Research Article

Robust Model-Free Control for Robot Manipulator under Actuator Dynamics

Dorsaf Elleuch and Tarak Damak

University of Sfax, National Engineering School of Sfax (ENIS), Laboratory of Sciences and Techniques of Automatic Control & Computer Engineering (Lab-STA), Tunisia B. P. 1173, 3038 Sfax, Tunisia

Correspondence should be addressed to Dorsaf Elleuch; dorsafelleuch@yahoo.fr

Received 26 April 2020; Revised 12 July 2020; Accepted 17 July 2020; Published 3 August 2020

Guest Editor: Cuimei Jiang

Copyright © 2020 Dorsaf Elleuch and Tarak Damak. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

An intelligent proportional-derivative sliding mode controller (i-PDSMC) is presented to overcome the unmodeled complexity of the robot manipulator under an actuator. i-PDSMC is a free model intelligent control based on the ultralocal, sliding mode, and PD control structure. A stability condition is determined by the Lyapunov theory. A comparative study between a classical PD, an intelligent PD control, and i-PDSMC is done through a robot manipulator under actuators. The simulation results prove that the proposed controller is more robust to trajectory tracking under parameter variations and external disturbances.

1. Introduction

Disturbance and uncertainty are the major problems of the robot manipulator control. In the literature, many model-based controls have shown a robustness to overcome uncertainty such as the sliding mode control, but the chattering phenomenon is still its major problem, or the backstepping control, which is limited for strict feedback nonlinear systems. Researchers take more interest in PD control because of its rapidity and its steady-state tracking error which is imposed.

Because of the sliding mode control robustness, in [1–4], the authors have developed a sliding mode control in which the equivalent control term is the PD control and the sliding surface is of the PD form. In this controller, named PD-SMC, only a measured feedback is considered. The proposed PD-SMC applied to a robot manipulator proved a better performance than the classical SMC in reducing tracking errors. In [1], a comparative study is done between PD, SMC, and PD-SMC in which the last controller showed the best performance under parameter variations. However, the PD-SMC parameters must be carefully chosen due to the significant and complicate effect on the tracking error. In [4], the authors proposed an adaptive PD-SMC (APD-SMC) scheme in which the discontinuous term is obtained by an adaptation law. The APD-SMC performances are tested and validated through a 4 DOF SCARA robotic manipulator.

Formerly, the model complication was removed, and a nonmodel or free model control was introduced. This control, named model-free control (MFC), does not need a model to determine the control law. It is based on an ultralocal model, a PID control, and a compensate term. The ultralocal model is online estimated using input and output measurements. This MFC is called an intelligent PID (i-PID) control. In [5, 6], the authors prove that i-PID is robust to trajectory tracking than the classical PID. Also, the intelligent PI (i-PI) [7, 8] and the intelligent PD (i-PD) [9–11] are studied in the literature. Both controllers, which are validated on complex nonlinear systems, show the best performance compared to other different model-based controls. The advantages of MFC have attracted many researchers to develop a new control form. For that, a partial MFC in which the control law depends on a part of
the model (input matrix) and the controller gains are
optimized using a linear quadratic regulator (LQR) is
proposed in [12]. The partial MFC performances are
proved through a helicopter system. In [13–15], the
authors have developed a robust MFC based on the sliding
mode technique for a multilink flexible robot-based energy
relationship of the system.

Sometimes a control-based observer or state estima-
tion is named as the model-free control [16, 17]. But, this
method still depends totally or partially on the model
structure to build an estimator or an observer. In [16], the
authors proposed a model-free filtered backstepping
control for marine power systems. The estimated dynamic
model converges slowly to its real value. In [17], a terminal
sliding mode controller combined with a nonlinear dis-
turbance observer is proposed. The observer is based on
the model dynamic system part, and the control results did
not have a well performance.

A model-free sliding mode controller can be a more
attractive approach. In [18, 19], the control law was designed
using only the measured state and the previous control
input. A nonlinear second-order system is considered to
show the proposed controller performance under measured
noise and parameter variations. These results were extended
to multi-input multioutput systems [20, 21]. The MFC
sliding mode control (MFC-SMC) is developed based on a PI
control. The designed control is validated on a twin-rotor
aerodynamic system (TRAS), and it is implemented as two
SISO control loops. The corresponding simulation and
experimental results show a good performance of the MFC-
SMC compared to an i-PI controller.

To overcome the control problem of robot manipulators
under unmodeled actuator dynamics and control com-
plexity, many controller algorithms are proposed in the
literature. In [22], the authors proposed an intelligent
control which is compared to a PD and a sliding mode
controller. The proposed control did not prove a well per-
formance. Besides, it is difficult to im-
plement the intelligent control.

To benefit the i-PD and SMC advantages, an i-PDSMC is
proposed in this paper. A basic SMC form is used when the 1st
SMC term is an i-PD controller and the 2nd term is a classical
discontinuous form. The proposed controller is compared to
a classical PD controller and an i-PD controller, and robustness
under parameter variations and external disturbance is vali-
dated using a robot manipulator under actuator dynamics. The
rest of the paper is organized as follows.

Firstly, a robot manipulator under actuator dynamics is
presented. Next, a brief description of a classical PD control is
introduced. In Section 4, a model-free control is presented, and
i-PD and i-PDSM control laws are defined. i-PDSM stability
to the trajectory tracking problem is proved. In the last section,
the three previous controllers are applied to the robot ma-
nipulator, and simulation results under parameter variation and
external disturbance are presented.

1.1. Robot Manipulator under Actuator Dynamics. In
Lagrange form, an n–link interconnected to an n robot
manipulator model is expressed as follows [22]:

\[ M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + N = \tau, \]

in which \(M(q)\) is the inertia matrix, \(C(q, \dot{q})\) is the centripetal
and Coriolis forces, \(G(q)\) is the gravity vector, \(N\) is the
external disturbance \(\tau + \text{friction term } f(q)\) + unmodeled
dynamics, \(\tau\) is the torque control, \(q\) is the vector of the joint
position, \(\dot{q}\) is the joint velocity, and \(\ddot{q}\) is the acceleration
vector.

Recently, an actuator dynamic related to the robot ma-
nipulator model effect is considered in many interesting re-
search studies. Considering an armature DC-servo motor as
an actuator in each joint, it can be expressed as follows:

\[ \tau_e = K_T I_a, \]

\[ \tau_e = J_a \dot{\theta}_m + B_m \dot{\theta}_m + \tau_m, \]

\[ u = R_a I_a + L_a \dot{I}_a + K_E \dot{\theta}_m. \]

The gear ratio \(g_r\), relating the joint position \(q\) and a
motor shaft position \(\theta_m\), is expressed as

\[ g_r = \frac{\theta_m}{\tau} = \frac{\tau}{q}, \]

where \(\tau\) is the electromagnetic torque, \(K_T\) is the diagonal
matrix of motor torque, \(I_a\) is the armature currents, \(J_m\) is
the diagonal matrix of the moment inertia, \(B_m\) is the
diagonal matrix of torsional damping coefficients, \(\tau_m\) is the
load torque vector, \(R_a\) is the diagonal matrix of armature
resistance, \(L_a\) is the diagonal matrix of armature inductance,
\(K_E\) is the diagonal matrix of the feedback electromotive
force (EMF) coefficients, \(\theta_m\) is the vector of motor shaft
position, \(\dot{\theta}_m\) is the velocity of the motor shaft, and \(\dot{\theta}_m\)
is the acceleration of the motor shaft.

Considering (2), (3), and (5), the armature input voltage
could be rewritten as

\[ u = L_n \ddot{\theta} + R_n \dot{\theta} + J_n \dot{\theta} + (R_n I_n + L_n B_m) \ddot{\theta} + (R_i B_m + K_{Em}) \dot{\theta}, \]

where \(L_n = L_a (g_r K_T)^{-1}\), \(R_a = R_a (g_r K_T)^{-1}\), \(J_n = g_r^2 J_m, \)
\(B_n = g_r^2 B_m\), and \(K_{Em} = K_E g_r^2\).

Including the actuator dynamic expression (6), the
model expression (1) will be rewritten as

\[ M^* \ddot{\theta} + D(\theta, \dot{\theta}, \ddot{\theta}) + d = u. \]

Noting

\[ M(q) = M_n + M_\theta(q), \]

\[ M^* = L_n [M_n + J_n], \]
Without the differential equation model, a model-free steady state performance is decreased. and improves the rise time and oscillation effect, but the controller is a fast controller, besides its derivative action, widely employed to the robotic control, the PD control is 

2. Classical PD Controller

Widely employed to the robotic control, the PD control is robust to trajectory tracking. In a closed loop system, the PD controller is a fast controller, besides its derivative action, and improves the rise time and oscillation effect, but the steady state performance is decreased.

A PD controller has the following form:

\[ \tau = K_p e + K_d \dot{e}, \]  
(12)

where \( e \) is the tracking error defined as \( y - y_d \).

The PD controller inconvenience is the vibration problem caused by a measured noise due its high control gain. To improve controller performance, recently, many researchers have shown interest in an intelligent PD controller.

3. Model-Free Control

Without the differential equation model, a model-free control is a nonlinear control developed recently in [1, 4]. Then, an ultra-local model replaces the unknown mathematical model:

\[ y^{(\theta)} = F + au, \]  
(13)

where

(i) \( \alpha \in \mathbb{R} \) is a constant parameter chosen arbitrary such that \( y^{(\theta)} \) and \( au \) have the same magnitude.

(ii) \( y^{(\theta)} \) is the derivative of \( y \) at \( \theta \) order, where \( \theta \geq 1 \). In practice, \( \theta \) is chosen equal to 1 or 2 and does not have connection with the system order.

(iii) \( F \) is the unknown plant and any uncertainty, and it is estimated in real time through the input and output information measured system behavior.

The estimate of \( F \) is defined as follows:

\[ \tilde{F} = \tilde{y}^{(\theta)} - \alpha \tilde{u}, \]  
(14)

where

(i) \( \tilde{y}^{(\theta)} \) is the estimate of \( y^{(\theta)} \). Diverse estimation methods are used to estimate \( y^{(\theta)} \), like algebraic methods [4], a low-pass filter. To avoid the algebraic loop problem, \( \tilde{y}^{(\theta)} \) is generated by a first-order derivative plus the low-pass filter with the transfer function:

\[ H_{lpf} = \left( \frac{K_{lpf} s}{T_{lpf} s + 1} \right)^\theta. \]  
(15)

(ii) \( \tilde{u} \) is the estimate of \( u \). It is chosen as a past value of \( u \) to avoid algebraic loops in the controllers such that \( \tilde{u} = u(t - 1) \).

Then, the model-free controller is as follows:

\[ u = \frac{1}{\alpha} \left( \tilde{y}^{(\theta)} - F - \Lambda e^{-\xi \zeta} \right), \]  
(16)

in which \( \Lambda \) is a function set to \( \mathbb{R}^{+} \rightarrow \mathbb{R} \). The dynamical error described by \( e^{(\theta)} = \Lambda (e^{-\xi \zeta}) \) is asymptotically stable.

Remark 1. \( e^{-\xi \zeta} = (\xi, e, \xi, \xi, e, \xi, \xi, e, \xi, \xi, e, \xi, \xi, e, \xi, \xi, e, \xi, \xi, e) \); \( \xi, \zeta \in [0, \theta]; \int e \) is the \( k \) integral term.

3.1. Intelligent PD (i-PD). The derivative order \( \theta \) is generally equal to 1 or 2, yielding the intelligent PID (i-PID).

If \( \theta = 2 \),

\[ u = \frac{1}{\alpha} \left( \tilde{y}^{(2)} - \tilde{F} - K_p e - K_i \int e - K_D \dot{e} \right). \]  
(17)

Then, if \( K_i = 0 \), the controller becomes an i-PD controller, and the control input of the feedback is expressed as

\[ u = \frac{1}{\alpha} \left( \tilde{y}^{(2)} - \tilde{F} - K_p e - K_D \dot{e} \right), \]  
(18)

where \( K_p \) and \( K_D \) are the proportional and derivative gains.

3.2. Intelligent i-PD Sliding Mode Control (i-PDSMC). A general classical sliding mode control form is

\[ u_{SM} = u_{eq} + u_{dis}, \]  
(19)

where \( u_{eq} \) depends on the system mathematical model and \( u_{dis} \) is a discontinuous control.

In this study, an i-PD sliding mode control is introduced by choosing the equivalent control \( u_{eq} \) such that

\[ u_{eq} = \frac{1}{\alpha} \left( \tilde{y}^{(2)} - \tilde{F} - K_p e - K_D \dot{e} \right). \]  
(20)

The system model (7) is written as an interconnected SISO nonlinear system. Then, the discontinuous control is considered as follows:

\[ u_{dis} = -KS - \mu \text{sign}(S), \]  
(21)

in which \( S \) is the sliding surface verifying Filippov theory, and it is chosen as

\[ S = \dot{e} + \lambda e. \]  
(22)

i-PDSMC has the following form:
Figure 1: Continued.
Figure 1: Block diagram of the i-PDSMC. (a) Block diagram of the i-PDSMC system. (b) Block diagram of the i-PDSMC structure. (c) Block diagram of the i-PDSMC discontinuous term.

Figure 2: Continued.
in which \( \text{sat}(S, \varepsilon) \) is defined as:

\[
\text{sat}(S, \varepsilon) = \begin{cases} 
-1 & \text{if } S < -\varepsilon \\
S/\varepsilon & \text{if } |S| \leq \varepsilon \\
1 & \text{if } S > \varepsilon 
\end{cases}
\]

and \( \varepsilon > 0 \) is the boundary layer thickness.

**Theorem 1.** The i-PDSM control (23) is stable if and only if the control gains satisfy the following equalities:

\[
K_p > 0; \quad \lambda > K_D; \quad \alpha \neq 0; \quad k > - (\mu/\varepsilon); \quad \forall \alpha, \mu > 0; \quad x_2 > (K_p x_1)/(\lambda - K_D).
\]

3.2.1. i-PDSMC Stability Analysis. Introducing a state variable error such as

\[
x_1 = \varepsilon, \\
x_2 = \dot{\varepsilon}.
\]  

The sliding surface is rewritten using the new state variables:

\[
S = x_2 + \lambda x_1.
\]

The sliding surface is rewritten using the new state variables:

\[
\begin{aligned}
q_1 &\text{d} \\
q_1 &\text{PD} \\
q_1 &\text{i-PD} \\
q_1 &\text{i-PDSM}
\end{aligned}
\]

\[
\begin{aligned}
q_2 &\text{d} \\
q_2 &\text{PD} \\
q_2 &\text{i-PD} \\
q_2 &\text{i-PDSM}
\end{aligned}
\]

\[
\begin{aligned}
e_1 &\text{PD} \\
e_1 &\text{i-PD} \\
e_1 &\text{i-PDSM}
\end{aligned}
\]

\[
\begin{aligned}
e_2 &\text{PD} \\
e_2 &\text{i-PD} \\
e_2 &\text{i-PDSM}
\end{aligned}
\]

Figure 2: 1st case: nominal situation. (a) The control input at the nominal situation. (b) Tracking trajectory and error tracking trajectory to \( q_1 \) and \( q_2 \) at the nominal situation.

\[
u_{SM} = \frac{1}{\alpha}(\dot{y}_d - F - K_p \varepsilon - K_D \dot{\varepsilon}) - KS - \mu \text{sat}(S, \varepsilon),
\]  

\[
\begin{aligned}
x_1 &= \varepsilon, \\
x_2 &= \dot{\varepsilon}.
\end{aligned}
\]  

The sliding surface is rewritten using the new state variables:

\[
S = x_2 + \lambda x_1.
\]

So, \( \dot{x}_2 = \dot{y} - \dot{y}_d \).

An estimate error \( e_{\text{est}} \) is considered such that

\[
e_{\text{est}} = F - F = \dot{y} - \dot{\tilde{y}} = \dot{y}_d - \dot{\tilde{y}}_d.
\]

Let \( \theta = 2 \), then

\[
\ddot{y} = F + au_{SM}.
\]  

Introducing (23) in (27), we have
Figure 3: 2nd case: presence of parameter variation and external disturbance. (a) The control input: parameter variation and an external disturbance is injected. (b) Tracking trajectory $q_1$ and $q_2$: parameter variation and an external disturbance is injected.
Figure 4: 3rd case: presence of parameter variation, external disturbance, and friction force. (a) The control input: parameter variation and an external disturbance is injected adding a friction force at \( t = 7s \). (b) Tracking trajectory \( q_1 \) and \( q_2 \): parameter variation and an external disturbance is injected adding a friction force at \( t = 7s \).
\[ \dot{y} = F + a \left( \frac{1}{\alpha} \left( \ddot{y}_d - \ddot{F} - K_p e - K_D \dot{e} \right) - K_S - \mu \text{sat}(S, \varepsilon) \right). \]  

(28)

So, \[ \dot{x}_2 = \dot{y} - \ddot{y}_d = F - \ddot{F} - (\ddot{y}_d - \ddot{y}_d) - K_p e - K_D \dot{e} - \alpha \right), \]

(29)

The sliding surface derivative is \( \dot{S} = \dot{x}_2 + \lambda x_2 = \dot{x}_2 + \lambda x_2. \)

Then, \( \dot{S} = -K_p x_1 - K_D x_2 - \alpha \right) + \lambda x_2. \)

Considering the following Lyapunov function:

\[ V = \frac{1}{2} |S|^2. \]  

(30)

Its derivative is

\[ \dot{V} = SS', \]

\[ \dot{V} = S (-K_p x_1 - K_D x_2 - \alpha \right) + \lambda x_2). \]  

(31)

If \( |S| \leq \varepsilon, \) then

\[ \dot{V} \leq S \left(-K_p x_1 - K_D x_2 - \alpha \right) + \lambda x_2), \]

\[ \Rightarrow \dot{V} \leq -a \left( k + \frac{\mu}{\varepsilon} \right) S^2, \]

\[ -K_p x_1 - (K_D - \lambda) x_2 = 0. \]  

(33)

Equation (33) is implying that \( x_1(\varepsilon) = e^{-(K_p/\lambda - K_D/\mu)} \).

When \( t \rightarrow \infty, \) \( x_1 \rightarrow 0 \) if only if \( (K_p/\lambda - K_D/\mu) \).

Then, \( K_p > 0 \) and \( \lambda > K_D. \)

To guarantee stability, the term \( \alpha (k + (\mu/\varepsilon)) \) is positive.

For that, \( \alpha \) must be different to zeros and \( k > - (\mu/\varepsilon). \)

If \( |S| > \varepsilon, \) then the Lyapunov derivative function becomes

\[ \dot{V} \leq S \left(-K_p x_1 - K_D x_2 - \alpha KS - a \mu + \lambda x_2 \right), \]

\[ \Rightarrow \dot{V} \leq -aKS^2. \]  

(34)

And, \( -K_p x_1 - K_D x_2 - a \mu + \lambda x_2 = 0; \) then, \( K_p x_1 = (\lambda - K_D)x_2 - a \mu; \)

\[ \Rightarrow x_2 = \frac{K_p x_1 + a \mu}{\lambda - K_D}. \]  

(35)

Then, \( \lambda \neq K_D \) and \( \forall \alpha, \mu > 0, \) \( x_2 > (K_p x_1)/\lambda - K_D. \)

3.3. Simulation Results. To verify the robustness, the previously defined controls are applied to a robot manipulator under actuator dynamics described in (7). The current voltage \( u \) is the control input.

The parameters of the robot manipulator are considered as follows:

\[ J_{m1} = 3.7 \times 10^{-5} \text{ kg} \cdot \text{m}^2, \]

\[ J_{m2} = 1.47 \times 10^{-4} \text{ kg} \cdot \text{m}^2, \]

\[ R_{a1} = 2.8 \Omega, \]

\[ R_{a2} = 4.8 \Omega, \]

\[ l_1 = 205 \text{ mm}, \]

\[ l_2 = 210 \text{ mm}, \]

\[ l_1 = 154.8 \text{ mm}, \]

\[ l_2 = 105 \text{ mm}, \]

\[ m_1 = 3.55 \text{ kg}, \]

\[ m_2 = 0.75 \text{ kg}, \]

\[ B_{m1} = 1.3 \times 10^{-5} \text{ N} \cdot \text{m/s}, \]

\[ B_{m2} = 2 \times 10^{-5} \text{ N} \cdot \text{m/s}, \]

\[ K_{T1} = 0.21 \text{ N} \cdot \text{m/A}, \]

\[ K_{T2} = 0.23 \text{ N} \cdot \text{m/A}, \]

\[ I_{a1} = 3 \text{ mH}, \]

\[ I_{a2} = 2.4 \text{ mH}, \]

\[ K_{E1} = 2.42 \times 10^{-4} \text{s/rad} \cdot \text{V}, \]

\[ K_{E2} = 2.18 \times 10^{-4} \text{s/rad} \cdot \text{V}, \]

\[ g = 9.8 \text{ m/s}^2, \]

\[ g_{t1} = 60, \]

\[ g_{t2} = 30. \]

The control robustness is studied under the following three cases:

(i) 1st case: at nominal situation (without disturbance \( N = 0)\)

(ii) 2nd case: a parameter variation appears at \( t = 4s \) in which 1kg is added to link 2 masses \( (m_2 \) becomes 0.75kg + 1kg), and an external disturbance \( N = \tau = [5 \sin (5t) 0.5 \sin (5t)]^T \) is occurred at \( t = 7s \)

(iii) 3rd case: added to the 2nd case, a friction force is injected at \( t = 7s, \) having the following form:

\[ f(\dot{q}) = [20q_1 + 1.6\text{sgn}(q_1) + 4q_2 + 3.2\text{sgn}(q_2)]^T. \]  

(37)

Then, the external disturbance becomes \( N = \tau + f(\dot{q}). \)

3.3.1. The Control Simulation Parameters. Error tracking trajectory: \( e = y - y_d, \) where \( y = [q_1 q_2]^T: \)

(i) PD control: \( u_{pd} = K_a e + K_b \dot{e} \)

\[ K_a = \begin{bmatrix} 70 & 0 \\ 0 & 50 \end{bmatrix}, K_b = \begin{bmatrix} 7 & 0 \\ 0 & 1 \end{bmatrix}. \]
(ii) i-PD control: \( u_{ipd} = (1/\alpha)(-\tilde{F} + \tilde{y}_d - K_pe - K_D\dot{e}) \),
where \( \tilde{F} = \tilde{y} - au(t - 1); \) \( \tilde{y} = ((K_{ipd}s)/T_{ipd}s + 1)^2 y \);
\( \dot{y}_d = ((K_{ipd}s)/T_{ipd}s + 1)^2 y_d \);
\( K_p = \begin{bmatrix} 15 & 0 \\ 0 & 20 \end{bmatrix} ; K_D = \begin{bmatrix} 3 & 0 \\ 0 & 2 \end{bmatrix} ; K_{ipd} = \begin{bmatrix} 0.6 \\ 0.6 \end{bmatrix} ; \lambda = \begin{bmatrix} 0.002 \\ 0.002 \end{bmatrix} ; \alpha = \begin{bmatrix} 90 \\ 100 \end{bmatrix} . \)

(iii) i-PDSM control: \( u_{ipd} = (1/\alpha)(-\tilde{F} + \tilde{y}_d - K_pe - K_D\dot{e}) - (KS + \mu s)(S) \),
where \( S = \dot{e} + \lambda e; \) \( \tilde{y} = ((K_{ipd}s)/T_{ipd}s + 1)^2 y \);
\( \dot{y}_d = ((K_{ipd}s)/T_{ipd}s + 1)^2 y_d \);
\( K_p = \begin{bmatrix} 15 & 0 \\ 0 & 20 \end{bmatrix} ; K_D = \begin{bmatrix} 3 & 0 \\ 0 & 2 \end{bmatrix} ; K_{ipd} = \begin{bmatrix} 0.6 \\ 0.6 \end{bmatrix} ; \alpha = \begin{bmatrix} 0.002 \\ 0.002 \end{bmatrix} ; \lambda = \begin{bmatrix} 0.04 \\ 0.004 \end{bmatrix} ; K = \begin{bmatrix} 0.2 \\ 0.2 \end{bmatrix} . \)

The block diagram of the i-PDSMC is presented in Figure 1. All controller terms are developed in Figures 1(a)–1(c). The control law is implemented using the MATLAB program.

In the first case, the simulation results show that the PD control has an important impact at the control effort \( u_1 \) in the initial instant, but, in the control effort \( u_2 \), the i-PDSM control has a higher impulse than PD and i-PD controllers (Figure 2(a)).

The PD and i-PD controls have a transition phase smoother than the i-PDSM one. In the steady state, all the controllers have the same performances. But, the i-PDSM control has a lower error to tracking trajectory than the others (Figure 2(b)).

For the 2\(^{nd}\) case, the PD, i-PD, and i-PDSM controllers keep their control effort robustness (Figure 3(a)).

At \( t = 7s \), i-PDSMC is more robust to tracking trajectory, and this is due to the sliding mode capability to reject uncertainty and disturbance; however, the PD and i-PD controllers lose their performance (Figure 3(b)).

In the 3\(^{rd}\) case, the PD and i-PD controls lose their performances to tracking trajectory under external disturbance, while the i-PDSM control is still robust in the steady state (Figure 4). To tracking trajectory, the i-PDSM control converges more rapidly than the other controllers.

4. Conclusion

In this work, an intelligent sliding mode controller is defined based on an intelligent PD controller as an equivalent control term and a classical discontinuous control term. It is called as i-PDSMC. The stability of i-PDSMC is studied, and a sufficient condition to guarantee tracking trajectory robustness is determined. The proposed controller is compared to a classical PD and an intelligent PD (i-PD) controller. These controllers are applied to a robot manipulator under actuators to test their performances in the presence of parameter variations and external disturbances. The simulation results prove that the proposed i-PDSMC is robust than PD and i-PD controllers under uncertainties.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] P. R. Ouyang, J. Acob, and V. Pano, “PD with sliding mode control for trajectory tracking of robotic system,” Robotics and Computer-Integrated Manufacturing, vol. 30, no. 2, pp. 189–200, 2014.
[2] P. R. Ouyang, J. Tang, W. H. Yue, and S. Jayasinghe, “Adaptive PD plus sliding mode control for robotic manipulator,” in Proceedings of the 2016 IEEE International Conference on Advanced Intelligent Mechatronics (AIM), Banff, Canada, July 2016.
[3] J. Sangrok, B. Jeongae, K. Jongwon, S. Taewon, and S. K. Hwa, “Switching PD-based sliding mode control for hovering of a tilting-thruster under water robot,” PLoS One, vol. 13, no. 3, Article ID e0194427, 2018.
[4] T. Kara and A. H. Mary, “Adaptive PD-SMC for nonlinear robotic manipulator tracking control,” Studies in Informatics and Control, vol. 26, no. 1, 2017.
[5] M. Fliss and C. Join, “Model-free control and intelligent PID controllers: towards a possible Trivialization of nonlinear control?” in Proceedings of the 15th IFAC Symposium on System Identification, Saint-Malo, France, July 2009.
[6] M. Fliss and C. Join, “Model-free control,” International Journal of Control, Taylor & Francis, vol. 86, no. 12, pp. 2228–2252, 2013.
[7] F. Lafont, N. Pessel, J. F. Balmat, and M. Fliss, “On the model-free control of an experimental greenhouse,” in Proceedings of the World Congress on Engineering and Computer Science, WCECS 2013, pp. 23–25, San Francisco, CA, USA, October 2013.
[8] Y. Liu, W. Yan, D. Xu, W. Yang, and W. Zhang, “Direct torque control of PMSM based on model free iPI controller,” in Proceedings of the 7th Data Driven Control and Learning Systems Conference, Enshi, China, May 2018.
[9] M. Ticherfatine and Z. Qidan, “Model-free approach based on intelligent PD controller for vertical motion reduction in fast ferries,” Turkish Journal of Electrical Engineering & Computer Sciences, vol. 26, pp. 393–406, 2018.
[10] H. Wang, X. Ye, Y. Tian, and N. Christov, “Attitude control of a quadrotor using model free based sliding mode controller,” in Proceedings of the 20th International Conference on Control Systems and Sciences, Bucharest, Romania, May 2015.
[11] M. Ticherfatine and Z. Qidan, “Fast ferry smoothing motion via intelligent PD controller,” Journal of Marine Science and Application, vol. 17, no. 2, pp. 273–279, 2018.
[12] Y. Xin, Z. C. Qin, W. G. Wu, and J. Q. Sun, “Partial model-free control of a 2-input and 2-output helicopter system,” in Proceedings of the ENOC, Budapest, Hungary, June 2017.
[13] T. H. Lee, S. S. Ge, and Z. P. Wang, “Adaptive robust controller design for multi-link flexible robots,” Mechatronics, vol. 11, no. 8, pp. 951–967, 2001.
[14] S. S. Ge, T. H. Lee, and G. Zhu, “Energy-based robust controller design for multi-link flexible robots,” Mechatronics, vol. 6, no. 7, pp. 779–798, 1996.
[15] C. K. Pang, G. Dai, T. H. Lee, and M. Nagashima, "Adaptive non-model-based vibration control of critical flexible modes in Mechatronic systems," in *Proceedings of the 18th World Congress of the International Federation of Automatic Control*, Milano, Italy, August 2011.

[16] H. Zhou, D. Xu, and B. Jiang, "Model free command filtered backstepping control for marine power systems," *Mathematical Problems in Engineering*, vol. 2015, Article ID 619430, 8 pages, 2015.

[17] X. Liu, H. Yu, J. Yu, and L. Zhao, "Combined speed and current terminal sliding mode control with nonlinear disturbance observer for PMSM drive," *IEEE Access*, vol. 6, pp. 29594–29601, 2018.

[18] A. Crassidis and A. Mizov, "A model-free control algorithm derived using the sliding mode control method," in *Proceedings of the 2nd International Conference of Control, Dynamic Systems, and Robotics*, Ottawa, Canada, May 2015.

[19] A. Crassidis and R. M. Reis, "Model-free sliding mode control method," in *Proceedings of the 3rd International Conference on Control, Dynamic Systems, and Robotics (CDSR’16)*, Ottawa, Canada, May 2016.

[20] R.-E. Precup, M.-B. Radac, R.-C. Roman, and E. M. Petriu, "Model-free sliding mode control of nonlinear systems: algorithms and experiments," *Information Sciences*, vol. 381, pp. 176–192, 2017.

[21] R. M. Reis, "A new model-free sliding mode control method with estimation of control input error," Rochester Institute of Technology, Rochester, NY, USA, 2016.

[22] R.-J. Wai and P.-C. Chen, "Robust neural fuzzy network control for robot manipulator including actuator dynamics," *IEEE Transactions on Industrial Electronics*, vol. 53, no. 4, pp. 1328–1349, 2006.