within time $t_y = t_1 + t_2$:

$$t_y \leq \frac{2r_2}{\eta_2(r_1 - r_2)} + \frac{2r_2}{\eta_1(r_2 - 1)} \tag{5}$$

We see that t_y is free from $y(t_0)$, which implies fixed-time convergence of y. Also, t_y has a simpler expression than [29] and can be estimated with only four parameters r_1, r_2, k_1, k_2.

B. Proposed Fast Reaching Law

We propose a novel fast fixed-time convergent reaching law:

$$\dot{s} = -k |s|^{m_1 \tanh(\alpha^{f(s)} |s|^2)} sign(s) \tag{6}$$

where $f(s) = \tanh(\beta(|s| - 1))$, $k > 0$, $\alpha, \beta >> 1$, $m_1 > \tanh(1)^{-1}$. The reaching law (6) drives sliding surface to $s = 0$ within time t_s from any initial s. t_s is estimated apriori as:

$$t_s \leq \frac{\sqrt{2^{1 - m_1 \tanh(1)}}}{k(m_1 \tanh(1) - 1)} + \frac{\sqrt{2}}{k\ell} \tag{7}$$

where $\ell = min(|s|^{m_1 \tanh(\alpha^{f(s)} |s|^2)}) \neq 0$. Only two tuning parameters m_1 and k are needed to estimate t_s as α, β are selected large enough. Also, t_s does not have initial s, stating fixed-time convergent s. The proof is given in Appendix A.

In (6), the exponent of $|s|$ is responsible for the convergence speed of s. We choose a novel exponent $\vartheta(s) = m_1 \tanh(\alpha^{f(s)} |s|^2)$ that has a non-decreasing value $\approx m_1$ for $|s| > 1$ and becomes zero for $|s| < 1$. This implies faster convergent s as follows: for $|s| > 1$, $\tanh(\alpha^{f(s)} |s|^2) \approx 1$ as $\alpha, \beta >> 1$ and hence, the exponent becomes $\vartheta \approx m_1$ for entire $|s| > 1$ unlike decreasing ϑ in [21]. As $|s|^a > |s|^b$ for $a > b > 1$ in $|s| > 1$, larger ϑ shows faster convergence in $|s| > 1$. While exponent is $\vartheta(1) = m_1 tanh(1)$ at $|s| = 1$, the condition $m_1 > \tanh(1)^{-1}$ keeps the exponent > 1 for $|s| \geq 1$. For $|s| < 1$, $f(s) \approx -1$ implies $\tanh(\alpha^{f(s)} |s|^2) \approx \tanh(\frac{|s|^2}{\alpha}) \approx 0$ as $\alpha >> 1$, thus $\vartheta \approx 0$. In this case, $|s|^0 = 1 > |s|^c$, $c > 0$, thus lower ϑ infers faster convergence. We can conclude that our exponent ϑ shows faster convergence for the entire s.

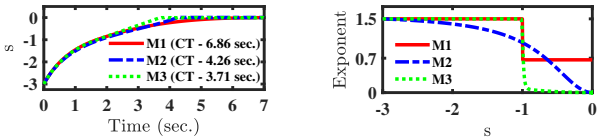


Fig. 1: Faster convergence i.e. lower CT (Convergence Time) of proposed M3 than M1, M2. CT is a time instant when $|y| \leq 0.0025$.

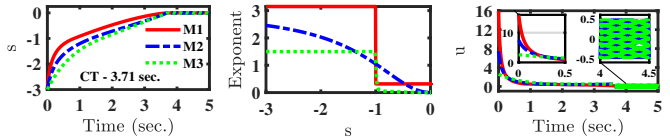


Fig. 2: Lower control input of M3 than M1, M2 in case of same CT. Maximum control effort in M1, M2, and M3 are 15.9, 7.4, 2.6.

In addition to faster convergence, (6) requires lower initial control effort. This feature is advantageous in fixed-time convergence when actuator saturation prevents the use of large input. We justify these claims by comparing with recent reaching laws, M1 [14], M2 [21], and proposed M3 in (6),

$$\text{M1} \implies \dot{s} = -k|s|^{m_3^{sign(|s|-1)}} sign(s) \tag{8}$$

$$\text{M2} \implies \dot{s} = -k|s|^{\frac{\lambda s^2}{1+\mu s^2}} sign(s) \tag{9}$$

Faster convergence: For a fair comparison, we set the same initial exponent of $|s|$ in all three methods using parameters $k = 0.5, \lambda = 2.7917, \mu = 1.75, \alpha = \beta = 20, m_1 = m_3 = 1.5$. In Fig. 1, the faster convergence of s in M3 is visible. This is due to the M3 exponent that remains ~ 1.5, same as initial up to $|s| \geq 1$, responsible for faster convergence in $|s| \geq 1$ than M2 because M2 exponent gradually decays from the initial exponent. For $|s| < 1$, M3 exponent quickly decays to zero, resulting in faster convergence than M1 and M2 where the M2 exponent is 0.67, and M1 exponent slowly decays to zero. In conclusion, (6) always refers to faster convergence than (9) whereas faster convergence than (8) for $|s| < 1$.

Lower initial control input: The control input u is compared when three methods achieve the same convergence time (CT) of $3.71sec$ with $k = 0.5, \lambda = 1.625, \mu = 0.55, m_3 = 3.15, m_1 = 1.5, \alpha = \beta = 20$ where u is the right side of the reaching law of respective methods. As, M3 has faster convergence than M1 and M2 in $|s| \leq 1$, the exponent of M3 is relatively lower compared to M1, M2 for $|s| \geq 1$ to have same CT. This causes a comparatively lower control effort than M1, M2 (Fig. 2).

C. Framework of Neural-FxSMC

In the zoomed portion of u in Fig 2, a chattering issue exists because of $\dot{s} = -k\ sign(s)$ as $s \to 0$ in (6). This pure switching action provides robustness near equilibrium at the expanse of control chattering which magnitude depends on gain k. Usually, k is set larger than the maximum disturbance bound [21] but since the bound is unknown beforehand, a heuristic value for a large k may lead to significant chattering.

The motivation behind *Neural-FxSMC* is to eliminate the chattering while retaining robustness. This is performed by adapting gain k intelligently based on real-time estimates of uncertainties using Radial Basis Function Network (RBFN). *Neural-FxSMC* uses NFTSS sliding surface (2), reaching law (6) and augment RBFN to obtain the control law as follows.

For dynamics (1), time derivative of s in (2) is written as:

$$\begin{aligned} \dot{s} &= \dot{y} + k_1 r_1 |y|^{r_1 - 1}\dot{y} + k_2 r_2 |\dot{y}|^{r_2 - 1}\ddot{y} \\ \dot{s} &= k_2 r_2 |\dot{y}|^{r_2 - 1} g_y (u + \Sigma) \end{aligned} \tag{10}$$

where, $\Sigma = g_y^{-1}(f_y + d + (|\dot{y}|^{2-r_2}(1 + k_1 r_1 |y|^{r_1 - 1})/r_2 k_2))$.

Noticeably, we consider the case of unknown system dynamics f_y, g_y with $g_y \neq 0$ and disturbance d in (1). We treat unknown dynamics and disturbance as lumped uncertainties, denoted as Σ. This Σ is estimated by RBFN via the universal approximation property of RBFN [28] and given as:

$$\Sigma = W^T \Phi(\kappa) + \varepsilon \tag{11}$$

where, $\kappa = [y, \dot{y}]$ is input to RBFN, selected based on Σ structure, $\Phi(\kappa) = [\phi_1, \phi_2, ...\phi_l] \in \mathbb{R}^l$ is a radial basis function vector with l hidden layer neurons, $\phi_l = exp(\frac{-||\kappa - c_l||^2}{2\hbar^2})$. c_l is the center of l^{th} hidden layer neuron and $\hbar$ is width of ϕ_l. W are ideal weight values, and ε is the approximation error.

Generally, W is unknown in advance, thus we estimate it as $\hat{W}$. The structure of *Neural-FxSMC* is proposed as:

$$u = -k_3 s - k_4 . |s|^{\varkappa(s)} sign(s) - \hat{\varepsilon} \tag{12}$$

where, s is in (2), $\varkappa(s) = m_2 \tanh(\alpha^{tanh(\beta(|s|-1))}|s|^2)$, $k_3 > 0.5$, and $m_2 > \tanh(1)^{-1}$. Noticeably, RBFN estimate is not canceled out with the uncertainty estimate, unlike [13], [23], [25]. Instead, it drives the switching gain $k_4 = |\hat{W}^T \Phi(\kappa)|$ to counter the uncertainties by updating RBFN weights $\hat{W}$ online. We also update ε online, which helps in direct gravity compensation in quadrotor control. These update laws are discussed in the next section with the stability analysis.

Note that $\dot{s}$ in (10) does not contain any term, unlike [14], which can lead to singularity issue. Next, the adaptive gain k helps attenuate the control chattering in (12) unlike [21].

III. *Neural-FxSMC* Design for Quadrotor

A 6Dof quadrotor dynamics can be written as follows [27]:

$$\begin{cases} \ddot{x} = -\frac{\rho_x \dot{x}}{m} + (\cos\phi \sin\theta \cos\psi + \sin\phi \sin\psi)\frac{u_1}{m} + d_1 \\ \ddot{y} = -\frac{\rho_y \dot{y}}{m} + (\cos\phi \sin\theta \sin\psi - \sin\phi \cos\psi)\frac{u_1}{m} + d_2 \\ \ddot{z} = -g - \frac{\rho_z \dot{z}}{m} + (\cos\phi \cos\theta)\frac{u_1}{m} + d_3 \\ \ddot{\phi} = (\frac{J_y - J_z}{J_x})\dot{\theta}\dot{\psi} - \frac{I_r}{J_x}\dot{\theta}\bar{\omega} - \frac{\rho_\phi}{J_x}\dot{\phi} + \frac{u_2}{J_x} + d_4 \\ \ddot{\theta} = (\frac{J_z - J_x}{J_y})\dot{\phi}\dot{\psi} - \frac{I_r}{J_y}\dot{\phi}\bar{\omega} - \frac{\rho_\theta}{J_y}\dot{\theta} + \frac{u_3}{J_y} + d_5 \\ \ddot{\psi} = (\frac{J_x - J_y}{J_z})\dot{\phi}\dot{\theta} - \frac{\rho_\psi}{J_z}\dot{\psi} + \frac{u_4}{J_z} + d_6 \end{cases} \tag{13}$$

where, x, y, z are the positions and ϕ, θ, ψ are roll, pitch and yaw angle respectively. m is quadrotor mass and J_x, J_y, J_z are moment of inertia about x, y, z respectively. g is gravity. ρ_x, ρ_y, ρ_z, ρ_ϕ, ρ_θ and ρ_ψ represents aerodynamic damping coefficients. $\bar{\omega} = \omega_4 + \omega_3 - \omega_2 - \omega_1$ is residual angular rotor speed. I_r is rotor inertia. $d_1, d_2, ...d_6$ are external disturbances.

We employ a 2-stage cascaded control scheme consisting of a position and attitude controller for quadrotor control (Fig 3). Hence, the quadrotor's coupled dynamics (13) are written as two subsystems, namely the position and attitude subsystem. If $\Gamma = [x\ y\ z]^T$ and $\zeta = [\phi\ \ \theta\ \ \psi]^T$, we can write

$$\begin{cases} \underline{Position\ subsystem} & \ddot{\Gamma} = f_p(\Gamma, \dot{\Gamma}) + g_p(\Gamma, \dot{\Gamma})u_p + d_p \quad \text{(14a)} \\ \underline{Attitude\ subsystem} & \ddot{\zeta} = f_a(\zeta, \dot{\zeta}) + g_a(\zeta, \dot{\zeta})u_a + d_a \quad \text{(14b)} \end{cases}$$

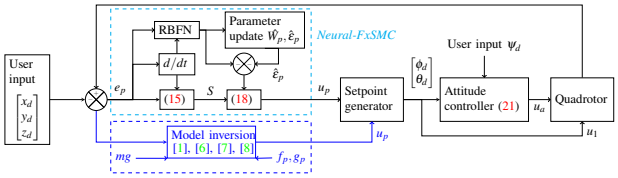


Fig. 3: Cascade control architecture of proposed approach

where $d_p = [d_1, d_2, d_3]^T$, $d_a = [d_4, d_5, d_6]^T$ and

$$f_p = \begin{bmatrix} -\frac{\rho_x \dot{x}}{m} \\ -\frac{\rho_y \dot{y}}{m} \\ -g - \frac{\rho_z \dot{z}}{m} \end{bmatrix}, f_a = \begin{bmatrix} (\frac{J_y - J_z}{J_x})\dot{\theta}\dot{\psi} - \frac{I_r \dot{\theta}\bar{\omega}}{J_x} - \frac{\rho_\phi \dot{\phi}}{J_x} \\ (\frac{J_z - J_x}{J_y})\dot{\phi}\dot{\psi} - \frac{I_r \dot{\phi}\bar{\omega}}{J_y} - \frac{\rho_\theta \dot{\theta}}{J_y} \\ (\frac{J_x - J_y}{J_z})\dot{\phi}\dot{\theta} - \frac{\rho_\psi}{J_z}\dot{\psi} \end{bmatrix}$$

$$g_a = diag\{\frac{1}{J_x}, \frac{1}{J_y}, \frac{1}{J_z}\}, g_p = diag\{\frac{1}{m}, \frac{1}{m}, \frac{1}{m}\}, u_a = [u_2, u_3, u_4]^T$$

$$u_p = [u_x, u_y, u_z]^T, u_x = (\cos\phi \sin\theta \cos\psi + \sin\phi \sin\psi)u_1$$

$$u_y = (\cos\phi \sin\theta \sin\psi - \sin\phi \cos\psi)u_1, u_z = (\cos\phi \cos\theta)u_1$$

Note that u_p is a virtual control input generated by position subsystem and transformed to thrust input $u_1 = ||u_p||$, desired roll angle $\phi_d = \sin^{-1}\left(\frac{u_x \sin\psi_d - u_y \cos\psi_d}{u_1}\right)$, and pitch angle $\theta_d = \tan^{-1}\left(\frac{u_x \cos\psi_d + u_y \sin\psi_d}{u_z}\right)$, while desired yaw ψ_d is provided by the user. The desired attitude $[\phi_d, \theta_d, \psi_d]$ is driven by the controller u_2, u_3, u_4. These four control inputs u_1, u_2, u_3, u_4 are applied to the quadrotor to achieve a desired 3D setpoint.

Generally, f_p, f_a, g_p, g_a are difficult to obtain precisely due to parametric/modeling uncertainties and unmodelled dynamics. Hence, the objective is to design controller u_p, u_a for unknown dynamics f_p, g_p, f_a, g_a and uncertainties d_p, d_a, without a priori bound assumption, such that, $\Gamma \to \Gamma_d, \zeta \to \zeta_d$ within a fixed-time T estimated without initial states.

A. Controller Design

To design u_p, consider the tracking error e_p defined as $e_p = \Gamma - \Gamma_d$. The sliding surface (8) is chosen as:

$$S = e_p + K_1 sig^{r_1}(e_p) + K_2 sig^{r_2}(\dot{e}_p) \tag{15}$$

where, $K_1, K_2 \in \mathbb{R}^{3\times3}_+$ are positive diagonal matrices, $1 < r_2 < r_1 < 2$. $S = [S_1, S_2, S_3]^T$, $e_p = [e_{p1}, e_{p2}, e_{p3}]^T$, $sig^{r_1}(e_p) = [|e_{p1}|^{r_1} sign(e_{p1}), |e_{p2}|^{r_1} sign(e_{p2}), |e_{p3}|^{r_1} sign(e_{p3})]^T$. The time derivative of S is $\dot{S} = \Xi(f_p + g_p u_p + d_p - \ddot{\Gamma}_d + F)$ and

$$g_p^{-1}\dot{S} = \Xi(u_p + \Omega_p) \tag{16}$$

where, $F = (r_2 K_2)^{-1}(I_3 + r_1 K_1 diag\{|e_p|^{r_1 - 1}) diag\{|\dot{e}_p|^{2-r_2}\}$, $\Xi = r_2 K_2 diag\{|\dot{e}_p|^{r_2 - 1}\}$, $\Omega_p = g_p^{-1}\left[f_p + d_p - \ddot{\Gamma}_d + F\right]$. $diag\{*\}$ is a diagonal matrix, I_3 is identity matrix. The lumped uncertainties Ω_p in (16) are estimated with an RBFN as:

$$\Omega_p = g_p^{-1}\left[f_p + d_p - \ddot{\Gamma}_d + F\right] = W_p^T \Phi_p + \varepsilon_p \tag{17}$$

We drop input with Φ_p as in (11) for brevity. We propose *Neural-FxSMC* for weight estimate $\hat{W}_p$ and gains $\Lambda_p \in \mathbb{R}^{3\times3}_+$,

$$u_p = -\Lambda_p S - diag\{|\hat{W}_p^T \Phi_p|\} sig^{\chi_p}(S) - \hat{\varepsilon}_p \tag{18}$$

where, $\chi_{pi} = m_{pi} \tanh(\alpha^{\tanh\beta(|S_i|-1)}|S_i|^2)$ is i^{th} entry of χ_p, $i = \{1,2,3\}$, $m_{pi} > \tanh(1)^{-1}$. The reaching law $(*)sig^{\chi_p}(S)$ helps in faster response toward S as shown in the section II-B.

Theorem 1: Controller (18) drives the response of the quadrotor position dynamics (14a) such that tracking error e_p converges to the neighborhood of zero in fixed-time T_p:

$$T_p \leq \frac{\sqrt{2^{1-m'_p \tanh(1)}}}{\Upsilon(m'_p \tanh(1) - 1)} + \frac{\sqrt{2}}{\delta\Upsilon} + \frac{2r_2}{\eta_4(r_1 - r_2)} + \frac{2r_2}{\eta_3(r_2 - 1)}$$

T_p is estimated before flying without initial quadrotor position where $\Upsilon, \delta, \eta_3, \eta_4, m'_p$ are given in Appendix B. Weights $\hat{W}_p, \hat{\varepsilon}_p$ remain bounded and updated online for $\rho_1, \rho_2 > 0$,

$$\dot{\hat{W}}_p = \frac{1}{\rho_1}(\Phi_p S^T \Xi - \varsigma_1 \hat{W}_p), \quad \dot{\hat{\varepsilon}}_p = \frac{1}{\rho_2}\Xi S \tag{19}$$

Proof: Kindly refer the Appendix B.

B. Attitude Controller

The desired attitude $\zeta_d = [\phi_d \ \ \theta_d \ \ \psi_d]^T$ follows attitude tracking error $e_a = \zeta - \zeta_d$. We define the sliding surface as:

$$S_a = e_a + K_3 sig^{r_3}(e_a) + K_4 sig^{r_4}(\dot{e}_a) \tag{20}$$

for $K_3, K_4 \in \mathbb{R}^{3\times3}_+$, $1 < r_4 < r_3 < 2$. The attitude controller is,

$$u_a = -\Lambda_a S_a - diag\{|\hat{W}_a^T \Phi_a|\} sig^{\chi_a}(S_a) - \hat{\varepsilon}_a \tag{21}$$

where $\chi_{ai} = m_{ai} \tanh(\alpha^{\tanh\beta(|S_{ai}|-1)}|S_{ai}|^2), i = \{1,2,3\}$ and $m_{ai} > \tanh(1)^{-1}$. The weights $\hat{W}_a$ and $\hat{\varepsilon}_a$ are updated online:

$$\dot{\hat{W}}_a = \frac{1}{\rho_3}(\Phi_a S_a^T \Xi_a - \varsigma_2 \hat{W}_a), \quad \dot{\hat{\varepsilon}}_a = \frac{1}{\rho_4}\Xi_a S_a \tag{22}$$

where $\rho_3, \rho_4 > 0$ and $\Xi_a = r_4 K_4 diag\{|\dot{e}_a|^{r_4 - 1}\}$. The fixed-time stability of u_a can be proven as Theorem 1.

Remark 1 (Uniqueness of Neural-FxSMC): Neural-FxSMC is not viable for sliding surface [21] in place of (2) because chattering persists at $e_p = 0$ due to nature of sliding surface.

Remark 2 (Unknown quadrotor dynamics): The controllers (18), (21) do not use any term related to quadrotor dynamics, uncertainties. Also, they have a very simple control structure.

Remark 3 (Chattering suppression): The weights W_p, W_a are updated based on uncertainties. In steady state or no uncertainty, $\hat{W}_p, \hat{W}_a$ reach zero, leading to chatter free response.

Remark 4 (Direct gravity compensation): Position dynamics (14a) consists of a constant gravity term challenging for RBFN to estimate. It acts as a bias term, the approximation error (ε_p) update helps in estimating this, enabling gravity compensation without explicit mass and gravity information.

This implementation of gravity compensation differs from [8], [9] (Fig. 3) that need mass and gravity along with the quadrotor dynamics. The adaptive approaches [1], [6], [7] require explicit mass and/or inertia adaptation with gravity.

Remark 5 (Robustness feature): Neural-FxSMC suppresses chattering and preserves robustness due to $sign(*)$ usage as (18). It may not be practically feasible in [1], [14], [21] where its approximation $tanh(*)$ is often used instead of $sign(*)$.

IV. Simulation Results

The quadrotor parameters are assumed as mass of $2.3Kg$, inertia $J_x = 0.01676$, $J_y = 0.01676$, $J_z = 0.02314$, $J_r = 0.016$ and hover thrust $mg = 22.54N$. For 40% hovering thrust, the maximum available thrust is $56.35N$. We conducted MATLAB simulations on quadrotor dynamics (13) for 1) Large setpoints, 2) Chattering reduction and tracking accuracy.

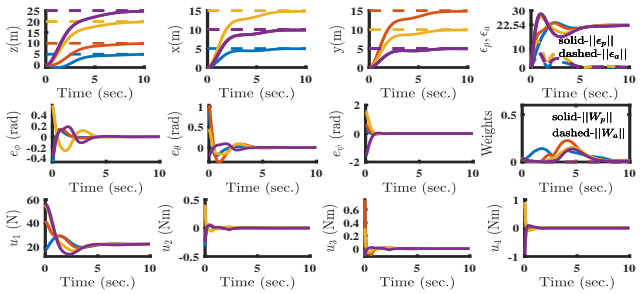


Fig. 4: State and control plots for large setpoints. Response of each setpoint is denoted by the same color as the corresponding plot.

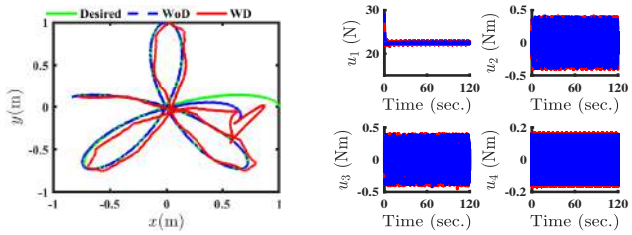


Fig. 5: CSMC without disturbance (WoD), with disturbance (WD).

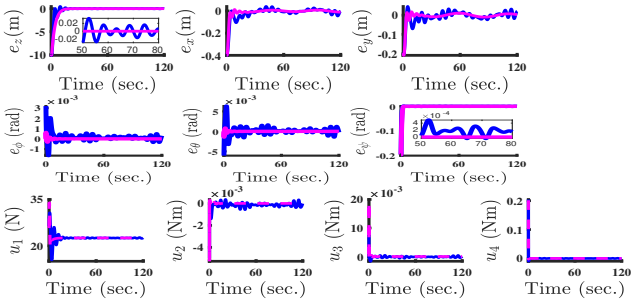


Fig. 6: Tracking performance of CSMC+NN and Neural-FxSMC.

Parameter Setting: We require two RBFNs, one for the position subsystem and the other for the attitude subsystem. Each RBFN architecture is chosen as $9-70-3$ with 9 input neurons, 70 hidden layer neurons (by trial and error), and 3 output neurons. The input to these RBFN are $[e_p, \dot{e}_p, \dot{\Gamma}_d]$ and $[e_a, \dot{e}_a, \ddot{\zeta}_d]$. All the weights are initialized with 0.01, and the RBFN centers are initialized randomly in $[-0.5\ 0.5]$. The *gains* are $\Lambda_p = diag\{1,1,1\}, \alpha = 10, \beta = 10, m_p = m_a = 1.5$. All the weights of RBFN and the approximation error are updated online with the adaptive laws (19) and (22).

A. Response to Large setpoints

Fixed-time control generally requires large control input for large initial states, we analyze this using the proposed *Neural-FxSMC*. The objective is to reach large 3D setpoints $\Gamma_d = [(5,5,5),(10,15,10),(15,10,20),(10,5,25)]$ (in meters) from origin. We also varied desired yaw angle $[\pi/16, 3\pi/16, \pi/2, -\pi/2]$ (in rad) to large values. In Fig 4, all the setpoints are reached successfully and attitude error also becomes zero. The maximum thrust input u_1 reaches to $56.3N$. Also, Roll, pitch and yaw torque i.e. u_1, u_2, u_3 also remain within $[-1\ \ 1]$ limits despite large setpoints. This justifies practical application of proposed fixed-time scheme.

Moreover, the gravity compensation can be justified in the ε_p update plot, which settles to the hovering thrust of $22.54N$.

B. Trajectory tracking accuracy and Chattering reduction

We justify that the faster convergence of the proposed reaching law can enhance the trajectory tracking accuracy. First, we simulate conventional SMC (CMSC) [1] with asymptotic error convergence to track a Clelia curve (defined in [9]) without and with disturbance $d_j = 0.01\dot{z} + 0.25\dot{x}\sin t + 0.1\dot{y}\sin t$ where $j = \{1,2,...6\}$. From Fig 5, the CMSC tracking has deteriorated in the presence of the disturbance. In addition, there is a large chattering in the control input.

Now to analyze the tracking accuracy of *Neural-FxSMC*, we augmented the CSMC with RBFN similarly as *Neural-FxSMC* and named it CSMC+NN. The same disturbance d_j is applied to both CSMC+NN and *Neural-FxSMC*. The obtained response is shown in Fig 6 where *Neural-FxSMC* has better tracking accuracy (see the position and attitude tracking plots) than CSMC+NN. In addition CSMC+NN handles the above disturbance well and control chattering is attenuated (see the control input plots Fig 6), unlike CSMC alone which justifies the RBFN augmentation.

V. Experimental Validation

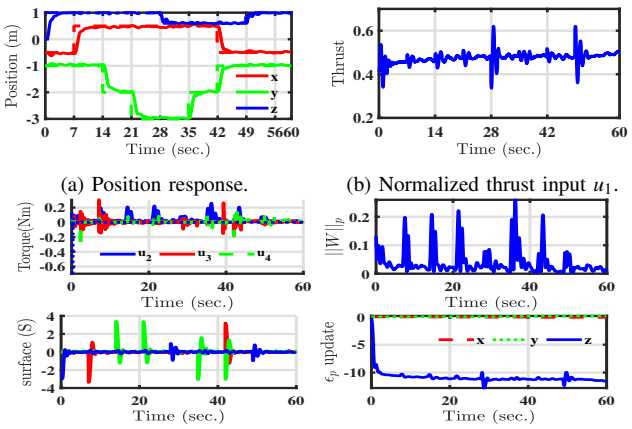
Experiments are conducted on a customized F450 quadrotor with Pixhawk flight controller. Intel Realsense-T265 is used to localize the quadrotor indoors. NVIDIA Jetson-Nano is used to run the localization and controller nodes onboard.

To show the effectiveness of the proposed scheme, we compare *Neural-FxSMC* with [14], which we refer FxSMC henceforth. The controller parameters are set the same as in simulations. See the video for experimental demonstrations.

A. Hover with Reference Jumps

We analyze the *Responsiveness* of *Neural-FxSMC* at sudden reference jumps in x, y, z, with hovering for $7sec$ at each setpoint (Fig 7a). Notably, quadrotor recovers its position with marginal error without overshooting at each setpoint.

The analysis of *Neural-FxSMC* is shown in Fig 7, where the RBFN weights and the approximation error ε_p converge after each setpoint change. ε_p has a non-zero value in z, compensating for gravity by updating the required thrust for hovering. Notably, the sliding surface (S) evolves at every reference jump and returns to zero quickly. Normalized thrust in $[0\ \ 1]$ and torque input remain within the limits.



(a) Position response. (b) Normalized thrust input u_1.

Fig. 7: Response with reference jump in the desired setpoints

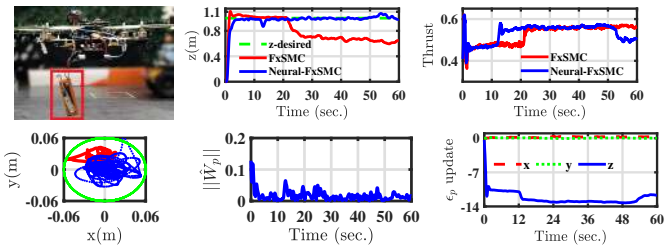


Fig. 8: Hover test while hanging a 260gm battery mid-flight

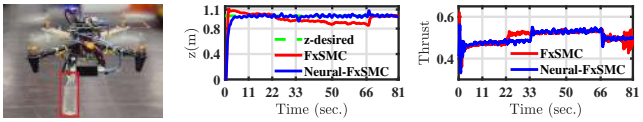


Fig. 9: Hang/remove half filled 200ml water bottle in hover

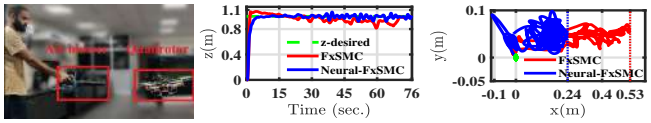


Fig. 10: Hover test in windy scene by air blower, hover position

B. Robust Hovering with Hanging Unknown Payload

We compare the response of FxSMC and *Neural-FxSMC* in case of sudden weight change during hovering. Two unknown payloads: a Lipo battery (260gm, 16% of the quadrotor weight), and a 200ml half-filled bottle (acting as a slosh payload) are suspended to the quadrotor off-center when the quadrotor is hovering at 1m height.

Fig 8 shows that the quadrotor stays within a 6cm circular bound with both schemes. However, FxSMC does not maintain the desired height after battery attachment (dropped to 60cm) because of no additional thrust update. Whereas, in *Neural-FxSMC*, the disturbance caused by hanging the battery is countered by ε_p update (Fig 8), leading to the required thrust update, and hence the desired height. After detaching the battery, the quadrotor stabilizes again at 1m, and ε_p returns to its original value.

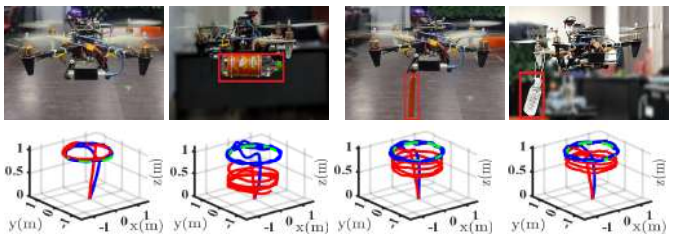
In the second case, a half-filled water bottle is hung while hovering to access induced vibrations due to the slosh payload. Fig 9 shows a height drop in the case of FxSMC, although FxSMC recovers height when the bottle is detached at $t = 66sec$. High vibrations are seen in the case of FxSMC (see video) due to generating large desired attitude commands. Also, the thrust command has more jittering in FxSMC, which is responsible for the high vibrations.

C. Robust Hovering in Heavy Wind Environment

We generate heavy wind via a high-speed air blower with a wind speed of 29m/sec at the nozzle (Fig 10). We set the hover height to 1m and then fired the blower continuously.

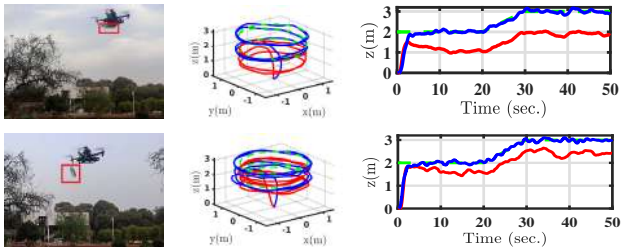
Interestingly, the quadrotor displaces to $x = 53$cm in the case of FxSMC due to the strong wind, whereas it displaces to only $x = 24$cm in the case of *Neural-FxSMC*. It happens due to the absence of a counter-force to overcome the wind effect in FxSMC. The average hovering position for FxSMC is $x = 26$cm, despite turning off the blower many times.

Whereas the quadrotor stays at $x = 12$cm on average with *Neural-FxSMC* despite the continuous firing of the air blower



(a) No Payload (b) Slosh (c) Hanging (d) Slosh on arm

Fig. 11: Different types of unknown payload transportation capability in circle tracking between FxSMC and *Neural-FxSMC*



(a) Snapshot (b) 3D plot (c) Varible height

Fig. 12: Outdoor experiment in helical trajectory with payload

without turning it off. The wind resistance of *Neural-FxSMC* is the effect of an additional force generated by RBFN weights and the approximation error ε_p. The $\hat{W}_p$ and ε_p are updated to counter the wind and maintain the setpoint.

D. Trajectory Tracking with Payload Transportation Case

This test is conducted to access payload-carrying capability while taking off with an attached unknown payload. Three types of payloads are considered as shown in Fig 11 to track a circle of radius 0.9m at 1m height.

With no payload, both FxSMC and *Neural-FxSMC* track the circle precisely. While with payloads, *Neural-FxSMC* tracks the circle precisely in all the cases while countering the disturbance caused by the payloads. Whereas, FxSMC does not produce enough thrust and fails in all payload scenarios.

E. Outdoor Experiments

Outdoor experiments are conducted to evaluate *Neural-FxSMC* in more realistic external disturbances e.g. wind gust. The quadrotor is required to track a helical trajectory with a hanging payload while varying heights up to 3m in a 1.2m radius. Two types of payloads, 600ml and 200ml fully filled water bottles, forming 34% and 12% of quadrotor weight respectively, are suspended off-center on quadrotor (Fig 12). Precise tracking with payload is visible by *Neural-FxSMC*, whereas FxSMC can not track. *Neural-FxSMC* also tracks desired variable height in contrast to FxSMC as in Fig 12.

F. Comparative Analysis and Discussion

Quantitative analysis is performed with performance index (P.I.), i.e. mean square error (MSE), average thrust T_{avg} command and average desired attitude angles (ϕ_d, θ_d). These are evaluated for three cases, i.e. hover, indoor and outdoor tracking over 10 runs of each task shown earlier. Table I shows

TABLE I: Comparison between FxSMC [14] & *Neural-FxSMC*

P.I	Approach	Hover	Indoor	Outdoor
MSE (m)	FxSMC	0.0897	0.2208	0.4007
	Neural-FxSMC	**0.0306**	**0.1149**	**0.1080**
T_{avg}	FxSMC	0.5166	0.5371	0.5982
	Neural-FxSMC	**0.5181**	**0.5358**	**0.6023**
ϕ_d, θ_d (rad)	FxSMC	0.017, 0.019	0.03, 0.023	0.033, 0.036
	Neural-FxSMC	**0.008, 0.007**	**0.016, 0.013**	**0.018, 0.017**

better tracking accuracy of *Neural-FxSMC* with only slightly high thrust input. The desired attitude is higher in FxSMC, which is the reason behind higher oscillations and a source of chattering. It can be seen in **attached video** with hanging slosh payload causing high oscillations in FxSMC.

Remark 6: A nice explanation of the effectiveness of neural network (NN) approximation in adaptive setting is provided in [9]. As NN by nature, can approximate well a function of states, we are addressing state-dependent uncertainties. Although time-varying disturbance can be tackled by integral action [9] in NN, our method helps to counter these uncertainties by approximation error update.

Remark 7: Our approach requires a very small initial control input (Sec. II-B) and can handle a general issue with fixed-time methods of generating large control effort for far initial conditions. An explicit consideration of actuator saturation can be investigated in future, as discussed in [15].

VI. Conclusions

This paper proposed a new fast fixed-time convergent sliding mode control using an adaptive neural network to handle control singularity and chattering issues. We establish fixed-time convergence of existing NFTSS, which is proven only finite-time convergent till now. NFTSS is then augmented with novel reaching law and neural network to consider unknown quadrotor dynamics and uncertainties. The chattering is attenuated by RBFN estimate by driving switching gain adaptively. We named this *Neural-FxSMC*.

Proposed *Neural-FxSMC* has several benefits. It doesn't require quadrotor dynamics and offers direct gravity compensation without relying on mass and gravity. Also, it allows the estimation of settling time before the flight, which can be adjusted to match task requirements without prior knowledge of initial states. Experiments demonstrate the effectiveness of this approach in unknown payload transport and disturbance-rich environments, outperforming existing FxSMC. Integrating motion planning in complex scenarios with obstacles is a potential direction for further research.

Appendix A

Consider a Lyapunov function $V = \frac{1}{2}s^2$. With the help of (6), the time derivative of V i.e. $\dot{V} = s\dot{s}$ can be expressed as:

$$\dot{V} = -k|s|.|s|^{m_1 \tanh(\alpha^{f(s)}|s|^2)} = -k|s|.|s|^{\vartheta(s)} \tag{23}$$

The fixed-time proof is divided into two parts as follows.

If $|s| \geq 1$: In this case, $\vartheta(s) > \vartheta(1) = m_1 \tanh(1) > 1$. Thus, (23) becomes $\dot{V} = -k|s|^{\vartheta(s)+1} \leq -k|s|^{\vartheta(1)+1}$. Also,

$$\dot{V} \leq -k2^{\frac{\vartheta(1)+1}{2}} V^{\frac{\vartheta(1)+1}{2}} = -k'V^{\frac{\vartheta(1)+1}{2}} \tag{24}$$

where $k' = 2^{\frac{\vartheta(1)+1}{2}}k$, We can further solve (24) as:

$$\int_{V(0)}^{1} \frac{dV}{V^{\frac{\vartheta(1)+1}{2}}} \leq \int_0^{t_1} -k'dt \implies \frac{2}{\vartheta(1)-1}\left[V^{\frac{1-\vartheta(1)}{2}}\right]_{V(0)}^{1} \leq k't_1$$

$$\implies t_1 \leq \frac{2}{k'(\vartheta(1)-1)}\left(1 - V(0)^{\frac{1-\vartheta(1)}{2}}\right) \leq \frac{2}{k'(\vartheta(1)-1)} \tag{25}$$

If $|s| \leq 1$: We can write $0 < min(|s|^{\vartheta(s)}) = \ell < 1$. So (23) becomes $\dot{V} \leq -k|s|.|s|^{\vartheta(s)} \leq -k\ell|s| = -k\ell\sqrt{2V}$. By solving,

$$\int_1^0 \frac{dV}{\sqrt{V}} \leq \int_{t_1}^{t_s} -(\sqrt{2}k\ell)dt \implies t_s \leq t_1 + \frac{\sqrt{2}}{k\ell}$$

By substituting t_1 from (25), we get convergence time for s:

$$t_s \leq \frac{\sqrt{2^{1-m_1\tanh(1)}}}{k(m_1\tanh(1)-1)} + \frac{\sqrt{2}}{k\ell} \tag{26}$$

This completes the proof.

Appendix B

Theorem 1 proof: By defining the weight estimation error and approximation error respectively as $\tilde{W}_p = W_p - \hat{W}_p, \tilde{\varepsilon}_p = \varepsilon_p - \hat{\varepsilon}_p$, consider a Lyapunov function as follows:

$$V_1 = \frac{1}{2}S^T g_p^{-1} S + \frac{\rho_1}{2}tr(\tilde{W}_p^T \tilde{W}_p) + \frac{\rho_2}{2}\tilde{\varepsilon}_p^T \tilde{\varepsilon}_p \tag{27}$$

The time derivative ($\dot{V}$) can be simplified as follows:

$$\begin{aligned}\dot{V}_1 &= S^T g_p^{-1}\dot{S} - \rho_1 tr(\tilde{W}_p^T \dot{\hat{W}}_p) - \rho_2 \tilde{\varepsilon}_p^T \dot{\hat{\varepsilon}}_p \\ &= -S^T \Xi \Lambda_p S + S^T \Xi \tilde{\varepsilon}_p + S^T \Xi \tilde{W}_p^T \Phi_p + S^T \Xi \hat{W}_p^T \Phi_p \\ &\quad - (|S|^{\chi_p+1})^T \Xi |\hat{W}_p^T \Phi_p| - \rho_1 tr(\tilde{W}_p^T \dot{\hat{W}}_p) - \rho_2 \tilde{\varepsilon}_p^T \dot{\hat{\varepsilon}}_p \end{aligned} \tag{28}$$

$$\dot{V}_1 \leq -S^T \Xi \Lambda_p S + tr(\varsigma_1 \tilde{W}_p^T \hat{W}_p) + \sum_{i=1}^{3} |S_i|\Xi_i |(\hat{W}_p^T \Phi_p)_i|(1 - |S_i|^{\chi_p})$$

The term $|S|^{\chi_p}$ never reaches zero, and its minimum value can be found analytically as $\delta = min(|S_i|^{\chi_{pi}}) \leq 1$, χ_{pi} is i^{th} entry of χ_p. Define $\eth = 1 - \delta \geq 0$, $\dot{V}_1$ can be written as:

$$\begin{aligned}\dot{V}_1 &\leq -S^T \Xi \Lambda_p S + tr(\varsigma_1 \tilde{W}_p^T \hat{W}_p) + \sum_{i=1}^{3} \eth_i |S_i| \Xi_i |(\hat{W}_p^T \Phi_p)_i| \\ &\leq -\sum_{i=1}^{3} |S_i|^2 \Xi_i \left(\Lambda_{pi} - \frac{\eth |\hat{W}_p^T \Phi_p|_i}{S_i}\right)^2 - \frac{\varsigma_1 tr(\tilde{W}_p^T \tilde{W}_p)}{2} + \frac{\varsigma_1 ||W_p||^2}{2}\end{aligned}$$

$\dot{V}_1 \leq -\mho V + \aleph$ where $\aleph = \frac{\varsigma_1}{2}||W_p||^2$, $\mho = min\{2\flat, \frac{\varsigma_1}{\rho_1}, \frac{\varsigma_2}{\rho_2}\}$, $\flat = min(\Xi_i g_{pi} \Lambda_{pi} - \eth |\hat{W}_p^T \Phi_p|_i / S_i)$. Hence, the uniform ultimate boundedness (UUB) of $S, \tilde{W}_p, \tilde{\varepsilon}_p$ is guaranteed from (27). Combining with the boundedness of W_p, ε_p, this implies the UUB of $\hat{\varepsilon}_p, \hat{W}_p$ and hence of Ξ, Φ_p. So, there exist a $\wp$ such that $S^T \Xi \tilde{\varepsilon}_p + S^T \Xi W_p^T \Phi_p \leq \wp$ is satisfied.

To prove fixed-time convergence, consider the Lyapunov function $V_2 = \frac{1}{2}S^T g_p^{-1} S$. Time derivative $\dot{V}_2$ is simplified as:

$$\dot{V}_2 = -S^T \Xi \Lambda_p S + S^T \Xi \tilde{\varepsilon}_p + S^T \Xi W_p^T \Phi_p - (|S|^{\chi_p+1})^T \Xi |\hat{W}_p^T \Phi_p|$$

The proof is divided into two parts i.e. $S \geq 1$ and $S < 1$ **(i)** $|S_i| \geq 1$**:** In this case, $\chi_{pi} \geq \chi_{pi}(1) = m_{pi}\tanh(1)$, $min(\chi_{pi}(1)) = min(m_{pi})\tanh(1) = c_p > 1$ that shows $\chi_p + 1 \geq c_p + 1 > 2$. Hence, $\dot{V}_2$ can be written as:

$$\dot{V}_2 \leq -(|S|^{c_p+1})^T \Xi |\hat{W}_p^T \Phi_p| + \wp \leq -\sum_{i=1}^{3} |S_i|^{c_p+1} \Xi_i |(\hat{W}_p^T \Phi_p)_i|$$

$$+ \wp \leq -\Upsilon(2V)^{\frac{c_p+1}{2}} + \wp \text{ for } \Upsilon = min(|\Xi_i (\hat{W}_p^T \Phi_p)_i| g_{pi})$$

We can say that S will converge to the neighbourhood of $|S| = 1$ i.e. $S \in \left\{ V_2 \leq 1 + (\frac{\wp}{\Upsilon(1-a_0)})^{\frac{2}{c_p+1}} \right\}$ for $0 < a_0 < 1$ from any $|s(0)| > 1$ within the fixed-time T_{s1} as $T_{s1} \leq \frac{\sqrt{2^{1-c_p}}}{\Upsilon(c_p-1)}$.

(ii) $|S_i| \leq 1$**:** In this case, $\dot{V}_2 - (|S|^{c_p+1})^T \Xi |\hat{W}_p^T \Phi_p| + \wp \leq -\delta\Upsilon\sqrt{2V} + \wp$ as $\delta = min(|S_i|^{\chi_{pi}})$. So S will converge to the neighborhood of origin i.e. $S \in \left\{ V_2 \leq (\frac{\wp}{\Psi(1-a_0)})^2 \right\}$ for $0 < a_0 < 1$ within a fixed time $T_{s2} \leq \frac{2}{\sqrt{2}\delta\Upsilon} = \frac{\sqrt{2}}{\delta\Upsilon}$.

We can say that sliding surface S will reach the neighborhood of origin in $\Delta = \left\{ S | V_2 \leq min\left[\frac{\wp}{\Upsilon(1-a_0)}, \frac{\wp}{\Psi(1-a_0)} \right] \right\}$ region within a fixed-time $T_s = T_{s1} + T_{s2}$.

Now, we prove that the tracking error e_p will also converge in the neighborhood of origin within fixed-time time T_e. Once, $S_i \leq \Delta_i$, $i \in \{1,2,3\}$, we can write (15) as:

$$S_i = e_{pi} + K_{1i} sig^{r_1}(e_{pi}) + K_{2i} sig^{r_2}(\dot{e}_{pi}) = \bar{\Delta}_i, \quad |\bar{\Delta}_i| \leq \Delta_i$$

where e_{pi} is i^{th} entry of e_p. The above expression simplifies:

$$e_{pi} + K_{1i} sig^{r_1}(e_{pi}) + K_{2i}\left(1 - \frac{\bar{\Delta}_i}{K_{2i} sig^{r_2}(\dot{e}_{pi})}\right) sig^{r_2}(\dot{e}_{pi}) = 0$$

For $\left(1 - \frac{\bar{\Delta}_i}{K_{2i} sig^{r_2}(\dot{e}_{pi})}\right) > 0$, the above expression is equivalent to (2) (when $s = 0$). Hence, the convergence time can be estimated on similar lines in section II-A. We can say that, $\dot{e}_{pi}$ will converge to $\dot{e}_{pi} \leq \left(\frac{|\bar{\Delta}_i|}{K_{ai}}\right)^{\frac{1}{r_2}} \leq \left(\frac{\Delta_i}{K_{ai}}\right)^{\frac{1}{r_2}}$ within some fixed-time $T_e \leq \frac{2r_2}{\eta_4(r_1-r_2)} + \frac{2r_2}{\eta_3(r_2-1)}$ where, $\eta_3 = (\frac{min(K_1)}{min(K_2)} 2^{\frac{r_2+1}{2}})^{\frac{1}{r_2}}$, $\eta_4 = (\frac{1}{min(K_2)} 2^{\frac{r_1+r_2}{2}})^{\frac{1}{r_2}}$. Furthermore, we can write

$$\begin{aligned} e_{pi} &\leq |e_{pi} + K_{1i} sig^{r_1}(e_{pi})| \leq \bar{\Delta}_i - K_{2i} sig^{r_2}(\dot{e}_{pi}) \\ e_{pi} &\leq |\bar{\Delta}_i| + K_{2i} sig^{r_2}(\dot{e}_{pi}) \leq 2\Delta_i \end{aligned} \tag{29}$$

Hence, we can conclude that the tracking error e_{pi} will also converge in the $e_{pi} \leq 2\Delta_i$ within fixed time T_e.

So, the settling time $T_p = T_s + T_e$ is estimated as:

$$T_p \leq \frac{\sqrt{2^{1-m'_p \tanh(1)}}}{\Upsilon(m'_p \tanh(1) - 1)} + \frac{\sqrt{2}}{\delta\Upsilon} + \frac{2r_2}{\eta_4(r_1-r_2)} + \frac{2r_2}{\eta_3(r_2-1)}$$

where $m'_p = min(m_{pi})$. This completes the proof.

REFERENCES

[1] V. N. Sankaranarayanan, S. Roy and S. Baldi, "Aerial Transportation of Unknown Payloads: Adaptive Path Tracking for Quadrotors," IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 2020, pp. 7710-7715.

[2] H. Wang, X. Ye, Y. Tian, G. Zheng and N. Christov, "Model-free-based terminal SMC of quadrotor attitude and position," IEEE Trans. on Aerospace & Elect. Systems, vol. 52, no. 5, pp. 2519-2528, 2016.

[3] S. Lian, W. Meng, K. Shao, J. Zheng, S. Zhu and H. Li, "Full Attitude Control of a Quadrotor Using Fast Nonsingular Terminal Sliding Mode With Angular Velocity Planning," IEEE Trans. on Industrial Electronics, vol. 70, no. 4, pp. 3975-3984, April 2023.

[4] A. Polyakov, "Nonlinear Feedback Design for Fixed-Time Stabilization of Linear Control Systems," IEEE Trans. on Automatic Control, vol. 57, no. 8, pp. 2106-2110, Aug. 2012.

[5] Wang, S., Polyakov, A., & Zheng, G. (2021). Generalized homogenization of linear controllers: Theory and experiment. International Journal of Robust and Nonlinear Control, 31(9), 3455-3479.

[6] T. Huang, T. Li and C. L. P. Chen, "Adaptive Tracking Control for a Quadrotor System Subject to Internal and External Uncertainties," IEEE Trans. on Circuits and Systems II: Express Briefs, 2022, doi: 10.1109/TCSII.2022.3218661.

[7] S. Lian et al., "Adaptive Attitude Control of a Quadrotor Using Fast Nonsingular Terminal Sliding Mode," IEEE Trans. on Industrial Electronics, vol. 69, no. 2, pp. 1597-1607, Feb. 2022.

[8] M. Bisheban and T. Lee, "Geometric Adaptive Control With Neural Networks for a Quadrotor in Wind Fields," IEEE Trans. on Control Systems Technology, vol. 29, no. 4, pp. 1533-1548, July 2021.

[9] G. Yu, J. Reis and C. Silvestre, "Quadrotor Neural Network Adaptive Control: Design and Experimental Validation," IEEE Robotics and Automation Letters, vol. 8, no. 5, pp. 2574-2581, May 2023.

[10] K. Liu and R. Wang, "Antisaturation Adaptive Fixed-Time Sliding Mode Controller Design to Achieve Faster Convergence Rate and Its Application," IEEE Trans. on Circuits and Systems II: Express Briefs, vol. 69, no. 8, pp. 3555-3559, Aug. 2022.

[11] Zhang, Y., Tang, S., & Guo, J, "Adaptive terminal angle constraint interception against maneuvering targets with fast fixed-time convergence", International Journal of Robust & Nonlinear Control, vol. 28, no. 8, pp. 2996-3014, 2018.

[12] H. Long, T. Guo and J. Zhao, "Adaptive Disturbance Observer-Based Novel Fixed-Time Nonsingular Terminal Sliding-Mode Control for a Class of DoF Nonlinear Systems," IEEE Trans. on Industrial Informatics, vol. 18, no. 9, pp. 5905-5914, Sep. 2022.

[13] M. Gao, L. Ding and X. Jin, "ELM-Based Adaptive Faster Fixed-Time Control of Robotic Manipulator Systems," IEEE Trans. on Neural Networks & Learning Systems, vol. 34, no. 8, pp. 4646-4658, 2023.

[14] L. Yu, G. He, X. Wang and L. Shen, "A Novel Fixed-Time Sliding Mode Control of Quadrotor With Experiments and Comparisons," in IEEE Control Systems Letters, vol. 6, pp. 770-775, 2022.

[15] Z. Liu, Y. Zhao, O. Zhang, W. Chen, J. Wang, Y. Gao and J. Liu, "A Novel Faster Fixed-Time Adaptive Control for Robotic Systems With Input Saturation," IEEE Trans. on Industrial Electronics, doi: 10.1109/TIE.2023.3281701.

[16] W. Shang et al., "Robust fixed time controller with motor dynamics and composite disturbances for a quadrotor unmanned aerial vehicle," International Journal of Advanced Robotic Systems. 2020;17(5).

[17] X. Yu, P. Li and Y. Zhang, "Fixed-Time Actuator Fault Accommodation Applied to Hypersonic Gliding Vehicles," IEEE Trans. on Automation Science & Engg., vol. 18, no. 3, pp. 1429-1440, July 2021.

[18] H. Li, Y. Cai, "On SFTSM control with fixed-time convergence," IET Control Theory & Applications, Vol. 11, no. 6, pp. 766-773, Apr. 2017.

[19] Chen, Qiang, Shuzong Xie, Xiongxiong He. "Neural-network-based adaptive singularity-free fixed-time attitude tracking control for spacecrafts." IEEE Trans. on Cybernetics 51.10 (2020): 5032-5045.

[20] M. Gao, X. Jin and L. Ding, "A Novel Nonsingular Fixed-Time Sliding Mode Control of Uncertain Euler–Lagrange Systems," IEEE Systems Journal, vol. 17, no. 1, pp. 467-478, March 2023.

[21] E. Moulay, V. Léchappé, E. Bernuau and F. Plestan, "Robust Fixed-Time Stability: Application to Sliding-Mode Control," IEEE Trans. on Automatic Control, vol. 67, no. 2, pp. 1061-1066, Feb. 2022.

[22] I. M. Boiko, "Chattering in sliding mode control systems with boundary layer approximation of discontinuous control," International Journal of Systems Science, Vol. 44, no. 6, pp. 1126-1133, 2013.

[23] J. Zhang et al. , "Fixed-Time Dynamic Surface Control for Pneumatic Manipulator System With Unknown Disturbances," IEEE Robotics and Automation Letters, vol. 7, no. 4, pp. 10890-10897, Oct. 2022.

[24] D. Y. N.-Chávez and J. A. Moreno, "Second-order sliding mode output feedback controller with adaptation", International Journal of Adaptive Control & Signal Processing, vol. 30, no. 8-10, pp. 1523-1543, 2016.

[25] Z. Li, L. Chen and H. Wang, "Fixed-time Sliding Mode-based Adaptive Path Tracking Control of Maize Plant Protection Robot via Extreme Learning Machine," IEEE Robotics and Automation Letters, doi: 10.1109/LRA.2023.3244125.

[26] C. Liu, G. Wen, Z. Zhao and R. Sedaghati, "Neural-Network-Based Sliding-Mode Control of an Uncertain Robot Using Dynamic Model Approximated Switching Gain," IEEE Trans. on Cybernetics, vol. 51, no. 5, pp. 2339-2346, May 2021.

[27] S. C. Yogi, V. K. Tripathi and L. Behera, "Adaptive Integral Sliding Mode Control Using Fully Connected Recurrent Neural Network for Position and Attitude Control of Quadrotor," IEEE Trans. on Neural Networks & Learning Systems, vol. 32, no. 12, pp. 5595-5609, 2021.

[28] J. Park and I. W. Sandberg, "Universal Approximation Using Radial-Basis-Function Networks," Neural Computation, vol. 3, no. 2, pp. 246-257, June 1991.

[29] Yang, L., and Yang, J., "Nonsingular fast terminal sliding-mode control for nonlinear dynamical systems", International Journal of Robust and Nonlinear Control, vol. 21, no. 16, pp. 1865-1879, 2011.