applied sciences

MDPI

Article

Fast Fractional-Order Terminal Sliding Mode Control for Seven-Axis Robot Manipulator

Jie Wang, Min Cheol Lee *, Jae Hyung Kim and Hyun Hee Kim

School of Mechanical Engineering, Pusan National University, Busan 46241, Korea; wj@pusan.ac.kr (J.W.); 11045kjh@naver.com (J.H.K.); sleepingjongmo@nate.com (H.H.K.)

* Correspondence: mclee@pusan.ac.kr; Tel.: +82-051-510-3081

Received: 8 October 2020; Accepted: 30 October 2020; Published: 2 November 2020

Abstract: This paper proposes a novel controller, fast fractional-order terminal sliding mode control (FFOTSMC), for a seven-degree-of-freedom (7-DOF) robot manipulator with tracking control. The new controller applies the fractional-order derivative on both the sliding surface design and the sliding control/reaching law. Compared to previous research, which only applies the fractional-order derivative on the sliding surface design, the proposed controller has a faster convergence for reaching the sliding surface and maintaining stay on it because of the new fractional-order control law, which helps the tracking accuracy. To implement the controller on the robot with less chattering, a sliding perturbation observer (SPO) is used to estimate the disturbance and uncertainties. Stability analysis is analyzed using Lyapunov functions for fractional-order systems. The controller performance is evaluated by a simulation of a single-input and single-output (SISO) system in MATLAB Simulink and experiments on the robot manipulator.

Keywords: robot control; sliding mode control; robust control; fractional calculus

1. Introduction

Sliding mode control (SMC) has been widely studied and applied to robot manipulators because of its simplicity of implementation, fast global convergence, high robustness to external variations, and insensitivity to modeling error and system parameter variations [1]. SMC includes a conventional linear sliding mode (LSM) control, which is asymptotically stable, and a terminal sliding mode (TSM) control with finite-time stability [2,3]. Typically, it includes two steps to design an SMC: the choice of sliding surface and the control/reaching law to reach the sliding surface and maintain stay on it.

TSM control (TSMC) utilizes a nonlinear sliding surface instead of a linear sliding surface in sliding surface design. It generates faster convergence, less tracking errors than LSM control (LSMC), and finite stability for reaching the equilibrium point. The control/reaching law in TSMC also contains a nonlinear term, which allows the sliding surface to be reached in finite time with fast convergence. Compared to the LSMC, TSMC shows various superior properties such as fast and finite-time convergence and few tracking errors. However, in TSMC, the first-order derivative of the nonlinear term in TSM causes a singularity problem. In recent years, many researchers have focused on solving the singularity [4,5]. Using fractional calculus can avoid this singularity because of the hereditary and memory property, such as fractional-order TSMC (FOTSMC) [6]. FOTSMC was proposed with a terminal sliding surface using the fractional-order derivative. It can help avoid the singularity, which exists by taking the first-order derivative to the sliding surface. In addition to this, compared to conventional TSMC, FOTSMC has a faster convergence speed.

The study on applying the fractional-order derivative to SMC techniques starts by applying it to the LSMC first. A fraction-order sliding mode control (FOSMC) was proposed based on LSM [7], and it exhibited better performance than the conventional LSMC (integer-order derivative-based).

The reason has been roughly analyzed and discussed. In [8], researchers analyzed and found that the solution of a fractional-order (larger than zero and less or equal to one) differential equation could be presented as a Mittag-Leffler function. This mathematical solution explains why FOSMC exhibited a higher convergence speed than LSMC (see Appendix A). However, FOSMC is not globally faster than LSMC. It has slow convergence near the terminal where small values are, which is discussed in detail in Appendix A.

FOTSMC can compensate for the slow convergence drawback near the equilibrium of FOSMC because the terminal sliding surface has a faster convergence speed than the linear sliding surface, especially in the field of small values. Compared to TSMC and FOSMC, FOTSMC is a complementary method, the terminal sliding surface helps the slow convergence in the interval of small values, and the fractional-order derivative accelerates the convergence in other intervals.

Even though many researchers have demonstrated the performance of FOTSMC through real applications and simulations [9–12], the specific reason why FOTSMC performs better than FOSMC has not been introduced clearly, which is discussed in Appendix A. Additionally, all previous works on FOTSMC only focused on applying the fractional-order derivative on the sliding surface design, without paying attention to the design of the control/reaching law for reaching the sliding surface, e.g., FOTSMC in [9] only utilized the fractional calculus in designing the sliding surface. It applied a control law, obtained by taking the first-order derivative to the sliding surface with respect to time, to reach the sliding surface and maintain stay on it. The control input of FOTSMC was merely derived simply by the desired dynamics of the sliding surface.

In this research, the proposed controller, fast fractional-order terminal sliding mode control (FFOTSMC), uses the fractional-order derivative on both the sliding surface design and the control/reaching law. The new control law is conducive to reaching the sliding surface faster than the previously studied one (FOTSMC), designed using the integer-order derivative only. This fast reaching law is associated with fast convergence.

The controller is designed for a seven-degree-of-freedom (7-DOF) robot manipulator. Due to its complex dynamics, only using FFOTSMC is not enough to have an ideal performance. The high frequency switching in the control input of FFOTSMC can ensure the robustness of FFOTSMC; however, it could make the robot oscillate. In this research, we apply the sliding mode perturbation observer (SPO) [13] to estimate the perturbation. The estimated perturbation is used to compensate for the uncertainties and disturbances in the controller design. It reduces the switching gain of the control input of the FFOTSMC, which decreases the amplitude of chattering. SPO is efficient with high estimation accuracy for a low-frequency vibration, which has been demonstrated in electric motors and hydraulic cylinder actuated systems [14,15]. Through the compensation for the uncertainties and disturbances, the control precision is also improved.

The contributions of this paper are summarized as follows.

(1) The new proposed controller, FFOTSMC, uses fractional-order derivatives on both sliding surface design and sliding control/reaching law. It has faster convergence and more outstanding tracking accuracy than FOTSMC, which has been researched previously.
(2) The new controller FFOTSMC was designed to control a real 7-DOF robot manipulator for trajectory tracking. SPO is used to estimate the disturbance and uncertainties for reducing the chattering, ensuring the controller can be implemented in practice.
(3) Stability is analyzed using the Lyapunov functions for fractional-order systems [16].

The paper is organized as follows. In Section 2, the 7-DOF robot manipulator modeling, fundamental definitions, two useful properties, and one lemma for proving the stability are presented. Section 3 describes the stability analysis of the new proposed FFOTSMC with SPO (FFOTSMCSPO). In Section 4, simulations using the proposed controller and four different controllers are compared in MATLAB for a single-input and single-output (SISO) system. Then, comparative experiments on the robot manipulator are discussed. The conclusion of this work is shown in Section 5.

2. Preliminaries

2.1. Mathematical Model for Robot Manipulators and Problem Formulation

Consider an n-links robotic manipulator is model as

$$\mathbf{M}(\mathbf{x})\ddot{\mathbf{x}} + \mathbf{C}(\mathbf{x}, \dot{\mathbf{x}})\dot{\mathbf{x}} + \mathbf{G}(\mathbf{x}) = \boldsymbol{\tau}(t) + \mathbf{d}(t) \tag{1}$$

where $\mathbf{x} = [x_1, \ldots, x_n]^T \in R^n$ is the position state of the joints; $\dot{\mathbf{x}} \in R^n$ is the velocity state vector and $\ddot{\mathbf{x}} \in R^n$ is the acceleration state vector; $\mathbf{d} \in R^n$ is the disturbance vector; $\boldsymbol{\tau} \in R^n$ is the torque vector acted on the joints; $\mathbf{M} \in R^{n \times n}$, $\mathbf{C} \in R^{n \times n}$ and $\mathbf{G} \in R^n$ are the symmetric positive definite inertia matrix; the Coriolis and centrifugal matrix, and the gravitational matrix, respectively. After identifying system dynamics, the system in (1) can be rewritten with modeling parameters and uncertainties as

$$\ddot{\mathbf{x}} = \mathbf{M}_o(\mathbf{x})^{-1}(\boldsymbol{\tau}(t) - \mathbf{C}_o(\mathbf{x}, \dot{\mathbf{x}})\dot{\mathbf{x}}) + \boldsymbol{\Psi} \tag{2}$$

where $\mathbf{M}_o$ and $\mathbf{C}_o$ are identified dynamic parameters, and $\boldsymbol{\Psi} = [\Psi_1, \ldots, \Psi_n]^T \in R^n$ represents the perturbation, which contains the uncertainties of dynamics, the unknown gravitational term, and the disturbance as

$$\boldsymbol{\Psi}(\mathbf{x}, \dot{\mathbf{x}}, \ddot{\mathbf{x}}, t) = \mathbf{M}_o^{-1}(-\Delta\mathbf{M}(\mathbf{x})\ddot{\mathbf{x}} - \Delta\mathbf{C}(\mathbf{x}, \dot{\mathbf{x}})\dot{\mathbf{x}} - \mathbf{G}(\mathbf{x}) + \mathbf{d}(t)) \tag{3}$$

where $\Delta\mathbf{M}$ and $\Delta\mathbf{C}$ represent the uncertainties of the inertia and the Coriolis and centrifugal term of dynamics, respectively. $\boldsymbol{\Psi}$ is assumed to have a boundary limited as $|\boldsymbol{\Psi}| \leq \boldsymbol{\Phi} = [\Phi_1, \ldots, \Phi_n]^T \in R^n$. Hence, the new dynamics can be rewritten as follows

$$\ddot{\mathbf{x}} = \mathbf{M}_o(\mathbf{x})^{-1}(\boldsymbol{\tau}(t) - \mathbf{C}_o(\mathbf{x}, \dot{\mathbf{x}})\dot{\mathbf{x}}) + \boldsymbol{\Psi} \tag{4}$$

For trajectory tracking, the error state is defined as

$$\mathbf{e} = \mathbf{x} - \mathbf{x_d} \tag{5}$$

where $\mathbf{x_d} \in R^n$ represents the desired position planning, and $\mathbf{e} = [e_1, \ldots, e_n]^T \in R^n$. The notation of the desired velocity state vector and the acceleration state vector are $\dot{\mathbf{x}}_\mathbf{d}$ and $\ddot{\mathbf{x}}_\mathbf{d}$, respectively.

In this research, we aim to design a robust controller for a 7-DOF robot manipulator (n = 7), which is shown in Figure 1. All seven joints are rotary joints. The dynamic identification was finished in previous research [17] with a result of $\mathbf{C}_o = diag[2.47\ 10.359\ 0.849\ 2.210\ 0.555\ 0.560\ 0.560] \in R^{7\times7}$, $\mathbf{M}_o = diag[0.213\ 1.159\ 0.171\ 0.328\ 0.0643\ 0.057\ 0.0640] \in R^{7\times7}$.

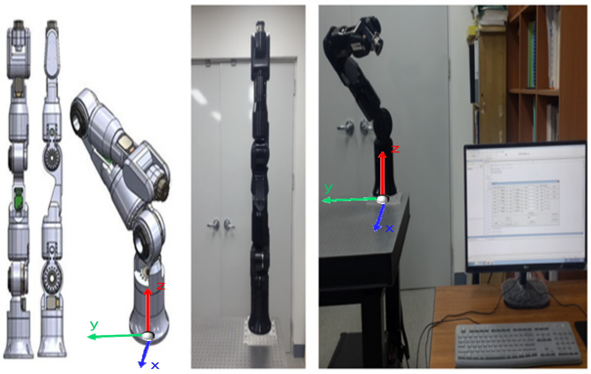


Figure 1. Seven-axis robot manipulator.

The goal of the proposed controller is to guarantee the error states to reach the sliding manifold and then slide to the equilibrium point (i.e., $\mathbf{e} = 0$) despite $\boldsymbol{\Psi}$.

2.2. Fractional Calculus

Fractional calculus defines the real number order for integral and derivative calculations. The Riemann–Liouville definition and the Caputo fractional derivative are adopted in this work.

Definition 1 [18]. *The α th-order Riemann–Liouville integral of function is given as*

$$D_{t_0,t}^{-\alpha} f(t) = \frac{1}{\Gamma(a)} \int_{t_0}^{t} (t-\tau)^{\alpha-1} f(t) d\tau,\ t > t_0 \tag{6}$$

where $\Gamma(\cdot)$ is the well-known Euler gamma function. t_0 is the initial time. The Riemann–Liouville integral has the semigroup property of $D_{t_0,t}^{-\alpha} D_{t_0,t}^{-\beta} = D_{t_0,t}^{-\beta-\alpha}$, $\forall \alpha, \beta > 0$. The Riemann–Liouville definition of the $\alpha \in R^+$ th-order derivative is

$$_{RL}D_{t_0,t}^{\alpha} f(t) = \frac{d^m}{dt^m} D_{t_0,t}^{-(m-\alpha)} f(t) = \frac{1}{\Gamma(m-a)} \frac{d^m}{dt^m} \int_{t_0}^{t} (t-\tau)^{m-\alpha-1} f(t) d\tau, t > t_0 \tag{7}$$

where $m-1 \le \alpha < m,\ m \in Z^+$. For convenience, the Riemann–Liouville derivative is abbreviated as $D_{t_0,t}^{\alpha}$ in this article.

Definition 2 [8]. *The $\alpha \in R^+$th-order Caputo fractional derivative on the half axis R^+ for a continuous function f(t) is defined as*

$$_{C}D_{t_0,t}^{\alpha} f(t) = D_{t_0,t}^{-(m-\alpha)} \frac{d^m}{dt^m}(f(t)) = \frac{1}{\Gamma(m-a)} \int_{t_0}^{t} \frac{f^{(m)}(\tau)}{(t-\tau)^{\alpha-m+1}} d\tau, t > t_0 \tag{8}$$

in which $m-1 \le \alpha < m,\ m \in Z^+$.

Property 1 [8]. *If the Riemann–Liouville derivative $D_{t_0,t}^{\beta} f(t)$ $(m-1 \le \beta < m, m \in Z^+)$ of a function $f(t)$ is integrable, then*

$$D_{t_0,t}^{-\alpha}(D_{t_0,t}^{\beta} f(t)) = D_{t_0,t}^{\beta-\alpha} f(t) - \sum_{j=1}^{n} [D_{t_0,t}^{\beta-j} f(t)]_{t=t_0} \frac{(t-t_0)^{\alpha-j}}{\Gamma(1+\alpha-j)} \tag{9}$$

Property 2 [18,19]. *The fractional-order Caputo fractional derivative has a relation to the Riemann–Liouville derivative.*

$$_{C}D_{0,t}^{\alpha} x(t) = D_{0,t}^{\alpha}(x(t) - \sum_{k=0}^{m-1} t^k x^{(k)}(0)/k!) \tag{10}$$

2.3. Stabilities

Lemma 1 [16,20]. *The stability analysis for fractional-order systems is given as*

$$_{C}D_{0,t}^{\alpha} x(t) = f(x(t))\ ,\ (0 < \alpha < 1,\ x(t) \in \mathbb{R}) \tag{11}$$

If the following condition is satisfied

$$x(t) f(x(t))\ < 0,\ \forall x \neq 0 \tag{12}$$

Then $x = 0$ is the equilibrium point, and the system (11) is asymptotically stable. Lemma 1 is proved by using the Lyapunov functions for fractional-order systems, which apply the fractional-order derivative to the Lyapunov function candidate as $_{C}D_{0,t}^{\alpha} V(t, x(t))$.

3. Controller Design

The controller FFOTSMC with SPO (FFOTSMCSPO) is proposed in this section. SPO is used to estimate the perturbation. Then, the estimated perturbation is transferred to the controller for compensation. The flow chart of the system and controller is shown in Figure 2. The usage of SPO is proposed first.

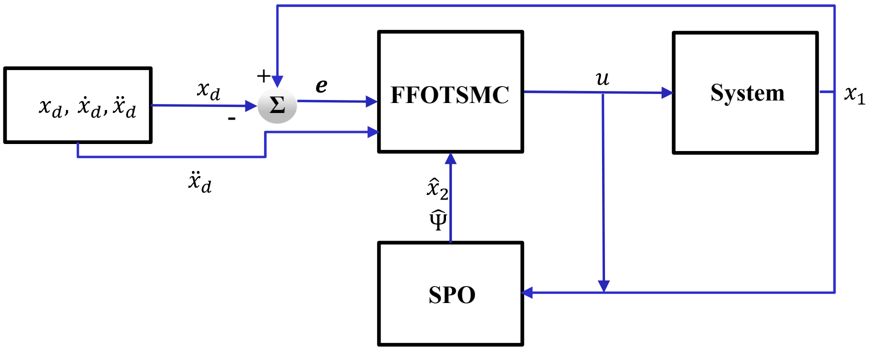


Figure 2. Block diagram of the controller.

3.1. SPO

SPO estimates partial unknown states and perturbation. It helps the controller to reduce the chattering against uncertainties and disturbances. It is efficient in estimating perturbation in low-frequency vibration [13]. For the motor-driven robot, the main component of the perturbation $\boldsymbol{\Psi}$ is the external disturbance $\mathbf{d}(t)$ and gravity $\mathbf{G}(\mathbf{x})$. The minor components include dynamic uncertainties and modeling errors.

The design of SPO is based on the state space. The state variables with a new state variable x_{3j} for the j-th $(j = 1, 2, \ldots n)$ joint of the robot can be represented as:

$$\begin{aligned} &\dot{x}_{1j} = x_{2j} \\ &\dot{x}_{2j} = \eta_j \overline{u}_j + \Psi_j \\ &\dot{x}_{3j} = \eta_j \dot{x}_{2j} - \dot{\Psi}_j / \eta_j \\ &y_j = x_{1j}. \end{aligned} \tag{13}$$

where η_j is a positive constant. $\overline{\mathbf{u}} = [\overline{u}_1, \ldots, \overline{u}_j, \ldots \overline{u}_n] \in R^n$ is the new equivalent control input to separate the perturbation from dynamics as follows:

$$\mathbf{M}_o(\mathbf{x})^{-1}(\boldsymbol{\tau}(t) - \mathbf{C}_o(\mathbf{x}, \dot{\mathbf{x}})\dot{\mathbf{x}}) = \boldsymbol{\eta}\overline{\mathbf{u}} \tag{14}$$

where $\boldsymbol{\eta} = [\eta_1, \ldots \eta_n]^T \in R^n$. The new state variable x_{3j} is defined as:

$$x_{3j} = \eta_j x_{2j} - \Psi_j / \eta_j. \tag{15}$$

The SPO for j-th joint is designed as follows:

$$\begin{aligned} &\dot{\hat{x}}_{1j} = \hat{x}_{2j} - g_{1j} sat(\tilde{x}_{1j}) \\ &\dot{\hat{x}}_{2j} = \eta_j \overline{u}_j - g_{2j} sat(\tilde{x}_{1j}) + \hat{\Psi}_j \\ &\dot{\hat{x}}_{3j} = \eta_j^2 (\overline{u}_j + \eta_j \hat{x}_{2j} - \hat{x}_{3j}) \\ &\hat{\psi}_j = \eta_j (\eta_j \hat{x}_{2j} - \hat{x}_{3_j}) \end{aligned} \tag{16}$$

where g_{1j} and g_{2j} are positive constants, and the notation $\hat{}$ means the estimation; $\sim$ means the estimation error, i.e., $\tilde{x}_{1j} = \hat{x}_{1j} - x_{1j}$; if $\left|\tilde{x}_{1j}\right| \geq \varepsilon_{0j}$, $\text{sat}(\tilde{x}_{1j}) = \tilde{x}_{1j} / \left|\tilde{x}_{1j}\right|$; else, $\text{sat}(\tilde{x}_{1j}) = \tilde{x}_{1j} / \varepsilon_{cj}$; ε_{0j} stands for the boundary layer of the estimation error.

The detailed stability analysis for the SPO has been proven in previous research [14]. Herein, we only use SPO to protect the controller against lesser uncertainties, which is reduced from Ψ_j to $\widetilde{\Psi}_j$. The perturbation estimation error has a relation to the real perturbation [13], as shown in a Laplace transform as:

$$\frac{\widetilde{\Psi}_j(p)}{\Psi_j(p)} = -\frac{p(p+g_{2j}/g_{1j})}{p^2+(g_{2j}/g_{1j})p+(g_{2j}/g_{1j})\eta_j^2} \tag{17}$$

where estimation error $\widetilde{\Psi}_j = \hat{\Psi}_j - \Psi_j$, and (17) shows a high-pass filter from the real perturbation to the perturbation estimation error. It signifies that the SPO is effective for the system with a low-frequency variability.

3.2. FFOTSMCSPO

The proposed controller has a fractional-order terminal sliding surface and a fractional-order type control law. It has faster convergence in both reaching mode and sliding mode, which can guarantee precise tracking.

3.2.1. Fractional-Order Sliding Surface

The fractional-order sliding surface of FFOTSMC is designed as:

$$\mathbf{s} = {}_cD^{\boldsymbol{\alpha}}_{0,t}\mathbf{e} + \boldsymbol{\lambda}_1 sig(\mathbf{e})^{\mathbf{r}_1} \tag{18}$$

where $\mathbf{s} = [s_1,\ldots,s_n]^T \in R^n$, $\boldsymbol{\lambda}_1 = diag(\lambda_{1j}) \in R^{n\times n}$, $(\lambda_{1j} > 0)$, ${}_cD^{\boldsymbol{\alpha}}_{0,t}\mathbf{e} = [{}_cD^{\alpha_1}_{0,t}e_1,\ldots,{}_cD^{\alpha_n}_{0,t}e_n]^T$, $sig(\mathbf{e})^{\mathbf{r}_1} \triangleq [sig(e_1)^{r_{1_1}},\ldots,sig(e_n)^{r_{1_n}}]^T$,$0 < r_{1_1},\ldots,r_{1_n} < 1$, and $sig(\cdot)^{r_{1n}} \triangleq sign(\cdot)|\cdot|^{r_{1n}}$. Differentiating it using Caputo fractional-order derivative with β-th order yields

$${}_cD^{\boldsymbol{\beta}}_{0,t}\mathbf{s} = {}_cD^{\boldsymbol{\beta}}_{0,t}\,{}_cD^{\boldsymbol{\alpha}}_{0,t}\mathbf{e} + {}_cD^{\boldsymbol{\beta}}_{0,t}(\boldsymbol{\lambda}_1 sig(\mathbf{e})^{\mathbf{r}_1}) \tag{19}$$

Depending on the definition of Caputo fractional derivative (8),

$${}_cD^{\boldsymbol{\beta}}_{0,t}\mathbf{s} = D^{\boldsymbol{\beta}-1}_{0,t}\frac{d}{dt}D^{\boldsymbol{\alpha}-1}_{0,t}\dot{\mathbf{e}} + {}_cD^{\boldsymbol{\beta}}_{0,t}(\boldsymbol{\lambda}_1 sig(\mathbf{e})^{\mathbf{r}_1}) \tag{20}$$

Applying the definition of Riemann–Liouville fractional derivative, we obtain

$$D^{\boldsymbol{\beta}-1}_{0,t}\frac{d}{dt}D^{\boldsymbol{\alpha}-1}_{0,t}\dot{\mathbf{e}} = D^{\boldsymbol{\beta}-1}_{0,t}D^{\boldsymbol{\alpha}}_{0,t}\dot{\mathbf{e}} \tag{21}$$

In practice, the trajectory tracking always commences with zero velocity error. Using Property 2 and assuming that the initial velocity error is zero.

$$D^{\boldsymbol{\beta}-1}_{0,t}D^{\boldsymbol{\alpha}}_{0,t}\dot{\mathbf{e}} = D^{\boldsymbol{\beta}-1}_{0,t}\,{}_CD^{\boldsymbol{\alpha}}_{0,t}\dot{\mathbf{e}} \tag{22}$$

Depending on the Definition 1 and 2, then

$$D^{\boldsymbol{\beta}-1}_{0,t}\,{}_CD^{\boldsymbol{\alpha}}_{0,t}\dot{\mathbf{e}} = D^{\boldsymbol{\beta}-1}_{0,t}D^{\alpha-1}_{t_0,t}\frac{d\dot{\mathbf{e}}}{dt} = D^{\boldsymbol{\beta}-1}_{0,t}D^{\alpha-1}_{t_0,t}\ddot{\mathbf{e}} \tag{23}$$

As the semigroup property of the fractional integral exists,

$$D^{\boldsymbol{\beta}-1}_{0,t}D^{\alpha-1}_{t_0,t}\ddot{\mathbf{e}} = D^{\boldsymbol{\beta}+\alpha-2}_{0,t}\ddot{\mathbf{e}} \tag{24}$$

It yields

$${}_cD^{\boldsymbol{\beta}}_{0,t}\mathbf{s} = D^{\boldsymbol{\alpha}+\boldsymbol{\beta}-2}_{0,t}\ddot{\mathbf{e}} + {}_cD^{\boldsymbol{\beta}}_{0,t}(\boldsymbol{\lambda}_1 sig(\mathbf{e})^{\mathbf{r}_1}) \tag{25}$$

3.2.2. Fast Fractional-Order Sliding Control/Reaching Law

The control law of the new proposed controller is designed based on the fast-TSM-type [3]. It applies the fractional-order derivative to build the desired dynamics of the sliding surface as

$$ {}_{c}D_{0,t}^{\boldsymbol{\beta}}\mathbf{s} = -\mathbf{k}_1\mathbf{s} - \mathbf{k}_2 sig(\mathbf{s})^{\mathbf{r}_2} - D_{0,t}^{\boldsymbol{\beta}+\boldsymbol{\alpha}-2}(\mathbf{k}_3 sgn(\mathbf{s}) + \widetilde{\boldsymbol{\Psi}}) \tag{26} $$

where $\boldsymbol{\beta} = [\beta_1, \ldots, \beta_n]^T \in R^n, (\beta_{j1} > 0.5), \mathbf{k}_1 = diag(k_{1j}) \in R^{n\times n}, (k_{1j} > 0), \mathbf{k}_2 = diag(k_{2j}) \in R^{n\times n}, (k_{2j} > 0)$, $sig(\mathbf{s})^{\mathbf{r}_2} \triangleq [sig(s_1)^{r_{2_1}}, \ldots, sig(s_n)^{r_{2_n}}]^T, 0 < r_{2_1}, \ldots, r_{2_n} < 1$ and $\mathbf{k}_3 = diag(k_{3j}) \in R^{n\times n}, (k_{3j} \geq \Omega_j)$. Ω_j is assumed as the boundary of perturbation estimation error $\widetilde{\Psi}_j$ for the j-th joint. $\widetilde{\Psi}_j$ can be estimated by using (17) with the assumption of real perturbation boundary Φ_j. The order (β_j and α_j) of the fractional derivative is always selected from 0.7 to 0.9. The influence of the selection of fractional-orders on the sliding surface design and the reaching law design is discussed in Appendix A.

Compared to the control law of previous research such as FOTSMC [9], in designing the control law, the desired dynamics of the sliding surface is formed after a first-order derivative, i.e., ($\beta_j = 1$ in (26)). In FFOTSMC, the fast control law ($\beta_j \in [0.5, 1)$ in (26)) adopts the fractional-order derivative. The fast control law results in a faster convergence speed to the sliding surface, which is the main contribution of this research.

Substituting $\ddot{\mathbf{e}}$ by $\ddot{\mathbf{x}} - \ddot{\mathbf{x}}_d$ and using the dynamics (14), the control input $\bar{\mathbf{u}}$ of FFOTSCM with SPO is given as (see Appendix B)

$$ \overline{\mathbf{u}}_{FFOTSMCSPO} = \boldsymbol{\eta}^{-1}[D_{0,t}^{2-\boldsymbol{\beta}-\boldsymbol{\alpha}}(-\mathbf{k}_1\mathbf{s} - \mathbf{k}_2 sig(\mathbf{s})^{\mathbf{r}_2}) - D_{0,t}^{1}D_{0,t}^{-\boldsymbol{\alpha}}D_{0,t}^{1}(\boldsymbol{\lambda}_1 sig(\mathbf{e})^{\mathbf{r}_1}) - \hat{\boldsymbol{\Psi}} + \ddot{\mathbf{x}}_d - \mathbf{k}_3 sgn(\mathbf{s})] \tag{27} $$

Remark 1. *The singularity of conventional TSMC is avoided by the fractional calculus, which is discussed in the numerical method (see Section 3.4) for implementing the fractional-order derivative.*

3.3. Stability Analysis

The asymptotic stability of reaching the sliding surface is proved by using Lemma 1 as follows. For the j-th joint, substituting the equivalent control input (27) to the dynamics (14) and (26), then applying $1 = sgn(s_j)sgn(s_j)$ yields (see Appendix B)

$$ \begin{aligned} & s_{jc}D_{0,t}^{\beta_j}s_j = -s_j(k_{1j}s_j + k_{2j}sig(s_j)^{r_{2_j}}) - s_j sgn(s_j)D_{0,t}^{\beta_j+\alpha_j-2}(k_{3j} + \widetilde{\Psi}_j sgn(s_j)) \\ & +D_{0,t}^{1-\beta_j-\alpha_j}[k_{1j}s_js_j + k_{2j}s_j sig(s_j)^{r_{2_j}}]_{t=0}\frac{t^{1-\beta_j-\alpha_j}}{\Gamma(2-\beta_j-\alpha_j)} \end{aligned} \tag{28} $$

As $k_{3j} \geq \Omega_j > |\widetilde{\Psi}_j|$

$$ D_{0,t}^{\beta_j+\alpha_j-2}(k_{3j} + \widetilde{\Psi}_j sgn(s_j)) > 0 \tag{29} $$

Additionally, for given parameters β_j and α_j, $D_{0,t}^{1-\beta_j-\alpha_j}[k_{1j}s_js_j + k_{2j}s_j sig(s_j)^{r_{2_j}}]_{t=0}$ is a positive constant. As $\frac{t^{1-\beta_j-\alpha_j}}{\Gamma(2-\beta_j-\alpha_j)}$ is a decreasing function of time t, $\frac{t^{1-\beta_j-\alpha_j}}{\Gamma(2-\beta_j-\alpha_j)} \approx 0$ when t is large. Hence, after a period of time t,

$$ \begin{aligned} & s_{jc}D_{0,t}^{\beta_j}s_j \approx -s_j(k_{1j}s_j + k_{2j}sig(s_j)^{r_{2_j}}) - s_j sgn(s_j)D_{0,t}^{\beta_j+\alpha_j-2}(k_{3j} + \widetilde{\Psi}_j sgn(s_j)) \\ & \leq -s_j(k_{1j}s_j + k_{2j}sig(s_j)^{r_{2_j}}) \leq 0,\ \forall s_j. \\ & s_{jc}D_{0,t}^{\beta_j}s_j < 0,\ \forall s_j \neq 0 \end{aligned} \tag{30} $$

Depending on Lemma 1, the sliding manifold s_j is asymptotically stable and can reach $s_j = 0$ from any initial condition.

After reaching the sliding surface (i.e., $s_j = 0$)

$$e_{jc}D_{0,t}^{\alpha_j}e_j = -\lambda_{1j}e_j sig(e_j)^{r_{1j}} \le 0 \tag{31}$$

$e_{jc}D_{0,t}^{\alpha_j}e_j = 0$ only if $e_j = 0$. It satisfies the asymptotically stable condition in Lemma 1.

Remark 2. *The asymptotic stability of the reaching mode and sliding mode is proved. However, the finite reaching time has not been obtained in this research. An explicit solution of a fractional differential equation with a nonlinear term cannot be solved, neither can an analysis solution. Despite this, in the tracking control of a robot manipulator, the fast convergence is more critical than the finite-time stability. It ensures an overall small tracking error. The fast convergence of FFOTSMC is discussed and obtained by analyzing the convergence characteristics of FOSMC and TSMC in Appendix A.*

3.4. Numerical Method

In order to implement the fractional-order derivative, the Grünwald–Letnikov fractional derivative is applied in this research. The Grünwald–Letnikov fractional derivative is derived from the limit definition for derivatives and equivalent to the Riemann–Liouville definition [18,19,21]. Thus, the equivalent control input of the sliding mode control (27), which uses the Riemann–Liouville derivative, was implemented using the Grünwald–Letnikov derivative. The Caputo derivative term in the sliding surface (18) was transferred to a Riemann–Liouville derivative with the initial error first using the properties mentioned. Then, it was implemented on the real system by the Grünwald–Letnikov derivative method. The α th-order Grünwald–Letnikov fractional derivative for $f(t)$ is

$$\begin{aligned} {}_{GL}D_{t_0,t}^{\alpha}f(t) &= \lim_{h\to 0} h^{-\alpha} \sum_{k=0}^{(t-t_0)/\mathrm{h}} \Gamma(k-a)\Gamma^{-1}(k+1)\Gamma^{-1}(-a)f(t-kh) \\ &\approx h^{-\alpha} \sum_{k=0}^{(t-t_0)/\mathrm{h}} \Gamma(k-a)\Gamma^{-1}(k+1)\Gamma^{-1}(-a)f(t-kh) \end{aligned} \tag{32}$$

where h is the sampling time. For the gamma function, the property is obtained as [18]

$$\Gamma(z+1)/\Gamma(z) = z, \forall z \in C \tag{33}$$

Thus, $\Gamma(k-a)/\Gamma(k+1)$ in (32) is rewritten as

$$\begin{aligned} \Gamma(k-a)/\Gamma(k+1) &= \frac{\Gamma(k-a)}{\Gamma(k-a-1)}\frac{\Gamma(k-a-1)}{\Gamma(k-a-2)}\frac{\Gamma(k-a-2)}{\Gamma(k-1)}\frac{\Gamma(k-1)}{\Gamma(k)}\frac{\Gamma(k)}{\Gamma(k+1)} \\ &= \frac{k-a-1}{k}\frac{k-a-2}{k-1}\frac{\Gamma(k-a-2)}{\Gamma(k-1)} = \frac{\Gamma(-a)}{\Gamma(1)}\prod_{n=1}^{k}\left(\frac{n-a-1}{n}\right) \end{aligned} \tag{34}$$

The result of the Grünwald–Letnikov fractional derivative can be calculated using (34). When k is large (large t), a considerable amount of calculation is needed. As the term $0 < (n-a-1)/n < 1$ for $n > 2$, $\Gamma(k-a)/\Gamma(k+1)$ keeps decreasing. $\Gamma(k-a)/\Gamma(k+1) \approx 0$ was assumed when $k \ge 50$. Thus, the fractional-order derivative was calculated in the real application as

$${}_{GL}D_{t_0,t}^{\alpha}f(t) \approx h^{-\alpha}\sum_{k=0}^{50}\Gamma(k-a)\Gamma^{-1}(k+1)\Gamma^{-1}(-a)f(t-kh) \tag{35}$$

Remark 3. *In the calculation of the fractional-order derivative with the numerical method, the singularity problem does not exist. Compared to the TSMC, the term in control input, such as $-D_{0,t}^{1}(\lambda_{1j}sig(e_j)^{r_{1j}})$, could have a singularity problem when $e_j = 0$ and $\dot{e}_j \ne 0$.*

4. Simulation and Experiment

4.1. Simulation

The designed controller, FFOTSMC with SPO (FFOTSMCSPO), was applied to a simple second-order SISO system simulated by MATLAB Simulink. It compared with four different controllers such as TSMC with SPO (TSMCSPO), LSMC with SPO (SMCSPO), FOSMC with SPO (FOSMCSPO) and FOTSMC with SPO (FOTSMCSPO). The advantages of using the fractional-order derivative and the terminal sliding surface were discussed via the simulation results. The dynamics of the second-order SISO system corresponding to the model (1) were defined as follows.

$$m\ddot{x} + c\dot{x} = \tau + d \tag{36}$$

where $m = 6.7\ \mathrm{kg}\cdot\mathrm{m}^2$, $c = 2400\ \mathrm{kg}\cdot\mathrm{m}^2\mathrm{s}^{-1}$ and $d = 10\sin(10t)\ \mathrm{N}\cdot\mathrm{m}$. The given dynamics were used to assume a hydraulic servo system, whose frictional coefficient term c has more impact than the moment of inertia m. The control inputs of all controllers were calculated using (27) with different parameters, as shown in Table 1. The simulation simulated as a regulator ($\mathbf{x_d} = \mathbf{0}$) with the initial state $\mathbf{x}_0 = [0, 0, 0]^{\mathbf{T}}$.

Table 1. Partial parameters of controllers.

Controller	β	α	r_1	r_2
SMCSPO	1	1	1	1
TSMCSPO	1	1	0.9	0.9
FOSMCSPO	1	0.7	1	1
FOTSMCSPO	1	0.7	0.9	0.9
FFOTSMCSPO	0.9	0.7	0.9	0.9

Where β determines whether the fractional-order derivative is used or not in the control law (27); α represents the fractional/integer-order of the derivative in the sliding surface design; when $\beta = 1$ $\alpha = 1$ it means the original first-order derivative. When selecting $\beta \in [0.5, 1)$ and $\alpha \in [0.5, 1)$, it means the fractional derivative is used. Selecting α and β closer to one, the fractional derivative is more similar to the first-order derivative. As the discussion in Appendix A shows, a smaller fractional-order (close to 0.5) results in a faster convergence speed in the beginning, leading to a short rising time in transient response; however, it causes a slow convergence in the terminal interval, resulting in a considerable settling time. r_1 is selected less than one when using the sliding surface of TSM; r_2 corresponding to the reaching law is associated with the fast control law in TSMC [3]. r_1 and r_2 can be viewed as terms to choose whether to approach linear control or discontinuous control between a linear control ($r_1 = 1$ or $r_2 = 1$) and a discontinuous one ($r_1 = 0$ or $r_2 = 0$). The discontinuous control input can generate faster convergence; however, the more discontinuous it is, the more chattering there is. The other parameters in (27) were selected as $\lambda_1 = \eta_1 = 10,\ \mathrm{k}_1 =\ 20\ \mathrm{k}_2 = 0.35$ and $\mathrm{k}_3 = 0.01$ in all five controllers. Large η_1 can help a more accurate estimating of SPO; however, it is limited by the sampling time [13]. The sliding surface slope λ_1 relates to the sliding manifold convergence speed, and it is also limited by the control input of the hardware limitation (maximum torque or voltage) in real applications. The reaching law gains (k_1, k_2 and k_3) are associated with the reaching speed from $s(t = 0) \neq 0$ to $s = 0$, and larger gains can generate a faster speed; however, chattering is also increased.

The simulation results for the SISO system are shown in Figure 3. Compared to SMCSPO, FOSMCSPO, and TSMCPO, FFOTSMCSPO exhibited the fewest errors, which means that both the fractional-order sliding surface and the TSM can generate a faster convergence. This was demonstrated by the result of the FOTSMCSPO, which applies the fractional-order derivative and terminal type in the sliding surface design. Compared to FOTSMCSPO, the proposed controller with a fast control law, FFOTSMCSPO exhibited a better tracking performance. Although an exact reaching time of

FFOTSMCSPO cannot be solved, the simulation results demonstrate that applying fractional-order to the control law design allowed the controller to have fast convergence.

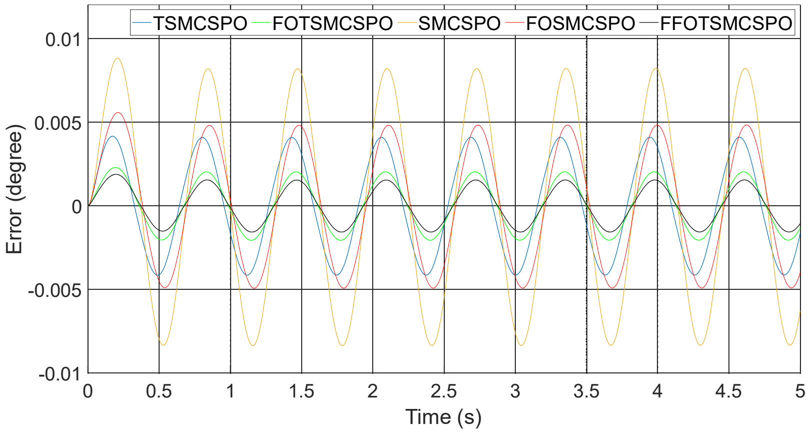


Figure 3. Simulation result.

4.2. Experiment

The experiments on the 7-DOF robot (Figure 1) compared the controller FFOTSMCSPO with FOTSMCSPO, TSMCSPO, and SMCSPO and FOSMCSPO under the same testing conditions for comparison. The dynamic modeling and identified parameters were obtained from previous research [17]. Control input $\boldsymbol{\tau}(t)$ was calculated by (27) and (14).

4.2.1. Setting

Figure 4 presents the configuration of the hardware and communication for the robot manipulator. The system uses a Windows operating system based PC for computation, IntervalZero RTX64 real-time extension to provide the real-time operation environment, two controller area network(CAN)-bus boards (each providing four channels) for the encoder data communication, and an eight-channel analog output board to provide the control input.

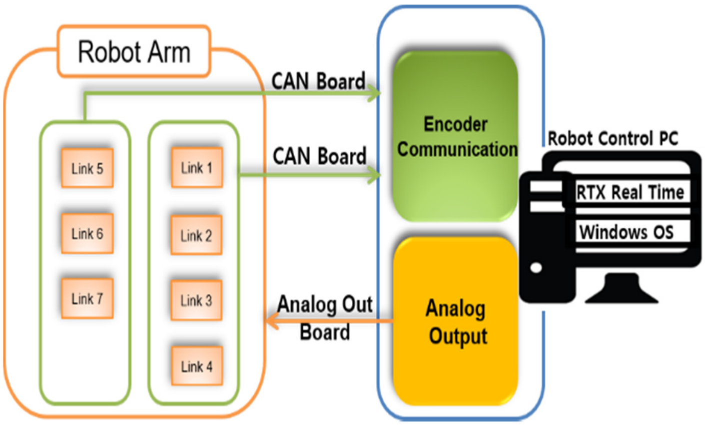
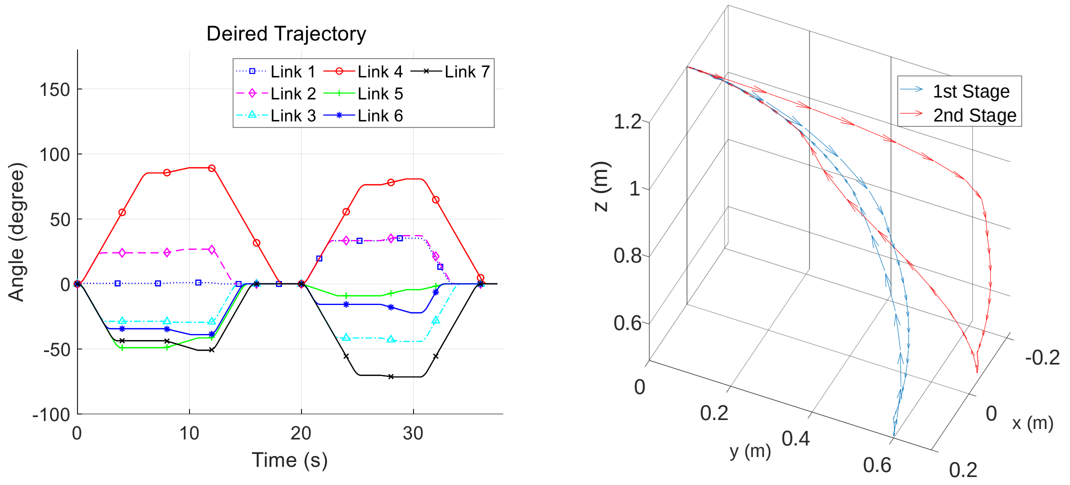


Figure 4. Robot arm configuration.

Considering the limited communication speed of CAN-Bus and the data volumes of seven encoders, the system sampling time for receiving the rotary encoder data and transmitting the control input was set to 2 ms. According to (A12) in previous research [13], the parameter η_j in SPO should be selected near 33.3. Thus, the controller parameters (27) for our proposed scheme are selected as $\lambda_{1j} = \eta_j = \{21, 14, 35.5, 20, 100, 34, 31\}$, other parameters $\text{k}_{1j} = \{10, 7, 8, 7.5, 8, 9.5, 10\}$, $\text{k}_{2j} = \{0.14, 0.15, 0.21, 0.165, 0.4, 0.25, 0.31\}$, $\text{k}_{3j} = \{0.01, 0.03, 0.01, 0.02, 0.01, 0.02, 0.1\}$, $\mathbf{r}_1 = \mathbf{r}_2 =$

$\{ 0.9, 0.9, 0.9\ , 0.89, 0.9, 0.89, 0.9 \}$, $\beta_j = \{ 0.9, 0.93, 0.92, 0.93, 0.9, 0.92, 0.9 \}$, where $j = 1, 2 \ldots 7$. For FOTSMCSPO, SMCSPO, TSMCSPOP, and FOSMCSPO, the values of β_j, α_j, r_{1j}, and r_{2j} are set as the same as the simulations, which determine whether the controllers use the fraction-order derivative and TSM or not (Table 1).

The desired trajectories are simulated assembly work on the end effector. They contain two stages of movement, the first stage simulated grabbing, picking up and homing; the second one simulated the movement that included moving to the assembled position and homing. The desired movement is shown in Figure 5a,b in joint space and the machine coordinate space (MCS), respectively (the base of the MCS is shown in Figure 1).



(**a**) Joint Space (**b**) Desired trajectory of end-effector

Figure 5. Desired trajectory of 7 links.

4.2.2. Experimental Results

The trajectory tracking errors of the fourth link in joint space and the end effector tracking errors on the Z-axis in the Cartesian coordinate are plotted in Figure 6. The right side of Figure 6 shows the magnified figures to clarify the tracking error results. The results of SMCSPO are not plotted because they have the largest errors and can affect the readability of the figure. The errors of other links (except for the fourth link) and the end effector errors in X- and Y-axes in the Cartesian coordinate are shown in Figure A3 in Appendix C.

The right side of Figure 6 shows that FFOTSMCSPO has the fewest tracking errors among the four controllers during the whole experiment interval. The controller of FOTSMCSPO using the fractional-order sliding surface showed fewer errors than TSMCSPO. This confirms that the fractional-order derivative improves control performance. The effect of TSMC can also be found by comparing the error result of FOTSMCSPO with FOSMCSPO. That is, it confirms that the FOTSMCSPO reduced more errors than FOSMCSPO.

The results in Figure 6 also show lesser chattering in all four controllers. This demonstrates that the approach of SPO in controllers is useful for reducing chattering and ensures the proposed controller will be applied in practice.

The numerical analysis of the experiment results is analyzed by comparing the mean absolute error (MAE). The MAE is calculated as

$$\frac{\sum\limits_{k=1}^{n} \mathrm{abs}(e(k))}{n}, n = 18750 \tag{37}$$

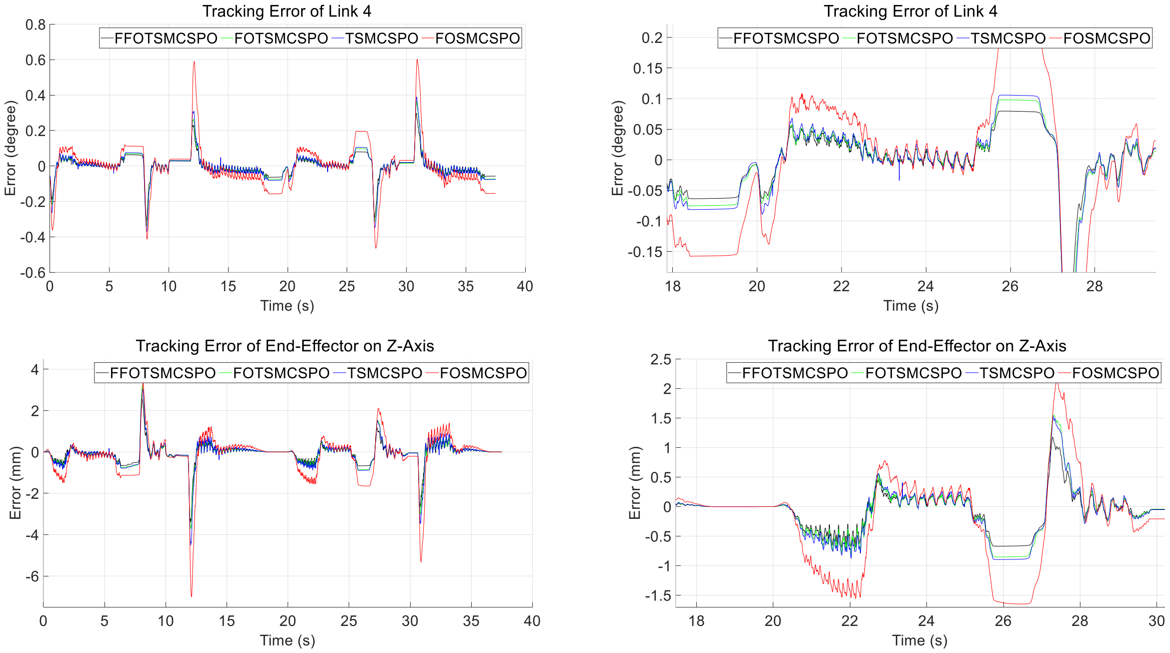


Figure 6. Tracking error results of each joint and end effector.

Table 2 shows the MAE results. In all seven links, the controller using fractional derivatives performed better with less MAE than the integer ones, FOTSMCSPO shows less MAE than TSMCSPO, and the results of FOSMCSPO are less than those using SMCSPO. Additionally, controllers using TSM show less errors than the conventional LSM ones, which is obtained from comparing FOTSMCSPO with FOSMCSPO, and TSMCSPO with SMCSPO. In the results of all five controllers, the new proposed controller FFOTSMCSPO, using the TSM and applying the fractional-order derivative in both the sliding surface design and the reaching law design, has the least MAE. It exhibits the fastest convergence and demonstrates the improvement. Comparing the results in FOSMCSPO with TSMCSPO, FOSMCSPO exhibits less MAE. This means that TSMCSPO is more efficient than FOSMCSPO.

Table 2. Mean absolute error.

Joint	FFOTSMCSPO *	FOTSMCSPO	TSMCSPO	FOSMCSPO	SMCSPO
1	0.02211	0.02430	0.02697	0.04297	0.04837
2	0.01668	0.02025	0.02178	0.04141	0.04412
3	0.02347	0.02650	0.02838	0.05033	0.05427
4	0.03689	0.04238	0.04600	0.08465	0.09360
5	0.06975	0.07661	0.08230	0.11177	0.12822
6	0.04323	0.04652	0.04934	0.08188	0.08902
7	0.03573	0.03634	0.04380	0.05619	0.06348
X-axis	0.14354	0.16122	0.17215	0.26177	0.29921
Y-axis	0.31273	0.36022	0.39767	0.77066	0.86858
Z-axis	0.26544	0.31577	0.34160	0.60392	0.66034

* The tracking error unit in the joint space and the Cartesian coordinate is the degree and mm, respectively.

5. Conclusions

The proposed controller, FFOTSMC, applies fractional calculus on both the terminal sliding surface design and the control/reaching law for reaching the sliding surface and maintaining stay on it. This guarantees fast convergence and high trajectory tracking performance. FFOTSMC is used with SPO for less chattering, resisting the modeling error and disturbance from the environment. The stability of the controller is analyzed based on the Lyapunov stability theory for fractional-order systems. Both the comparative simulation in MATLAB and experiments on the seven links robot manipulator verified the excellent tracking performance of the proposed controller. The experimental results demonstrate the feasibility of the proposed controller, which shows a small error and less chattering against the high gravity effect in dynamics and modeling errors.

Even the finite-time stability of FFOTSMC (reaching the sliding surface and the equilibrium in finite time) cannot be proved in this research. The experimental results and simulation results have demonstrated the outstanding performance in tracking and improvement by applying the fractional-order derivative on the control/reaching law design. The finite-time stability problem will be investigated in the future.

Author Contributions: Conceptualization, J.W. and M.C.L.; methodology, J.W.; validation, J.W., J.H.K. and H.H.K.; writing—original draft, J.W.; writing—review and editing, J.W. and M.C.L.; supervision, M.C.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Smart Factory Development Support Program (20005055, Development of Integrated Control Technology supporting both control of four robots and PLCopen standard-based programmable logic controller(PLC) languages (five types) for SME's smart factory) and the Technology Innovation Program (10073147, Development of Robot Manipulation Technology by Using Artificial Intelligence) funded by the Ministry of Trade, Industry and Energy (MOTIE, Korea).

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

The reason why FOTSMC provides benefits in terms of fast convergence is discussed here. Considering the sliding surface of FOSMC as

$$s = {}_{c}D^{r}_{0,t}e(t) - Ae(t), r \in (0,1], A \in R^{-} \tag{A1}$$

where $e(t)$ is required to stabilize at the equilibrium point. When selecting the derivative order $r = 1$, (A1) is a simple first-order LSM sliding surface. After reaching the sliding surface (i.e., s $= 0$), the solved solution in the form of the Mittag-Leffler function is [22]

$$e(t) = E_{r,1}(t)e_0 = e_0\sum_{k=0}^{\infty}\frac{A^k t^{rk}}{\Gamma(rk+1)} \tag{A2}$$

where $e_0 = e(t = 0)$. When $r = 1$, the solution corresponds to the first integer-order LSM. Its solution is an exponential function as

$$e(t) = E_{1,1}(t)e_0 = e_0\sum_{k=0}^{\infty}\frac{A^k t^{k}}{\Gamma(k+1)} = e_0\sum_{k=0}^{\infty}\frac{A^k t^{k}}{k!} = e_0 Exp(At) \tag{A3}$$

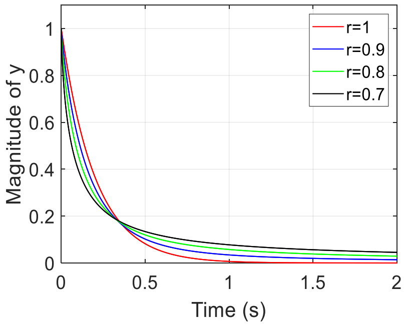


Figure A1. Solution of the fractional-order differential Equation (A1) where (0 < r < 1), e_0 = 1 and A = −5.

Figure A1 shows the result of the time response of FOSMC after reaching the sliding surface. The fractional-order one ($0 < r < 1$) has a faster decay than integer-order ($r = 1$) in the interval at the beginning (transient response); however, the convergence near the steady-state becomes slow. In the beginning, fast convergence is observed. It has a more significant effect on trajectory tracking, which can reduce the relatively large error fast. It explains why the fractional-order derivative allows for a better control performance in SMC for robot manipulator trajectory tracking. However, drawbacks of FOSMC, such as a slow convergence in the terminal interval (i.e., long settling time), exist. In addition, the smaller r is associated with faster convergence in the transient response part; however, extremely small r would result in a considerable settling time. Hence, selecting the order of the derivative is a tradeoff problem.

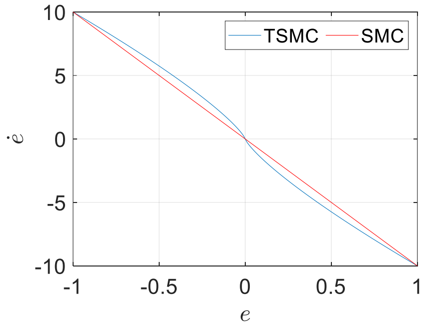


Figure A2. $\dot{e}$ corresponding with e.

In order to compensate for the slow convergence in the interval of small values in FOSMC, TSM control is expected to be used because the nonlinear sliding surface in TSM affects convergence speed more efficiently in the interval of a small error. Considering a TSM sliding surface is

$$s = \dot{e} + \lambda sign(e)|e|^{\beta} \tag{A4}$$

where $\lambda > 0$ and $0 < \beta < 1$. After reaching the sliding surface, it yields

$$\frac{|\dot{e}|}{|e|} = \lambda |e|^{r-1} \tag{A5}$$

The e in TSM (i.e., $\beta \in (0,1)$) generates larger $|\dot{e}|$ than using LSM (i.e., $\beta = 1$). Moreover, the smaller e results in a larger gradient, as shown in Figure A2. This signifies that the TSM has fast convergence benefits, especially in the small error field. This is useful to compensate for the drawbacks of FOSMC.

However, in the case of using TSM in FOSMC, FOTSMC, an explicit solution of the sliding surface, such as FOSMC shown in (A2), cannot be solved because of the nonlinear term in the TSM sliding surface. Thus, the finite time for reaching the sliding surface and the equilibrium has not been solved in this research.

Appendix B

The derivation of (27) and (28) are shown in this section. Using (25) and (26), we obtain

$$-\mathbf{k}_1\mathbf{s} - \mathbf{k}_2 sig(\mathbf{s})^{\mathbf{r}_2} - D_{0,t}^{\boldsymbol{\beta}+\boldsymbol{\alpha}-2}(\mathbf{k}_3 sgn(\mathbf{s}) + \widetilde{\Psi}) = D_{0,t}^{\boldsymbol{\alpha}+\boldsymbol{\beta}-2}\ddot{\mathbf{e}} + {}_{c}D_{0,t}^{\boldsymbol{\beta}}(\boldsymbol{\lambda}_1 sig(\mathbf{e})^{\mathbf{r}_1}) \tag{A6}$$

As $\ddot{\mathbf{e}} = \ddot{\mathbf{x}} - \ddot{\mathbf{x}}_d = \boldsymbol{\eta}\overline{\mathbf{u}} + \boldsymbol{\Psi} - \ddot{\mathbf{x}}_d$, it yields

$$-\mathbf{k}_1\mathbf{s} - \mathbf{k}_2 sig(\mathbf{s})^{\mathbf{r}_2} - D_{0,t}^{\boldsymbol{\beta}+\boldsymbol{\alpha}-2}(\mathbf{k}_3 sgn(\mathbf{s}) + \widetilde{\Psi}) = D_{0,t}^{\boldsymbol{\alpha}+\boldsymbol{\beta}-2}(\boldsymbol{\eta}\overline{\mathbf{u}} + \boldsymbol{\Psi}) + {}_{c}D_{0,t}^{\boldsymbol{\beta}}(\boldsymbol{\lambda}_1 sig(\mathbf{e})^{\mathbf{r}_1}) \tag{A7}$$

As the semigroup property of the Riemann–Liouville derivative is limited, such as $D_{0,t}^{2-\alpha-\beta} D_{0,t}^{\alpha+\beta-2} x(t) = x(t) \neq D_{0,t}^{\alpha+\beta-2} D_{0,t}^{2-\alpha-\beta} x(t)$ [8], taking the fractional derivative $D_{0,t}^{2-\alpha-\beta}$ on both sides of (A7) yields

$$D_{0,t}^{2-\alpha-\beta}(-\mathbf{k}_1\mathbf{s} - \mathbf{k}_2 sig(\mathbf{s})^{\mathbf{r}_2}) - \mathbf{k}_3 sgn(\mathbf{s}) - \widetilde{\mathbf{\Psi}} = \boldsymbol{\eta}\bar{\mathbf{u}} + \mathbf{\Psi} + D_{0,t}^{2-\alpha-\beta}\,{}_cD_{0,t}^{\beta}(\boldsymbol{\lambda}_1 sig(\mathbf{e})^{\mathbf{r}_1}) \tag{A8}$$

As ${}_cD_{0,t}^{\beta}(\boldsymbol{\lambda}_1 sig(\mathbf{e})^{\mathbf{r}_1}) = D_{0,t}^{-\beta}\frac{d}{dt}(\boldsymbol{\lambda}_1 sig(\mathbf{e})^{\mathbf{r}_1})$ and $\widetilde{\mathbf{\Psi}} = \hat{\mathbf{\Psi}} - \mathbf{\Psi}$, by substituting them for (A8), (27) is derived.

However, due to the limited semigroup property of the Riemann–Liouville derivative, using the equivalent control input (27) for the real dynamics of the sliding surface (25) cannot obtain the desired dynamics (26). This can be found after substituting (27) for (25) as

$${}_cD_{0,t}^{\beta}\mathbf{s} = D_{0,t}^{\alpha+\beta-2}[D_{0,t}^{2-\beta-\alpha}(-\mathbf{k}_1\mathbf{s} - \mathbf{k}_2 sig(\mathbf{s})^{\mathbf{r}_2}) - D_{0,t}^{1} D_{0,t}^{-\alpha} D_{0,t}^{1}(\boldsymbol{\lambda}_1 sig(\mathbf{e})^{\mathbf{r}_1}) - \widetilde{\mathbf{\Psi}} - \mathbf{k}_3 sgn(\mathbf{s})] + {}_cD_{0,t}^{\beta}(\boldsymbol{\lambda}_1 sig(\mathbf{e})^{\mathbf{r}_1}) \tag{A9}$$

The term $D_{0,t}^{\alpha+\beta-2} D_{0,t}^{2-\beta-\alpha}(-\mathbf{k}_1\mathbf{s} - \mathbf{k}_2 sig(\mathbf{s})^{\mathbf{r}_2})$ can be rewritten as follows, using Property 1.

$$D_{0,t}^{\alpha+\beta-2} D_{0,t}^{2-\beta-\alpha}(-\mathbf{k}_1\mathbf{s} - \mathbf{k}_2 sig(\mathbf{s})^{\mathbf{r}_2}) = -\mathbf{k}_1\mathbf{s} - \mathbf{k}_2 sig(\mathbf{s})^{\mathbf{r}_2} + D_{0,t}^{1-\beta-\alpha}[\mathbf{k}_1\mathbf{s} + \mathbf{k}_2 sig(\mathbf{s})^{\mathbf{r}_2}]_{t=0}\frac{t^{1-\beta-\alpha}}{\Gamma(2-\beta-\alpha)} \tag{A10}$$

With the assumption that the initial error of velocity and acceleration is zero, i.e., $\ddot{\mathbf{e}}(t=0) = \dot{\mathbf{e}}(t=0) = 0$, the term $D_{0,t}^{\alpha+\beta-2}[D_{0,t}^{1} D_{0,t}^{-\alpha} D_{0,t}^{1}(\boldsymbol{\lambda}_1 sig(\mathbf{e})^{\mathbf{r}_1})]$ in (A9) is simplified as

$$D_{0,t}^{\alpha+\beta-2} D_{0,t}^{1} D_{0,t}^{-\alpha} D_{0,t}^{1}(\boldsymbol{\lambda}_1 sig(\mathbf{e})^{\mathbf{r}_1}) = D_{0,t}^{\alpha+\beta-2}\,{}_cD_{0,t}^{1-a} D_{0,t}^{1}(\boldsymbol{\lambda}_1 sig(\mathbf{e})^{\mathbf{r}_1}) = D_{0,t}^{\alpha+\beta-2} D_{0,t}^{-a} D_{0,t}^{1} D_{0,t}^{1}(\boldsymbol{\lambda}_1 sig(\mathbf{e})^{\mathbf{r}_1}) \tag{A11}$$

As the semigroup property of the Riemann–Liouville integral exists, by using the Definition 2, (A11) is derived as

$$D_{0,t}^{\alpha+\beta-2} D_{0,t}^{-a} D_{0,t}^{1} D_{0,t}^{1}(\boldsymbol{\lambda}_1 sig(\mathbf{e})^{\mathbf{r}_1}) = D_{0,t}^{\beta-2} D_{0,t}^{1} D_{0,t}^{1}(\boldsymbol{\lambda}_1 sig(\mathbf{e})^{\mathbf{r}_1}) = {}_cD_{0,t}^{\beta}(\boldsymbol{\lambda}_1 sig(\mathbf{e})^{\mathbf{r}_1}) \tag{A12}$$

Thus, substituting (A10) and (A12) for (A9), the dynamics of the sliding surface for the *j*-th element

$$\begin{aligned} {}_cD_{0,t}^{\beta_j} s_j &= -(k_{1j}s_j + k_{2j}sig(s_j)^{r_{2j}}) - sgn(s_j) D_{0,t}^{\beta_j+\alpha_j-2}(k_{3j} + \widetilde{\Psi}_j sgn(s_j)) \\ &+ D_{0,t}^{1-\beta_j-\alpha_j}[k_{1j}s_j + k_{2j}sig(s_j)^{r_{2j}}]_{t=0}\frac{t^{1-\beta_j-\alpha_j}}{\Gamma(2-\beta_j-\alpha_j)} \end{aligned} \tag{A13}$$

are derived.

Appendix C

Partial results of experiments are shown here.

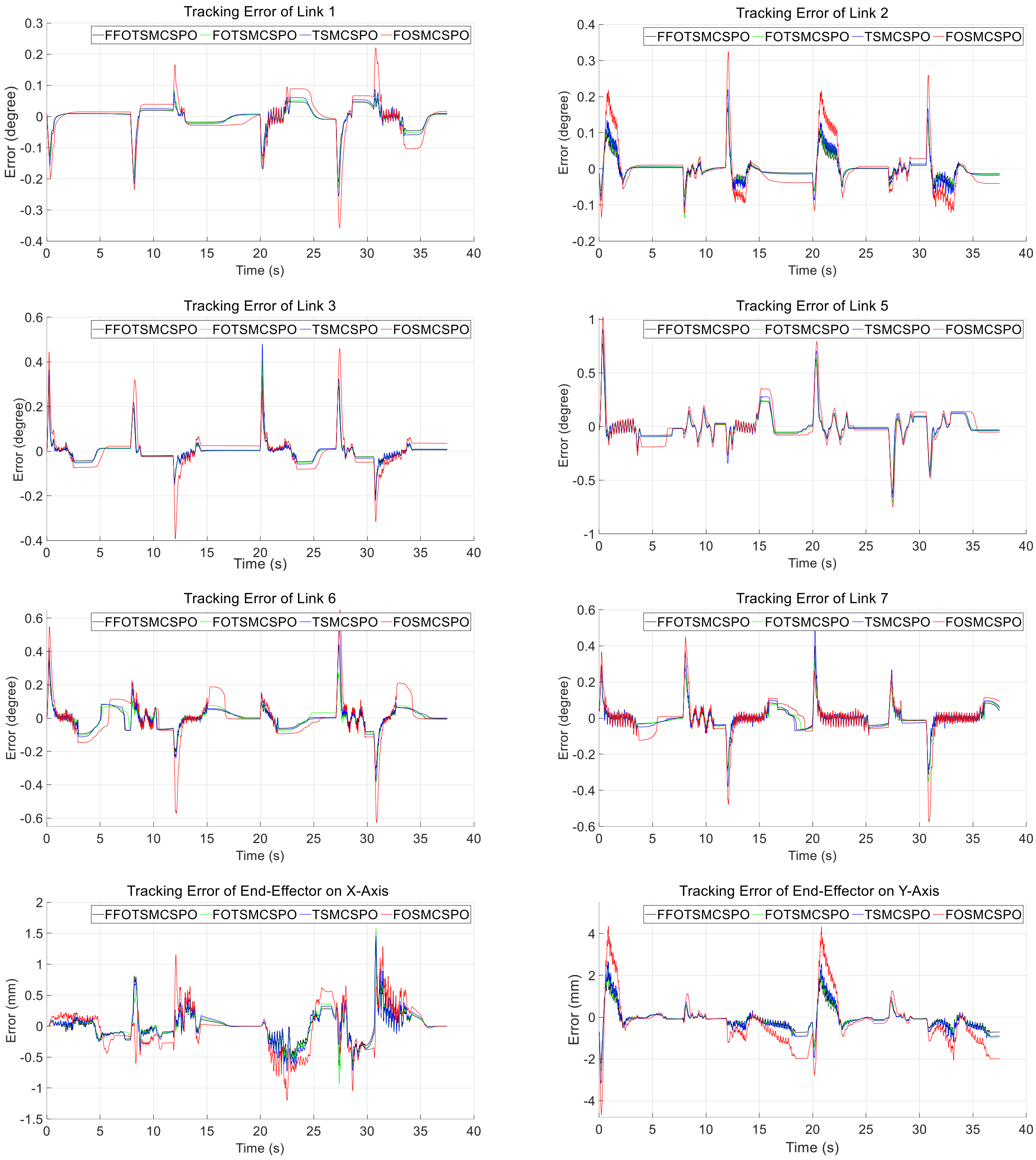


Figure A3. Tracking error results of each joint and end effector.

References

1. Young, K.D.; Utkin, V.I.; Ozguner, U. A control engineer's guide to sliding mode control. *IEEE Trans. Control. Syst. Technol.* **1999**, *7*, 328–342. [CrossRef]
2. Zhihong, M.; Yu, X.H. Terminal sliding mode control of MIMO linear systems. In Proceedings of the 35th IEEE Conference on Decision and Control, Kobe, Japan, 13 December 1996; Volume 4, pp. 4619–4624.
3. Yu, S.; Yu, X.; Shirinzadeh, B.; Man, Z. Continuous finite-time control for robotic manipulators with terminal sliding mode. *Automatica* **2005**, *41*, 1957–1964. [CrossRef]
4. Feng, Y.; Yu, X.; Man, Z. Non-singular terminal sliding mode control of rigid manipulators. *Automatica* **2002**, *38*, 2159–2167. [CrossRef]
5. Feng, Y.; Yu, X.; Han, F. On nonsingular terminal sliding-mode control of nonlinear systems. *Automatica* **2013**, *49*, 1715–1722. [CrossRef]
6. Dadras, S.; Momeni, H.R. Fractional terminal sliding mode control design for a class of dynamical systems with uncertainty. *Commun. Nonlinear Sci. Numer. Simul.* **2012**, *17*, 367–377. [CrossRef]

7. Zhang, B.T.; Pi, Y.G.; Luo, Y. Fractional order sliding-mode control based on parameters auto-tuning for velocity control of permanent magnet synchronous motor. *ISA Trans.* **2012**, *51*, 649–656. [CrossRef] [PubMed]
8. Kilbas, A.A.A.; Srivastava, H.M.; Trujillo, J.J. *Theory and Applications of Fractional Differential Equations*; Elsevier Science Limited: Amsterdam, The Netherlands, 2006.
9. Wang, Y.; Luo, G.; Gu, L.; Li, X. Fractional-order nonsingular terminal sliding mode control of hydraulic manipulators using time delay estimation. *J. Vib. Control* **2016**, *22*, 3998–4011. [CrossRef]
10. Sun, G.; Ma, Z.; Yu, J. Discrete-time fractional order terminal sliding mode tracking control for linear motor. *IEEE Trans. Ind. Electron.* **2017**, *65*, 3386–3394. [CrossRef]
11. Wang, Y.; Gu, L.; Xu, Y.; Cao, X. Practical tracking control of robot manipulators with continuous fractional-order nonsingular terminal sliding mode. *IEEE Trans. Ind. Electron.* **2016**, *63*, 6194–6204. [CrossRef]
12. Jie, W.; Yudong, Z.; Bao, Y.; Kim, H.H.; Lee, M.C. Trajectory tracking control using fractional-order terminal sliding mode control with sliding perturbation observer for a 7-DOF robot manipulator. *IEEE/ASME Trans. Mechatron.* **2020**, *25*, 1886–1893. [CrossRef]
13. Moura, J.T.; Elmali, H.; Olgac, N. Sliding mode control with sliding perturbation observer. *J. Dyn. Syst. Meas. Control* **1997**, *119*, 657–665. [CrossRef]
14. Wang, J.; Lee, M.C.; Kallu, K.D.; Abbasi, S.J.; Ahn, S. Trajectory tracking control of a hydraulic system using tsmcspo based on sliding perturbation observer. *Appl. Sci.* **2019**, *9*, 1455. [CrossRef]
15. Lee, M.C.; Kim, C.Y.; Yao, B.; Peine, W.J.; Song, Y.E. Reaction force estimation of surgical robot instrument using perturbation observer with SMCSPO algorithm. In Proceedings of the 2010 IEEE/ASME International Conference on Advanced Intelligent Mechatronics, Montreal, QC, Canada, 6–9 July 2010; pp. 181–186.
16. Aguila-Camacho, N.; Duarte-Mermoud, M.A.; Gallegos, J.A. Lyapunov functions for fractional order systems. *Commun. Nonlinear Sci. Numer. Simul.* **2014**, *19*, 2951–2957. [CrossRef]
17. Jie, W.; Jamshed, S.; Kim, D.J.; Yulong, B.; Lee, M.C. Trajectory tracking control of a 7-axis robot arm using SMCSPO. In Proceedings of the 2019 International Conference on Intelligent Robotics and Applications, Shenyang, China, 8–11 August 2019; Springer: Cham, Switzerland, 2019; pp. 701–708.
18. Podlubny, I. *Fractional Differential Equations: An Introduction to Fractional Derivatives, Fractional Differential Equations, to Methods of Their Solution and Some of Their Applications*; Elsevier: Amsterdam, The Netherlands, 1998.
19. Li, C.; Qian, D.; Chen, Y. On Riemann–Liouville and caputo derivatives. *Discret. Dyn. Nat. Soc.* **2011**, *2011*. [CrossRef]
20. Deng, W.; Li, C.; Lü, J. Stability analysis of linear fractional differential system with multiple time delays. *Nonlinear Dyn.* **2007**, *48*, 409–416. [CrossRef]
21. Scherer, R.; Kalla, S.L.; Tang, Y.; Huang, J. The Grünwald–Letnikov method for fractional differential equations. *Comput. Math. Appl.* **2011**, *62*, 902–917. [CrossRef]
22. Li, Y.; Chen, Y.Q.; Podlubny, I. Stability of fractional-order nonlinear dynamic systems: Lyapunov direct method and generalized Mittag–Leffler stability. *Comput. Math. Appl.* **2010**, *59*, 1810–1821. [CrossRef]

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.