Article

Proportional-Integral Sliding Mode Control with an Application in the Balance Control of a Two-Wheel Vehicle System

Chien-Hong Lin ¹ and Fu-Yuen Hsiao ^{2,*}

¹ Aeronautical Systems Research Division, National Chung-Shan Institute of Science and Technology, Taichung 40722, Taiwan; jhlin412@gmail.com

² Department of Aerospace Engineering, Tamkang University, New Taipei 25137, Taiwan

* Correspondence: fyhsiao@mail.tku.edu.tw; Tel.: +886-2-26215656 (ext. 3319)

Received: 29 May 2020; Accepted: 22 July 2020; Published: 24 July 2020

Abstract: A balance control method, the proportional-integral sliding mode control (PISMC), is proposed to control the tilt attitude of an experimental two-wheel vehicle system (TWVS). Based on our previous work of implementing a generalized PISMC to control a linearized dynamical system, this paper extends the algorithm to a wider range: First, the control design of a weighted-control system is proposed. Secondly, our algorithm was realized and verified in a TWVS using its original nonlinear model. Thirdly, a systematical way to tune parameters are presented. The robustness of the proposed algorithm is also discussed in this paper. The simulation results of this work validate that the PISMC has better robustness to counteract the external disturbances than the conventional sliding mode control (SMC) does. Additionally, the experimental results show that the PISMC is capable of autonomously balancing the TWVS more effectively than the conventional SMC. The successful implementation of our algorithm potentially extends the implementation of the PISMC to various nonlinear and emerging systems.

Keywords: proportional-integral sliding mode control; sliding mode control; two-wheel vehicle system; nonlinear system

1. Introduction

This paper presents a controller called the proportional-integral sliding mode controller (PISMC). The sliding mode control (SMC) has been widely accepted as an efficient method for the tracking or balance control of an uncertain nonlinear system. The capability of achieving perfect performance in principle is also demonstrated in the presence of parameter uncertainties and bounded external input disturbances. The proposed PISMC combines the proportional sliding mode control (PSMC) and the integral sliding mode control (ISMC). There are two main contributions in this paper. First, the PISMC provides more parameters with which to tune the controller. As a consequence, it reduces adverse effects due to the external disturbances effectively. Second, this paper sets up an experiment as an example to demonstrate the tracking accuracy in a nonlinear system. This example also shows how this algorithm can be implemented to a physical system. Though the name PISMC may sound similar to some algorithms regarding the PID sliding surface, the definition of sliding surface and the mathematical expression of PISMC are different.

The main structure of SMC theory was brought out by Utkin, a former USSR scholar, in 1977 [1]. After the original theory, various applications and extensions of the theory were proposed and presented. Some of the examples are given and reviewed in the following. In 1994, Sira-Ramirez et al. [2] proposed a dynamic multivariable discontinuous feedback control strategy

of the sliding mode type for the altitude stabilization of a nonlinear helicopter model in vertical flight. In 1996, Utkin and Jingxin [3] proposed a new sliding mode design concept named integral sliding mode control (ISMC). ISMC is different from the conventional sliding mode design approaches, and the only difference is the integral form of the error term in place of the original error term. In 2000, Baik et al. [4] designed an ISMC which has one separate boundary surface in order to decrease the chattering and improve the controllers' responses. In 2003, Lin et. al [5] proposed a PID sliding surface, where the sliding mode motion occurs with global invariance. In the same year, Hess and Wells [6] employed the SMC for flight control design. In 2004, Rafimanzelat and Yazdanpanah [7] investigated the low chattering SMC for balance control design. In 2005, Niu and Ho et al. [8] applied the robust integral sliding mode control to uncertain stochastic systems. Besides that, Waslander et al. applied the design of ISMC to a multi-agent quadrotor test-bed. Moreover, Bouabdallah, et al. also applied the back-stepping and sliding-mode techniques to an indoor micro-quad-rotor. Koshkouei et al. [9] in 2005 proposed a dynamic SMC which increases accuracy under high-frequency switching of control. More studies regarding the applications of sliding mode technique and its extensions are discussed in [10–13]. In 2009, Soylu et al. [14] studied the control of an underwater remote vehicle-manipulator system using sliding-mode control and H^∞ control. This paper showed that the sliding-mode and H^∞ controller combined approach provides superior dynamic coupling reduction performance.

Speaking of PISMC, Nawawi, Sam, and Osman have devoted themselves to the investigation of PISMC [15–18] since 2002. In 2011, Lin et al. applied the PISMC to stabilize the attitude and position of a hovering unmanned helicopter [19]. Sam et al. applied the PISMC to a hydraulically-actuated active suspension system [17,18]; Nawawi, et al. focused on the control and manipulation of robotic systems. [15,16]. In [20], robust integral sliding mode control (ISMC) was proposed for an underactuated rotary hook system in order to deal with parametric uncertainties presenting themselves in the system's parameters. Pan et al. studied the modification of the integral sliding variable degrades, and suggested that the switching element is smoothed by using a low-pass filter [21]. More works related to sliding mode control can be found in [22,23]. In [22] a SMC was designed to control a washout filter with constant impedance and nonlinear constant power loads. In [23] a PDSMC was designed to collaborate with a sliding perturbation observer to control a robot manipulator.

One thing to note is the work, cited in [15], done by Nawai et al. in 2006. In this reference, Nawawi suggested PISMC for the two-wheel inverted pendulum mobile robot to avoid high chattering drawbacks. Even though [15] had a similar topic and methodology to this paper, the details were significantly different. In [15], a linear model was investigated; the proportional and integral sliding modes shared the same coefficient, and the integration applied to the derivative of the states. In this paper, nevertheless, there are three major differences. First, we do not restrict our system to a linear one. Actually, the examples in this paper were presented and controlled using the original nonlinear form. Second, we introduce two parameters, instead of one coefficient as in [15], in the investigation. This setting provides us more degrees of freedom to tune parameters and improve the system performance. Third, in the integral we include not only the derivative of the states, but also the states themselves through the error term. Consequently, the work described here and our contributions in the PISMC apparently differ from those made by other researchers.

More specifically, we cite the following list the differences from [15]:

1. Design of sliding surface for the controlled system: Our paper presents $s = 2\dot{e} + (\lambda + 2\alpha)e + \alpha^2 \int e d\tau$. In [15], the sliding surface was designed as $s = k_p e + k_i \int e d\tau$.
2. The system of control in this paper is nonlinear, whereas the system in [15] was a linearized system.
3. Because the system to control in [15] was linear, their proposed controller was merely a function of the system, given by

$$u = Kx - (CB)^{-1}Cf - (CB)^{-1}\eta^2 \text{sgn}(s), \quad (1)$$

where f is the original form representing a nonlinear function describing the deviation from linearity in term of disturbances and un-modeled dynamics. Our controller, however, has two major differences from theirs: First, we do not employ the remainder of the nonlinearity terms directly. Instead, we employ the bound of the remainder nonlinearity terms. Secondly, the saturation function in our controller is more than a $\text{sgn}(s)$ function; the selection of parameters is different.

4. The tuning procedures of parameters are discussed in this paper, but not in [15].

The work presented in this paper extends the work presented in [19] to a wider range. The following novelties or contributions are presented in this paper. First, differently from the commonly known ISMC, PISM, and PID sliding surface, which usually each have the sliding surface of $s = k_p e + k_I \int e d\tau + k_d \dot{e} + k_1 e(0) + k_2 \dot{e}(0)$ and five parameters or more to tune, our algorithm in [19] extends the sliding surface to the n th order derivative of the output error with only two parameters to tune. In [19], however, a system of unweighted control was considered. Based on the work in [19], this paper proposes the sufficient condition to design a control for a system of weighted control. Secondly, our algorithm was realized and verified in a two wheel vehicle system. In the literature, researchers either applied their work to nonlinear systems using numerically simulated results [21,24], or realized their algorithms in a linear system [20,22,23]. Although nonlinearities were considered in [22,23], they were treated as disturbances instead of dynamics themselves. This paper not only verifies our capability of controlling a nonlinear system by numerical simulations, but also realizes the algorithms in a TWVS using its original nonlinear model. Instead of treating the nonlinearities as disturbances like works in the literature, we dealt with the nonlinearity as the dynamics itself. The successful implementation of our algorithm in a practical nonlinear mechanical system indicates that it is possible to extend the application of the PISM to a wide range of nonlinear systems, such as the control of microfluidics-based devices [25,26]. Thirdly, tuning of control parameters is always critical in designing a control. In our proposed algorithms, there are only four parameters to tune for all systems. From analysis and past experience, we roughly understand the effects of these parameters, though they were not rigorously proven. Systematic tuning procedures are also presented in this paper.

In detail, we first briefly review the SMC and PSMC and their drawbacks. The algorithms and control law by PISM are then introduced. Before applying the PISM control law to real cases, robustness, systematic tuning procedures, and performance are then discussed. Having shown the robustness and good performance of our algorithm, the control design for a weighted-control system is proposed. This paper then derives the dynamic model of an experimental TWVS with given parameters. The PISM was designed based on the TWVS model, and all the input information was obtained from the real tests which were conducted by the TWVS. The corresponding control to stabilize the TWVS was derived based on the proposed PISM algorithm, and tested in experiments. Discussion of experimental results and a conclusion are provided to demonstrate the validity of our proposed algorithm.

2. Control Algorithm

2.1. Conventional Sliding Mode Control

The theory of SMC has been extensively applied in linear and nonlinear systems, and different practical engineering control problems. Studies in the past few decades have demonstrated that the SMC methodology provides a systematic approach to solving control engineering problems. By introducing a sliding mode to the control of a system, one can achieve both stabilization and disturbance rejection.

The conventional PSMC is considered as a proportional type which forces system states to slide into the predetermined sliding surface, and to reach the origin of the phase plane eventually. When the state-trajectory is sliding on the sliding surface, the corresponding dynamic performance of the system

is governed by the equations of the sliding surface. Therefore, the PSMC has a strong robustness against external uncertainties and disturbances.

There are two major requirements for the PSMC. The first one is to design a sliding surface. Then the states are enforced to slide into the sliding surface by the switching the control law. Secondly, each of the tracking points on the sliding surface must satisfy the sliding conditions. The switch of the control law is determined by the predefined switching conditions. With the help of the control switch, the trajectory can be guided into the region bounded by boundary layers. Governed by the sliding conditions, the trajectory gradually moves onto the sliding surface and steps forward to the target point.

Consider a desired signal $\tilde{x}_d(t)$ and the system output $\tilde{x}(t)$, where both of $\tilde{x}_d(t)$ and $\tilde{x}(t)$ are in $\mathbb{R}$. The objective of the PSMC is to make the system's output approach the desired signal when $t \rightarrow \infty$; i.e.,

$$\lim_{t \rightarrow \infty} [\tilde{x}(t) - \tilde{x}_d(t)] = 0, \quad (2)$$

Here, we define the tracking error as

$$e(t) = \tilde{x}(t) - \tilde{x}_d(t), \quad (3)$$

where $e(t) \in \mathbb{R}$. Before the PSMC is applied, a sliding surface must be designed in advance. The goal of the PSMC is to guide the system trajectory toward the sliding surface, which naturally leads the trajectory to the target point in the end. Based on the magnitude of errors, we define a time-varying linear differential function in $\mathbb{R}^n$, $s(\tilde{x}, t)$, named the condition space [27], given by

$$s(\tilde{x}, t) = \left(\frac{d}{dt} + \lambda \right)^{n-1} e(t), \quad (4)$$

where λ is a positive constant. The equation $s(\tilde{x}, t) = 0$ represents a sliding surface passing through the origin of the phase plane. According to the above equation, $s = 0$ is an $(n - 1)$ th-order linear differential equation, and all the eigenvalues of the characteristic equation are located in the left complex plane. Once the sliding surfaces are determined, we may start to design the controller and define the sliding conditions as follows [27].

$$s\dot{s} \leq -\eta|s|, \quad (5)$$

where $\eta > 0$, and $\dot{s}$ denotes the time derivative of s .

2.2. Proportional-Integral Sliding Mode Control

To improve the performance of a SMC, this paper adopts a PISM as our algorithm. Consider a dynamic system described by an n th-order nonlinear differential equation

$$\tilde{x}^{(n)}(t) = f(\tilde{x}, \dot{\tilde{x}}, \dots, \tilde{x}^{(n-1)}, t) + \tilde{u}(t), \quad (6)$$

where $\tilde{x} \in \mathbb{R}$ is the output of the dynamic system, f is the system function and must be at least once differentiable, $\tilde{u} \in \mathbb{R}$ is the system input, and t is time. We define the state vector, $\tilde{\mathbf{X}} = [\tilde{x}, \dot{\tilde{x}}, \dots, \tilde{x}^{(n-1)}]^T$, for future derivations.

Assumption 1. Consider (6) and assume that there exists an uncertain parameter Δf in the dynamic function $f(\tilde{\mathbf{X}}, t)$, satisfying

$$f = \hat{f} + \Delta f, \quad (7)$$

where $\hat{f}$ is the estimation and Δf is the variation of function f . Assume

$$|\Delta f| \leq \tilde{F}(\tilde{\mathbf{X}}, t), \quad (8)$$

given $\tilde{F}$ the boundary of system uncertainty [27].

The goal of the PISMCM proposed in this paper is to design a switching control law to let the output $\tilde{x}(t)$ approach a given $\tilde{x}_d(t)$, as shown in (2). To design the control system, the system states are assumed to be attainable via a state observer, which sometimes requires measurement of an output or output error, as defined in (3). The sliding surface of the PISMCM consists of two sliding surfaces, s_1 and s_2 , the proportional sliding surface and the integral sliding surface, respectively. Here,

$$s_1 = \left(\frac{d}{dt} + \lambda \right)^{n-1} e(t),$$

as defined in (4), and

$$s_2 = \left(\frac{d}{dt} + \alpha \right)^n \xi_e, \quad (9)$$

where $\lambda > 0, \alpha > 0, \xi_e(t) = \int_0^t e(\tau) d\tau$. The operator $(d/dt + \gamma)^n$ is defined as

$$\left(\frac{d}{dt} + \gamma \right)^n = \frac{d^n}{dt^n} + \binom{n}{1} \gamma \frac{d^{n-1}}{dt^{n-1}} + \cdots + \binom{n}{n-1} \gamma^{n-1} \frac{d}{dt} + \gamma^n, \quad (10)$$

where $\gamma \in \mathbb{R}$ is a parameter to tune, and $\binom{n}{m}$ denotes the k -combinations of a set having n elements. Applying the operator to a function $h(t)$ yields

$$\left(\frac{d}{dt} + \gamma \right)^n h = h^{(n)} + \binom{n}{1} \gamma h^{(n-1)} + \cdots + \binom{n}{n-1} \gamma^{n-1} \dot{h} + \gamma^n h, \quad (11)$$

where the superscript with parentheses denotes the order of derivative, while the superscript without parentheses denotes the power of the parameter γ . In our case, $\gamma = \lambda$ or $\gamma = \alpha$. They are two complementary parameters, which provide the capability of tuning the gains of controller. We note that the expansion of (9) is different from the conventional PSMC or PID sliding surface, since λ and α are generally distinct. The PISMCM is designed as [19]:

$$\tilde{u} = \hat{u}_{eq} + v_{sw}, \quad (12)$$

where, by letting $s = s_1 + s_2$,

$$\hat{u}_{eq} = -\hat{f} - \hat{w}, \quad (13)$$

$$v_{sw} = -k \operatorname{sgn}(s), \quad (14)$$

with

$$\begin{aligned}\hat{w} = & -\hat{x}_d^{(n)} + \left[\sum_{m=0}^{n-2} \binom{n-1}{m} e^{(m+1)} \lambda^{n-m-1} \right] \\ & + \left[\sum_{m=0}^{n-1} \binom{n-1}{m} e^{(m+1)} \alpha^{n-m-1} \right] \\ & + \alpha \left[\sum_{m=0}^{n-1} \binom{n-1}{m} e^{(m)} \alpha^{n-m-1} \right],\end{aligned}\quad (15)$$

and

$$\text{sgn}(s) = \begin{cases} 1 & \text{if } s > 0 \\ 0 & \text{if } s = 0 \\ -1 & \text{if } s < 0 \end{cases}.$$

The controller design is summarized in Theorem 1.

Theorem 1. [19] Consider the system of (6) which satisfies Assumption 1. Let the function $\tilde{u}$ be specified in (12) with $\tilde{u}_{eq}$ being equivalent signal and v_{sw} being switching control defined in (13) and (14), respectively. The gain k is designed to satisfy (5) with the definition of $\hat{w}$ given in (15). If there is a k and an $\eta > 0$ such that

$$k = \tilde{F}(\tilde{\mathbf{X}}, t) + \eta, \quad (16)$$

then the tracking error as defined in (3) will tend to a neighborhood of zero within a finite time.

Proof. The proof is a brief version of [19] for further usage. Define the following Lyapunov function

$$V = \frac{1}{2}(s_1 + s_2)^2 = \frac{1}{2}s^2. \quad (17)$$

Choose an appropriate $\tilde{u}$ to satisfy $\dot{V}(s) < 0$. Then,

$$\begin{aligned}\dot{V} &= s\dot{s} \\ &= \left[\hat{x}^{(n)} - \hat{x}_d^{(n)} + \sum_{m=0}^{n-2} \binom{n-1}{m} e^{(m+1)} \lambda^{n-m-1} \right. \\ &\quad + \sum_{m=0}^{n-1} \binom{n-1}{m} e^{(m+1)} \alpha^{n-m-1} \\ &\quad \left. + \alpha \sum_{m=0}^{n-1} \binom{n-1}{m} e^{(m)} \alpha^{n-m-1} \right] s \\ &= (\hat{f} + \hat{w} + \Delta f + \tilde{u})s \\ &= [\Delta f - k\text{sgn}(s)]s \\ &\leq |\Delta f||s| - k|s| \\ &\leq -(k - \tilde{F})|s|.\end{aligned}\quad (18)$$

The sufficient condition of (18) to ensure the stability of system and to satisfy the sliding conditions as in (5) is given by

$$\dot{V} \leq 0.$$

Let

$$(k - \tilde{F})|s| = \eta|s| > 0,$$

provided $\eta > 0$. We conclude

$$k = \tilde{F}(\tilde{\mathbf{X}}, t) + \eta,$$

as stated in (16). It is also conclusive from (18) that $e(t)$ converges to the neighborhood around origin of the phase plane for any trajectory. $\square$

In fact, in the control signal there is a the discontinuous contribution v_{sw} , caused by the discontinuous switching function (sgn). This function results in an infinitely large switching frequency. However, this infinitely large switching frequency can not be realized in real physical system. Therefore, a switching component with extremely high-speed is applied. That way, the state trajectory would oscillate around the two sides of $s = 0$ which results in the chattering phenomenon.

In order to reduce the chattering problem, the concept of boundary layers for a sliding surface is employed, as depicted in Figure 1.

Figure 1. The definition of boundary layer for a sliding surface [19].

The definition of boundary layers is given by

$$B = \{\tilde{\mathbf{X}} | s(\tilde{\mathbf{X}}, t) = \pm\Phi\}, \quad \Phi > 0. \quad (19)$$

The error phase-plane of boundary layers of a sliding surface, s , coincides with the boundary range of ϵ . Note that $s = \pm\Phi$ is the boundary layer. Since the switch action can not be completed in a blink of time and the effect of delay exists, the state trajectory $\tilde{\mathbf{X}}$ would oscillate between the boundary layers. Figure 1 is the 2nd-order error phase plane, and it is the most common way to mitigate the oscillation by replacing the sliding surface with the boundary layers. v_{sw} is the switching law of the controller, defined as follows

$$v_{sw} = -k \text{sat}\left(\frac{s}{\Phi}\right). \quad (20)$$

Here $\text{sat}(s, \Phi)$ is the saturation function defined by

$$\text{sat}(s, \Phi) = \begin{cases} \frac{s}{\Phi} & \text{if } |s| \leq \Phi \\ \text{sgn}(s) & \text{if } |s| > \Phi \end{cases}.$$

2.3. Alternative from of the Controller

In this paper, we assume that all states are attainable via state observers, but do not touch the design of the observer. The original design of the controller $\tilde{u}$ in (12) may require additional estimation of $e^{(n)}$, which may need to employ new gain(s).

In order to avoid this drawback, an alternative design of $\tilde{u}$ is proposed by reducing the order of $e^{(n)}$. (3) leads to

$$\begin{aligned} e^{(n)} &= \tilde{x}^{(n)} - x_d^{(n)} \\ &= \hat{f} + \Delta f + \tilde{u} - x_d^{(n)}. \end{aligned} \quad (21)$$

To have $\dot{V} < 0$, the control $\tilde{u}$ was originally designed as

$$\begin{aligned} \tilde{u} &= -\hat{f} - k\text{sat}(s, \Phi) - \hat{w} \\ &= -\hat{f} - k\text{sat}(s, \Phi) - (-x_d^{(n)} + \hat{w}' + e^{(n)}) \\ &= -\hat{f} - k\text{sat}(s, \Phi) + x_d^{(n)} - \hat{w}' - \hat{f} - \Delta f - \tilde{u} + x_d^{(n)}, \end{aligned} \quad (22)$$

where $\hat{w} = -x_d^{(n)} + \hat{w}' + e^{(n)}$, and

$$\begin{aligned} \hat{w}' &= \left[\sum_{m=0}^{n-2} \binom{n-1}{m} e^{(m+1)} \lambda^{n-m-1} \right] \\ &\quad + \left[\sum_{m=0}^{n-2} \binom{n-1}{m} e^{(m+1)} \alpha^{n-m-1} \right] \\ &\quad + \alpha \left[\sum_{m=0}^{n-1} \binom{n-1}{m} e^{(m)} \alpha^{n-m-1} \right]. \end{aligned} \quad (23)$$

As a result, (22) can be rewritten as

$$2\tilde{u} = -2\hat{f} + 2x_d^{(n)} - \hat{w}' - \Delta f - k\text{sat}(s, \Phi), \quad (24)$$

or,

$$\tilde{u} = -\hat{f} + x_d^{(n)} - \frac{1}{2}(\hat{w}' + \Delta f + k\text{sat}(s, \Phi)), \quad (25)$$

To eliminate Δf , we here re-design $\tilde{u}$ as

$$\tilde{u} = \left[-\hat{f} + x_d^{(n)} - \frac{1}{2}\hat{w}' \right] - k\text{sat}(s, \Phi). \quad (26)$$

That is, replace the original $\hat{w}$ by letting

$$\hat{w} = -x_d^{(n)} + \frac{1}{2}\hat{w}'. \quad (27)$$

We claim that this controller works, and this controller does not need to measure $e^{(n)}$.

The proof is given as follows:

$$\begin{aligned}
 \dot{V} &= (\hat{f} + \Delta f + \hat{w} + \tilde{u})s \\
 &= \left(\hat{f} + \Delta f - x_d^{(n)} + \hat{w}' + e^{(n)} - \hat{f} + x_d^{(n)} - \text{ksat}(s, \Phi) - \frac{1}{2}\hat{w}' \right) s \\
 &= \left(\Delta f + e^{(n)} - \text{ksat}(s, \Phi) + \frac{1}{2}\hat{w}' \right) s \\
 &= \left(\Delta f + \hat{f} + \Delta f + \tilde{u} - x_d^{(n)} - \text{ksat}(s, \Phi) + \frac{1}{2}\hat{w}' \right) s \\
 &= \left(2\Delta f + \hat{f} - \hat{f} + x_d^{(n)} - \text{ksat}(s, \Phi) - \frac{1}{2}\hat{w}' - x_d^{(n)} - \text{ksat}(s, \Phi) + \frac{1}{2}\hat{w}' \right) s \\
 &= (2\Delta f - 2\text{ksat}(s, \Phi))s \\
 &\leq -2\eta \text{sat}(s, \Phi)s.
 \end{aligned} \tag{28}$$

2.4. Robustness Discussion

Unlike the design concept of conventional SMC, the insensitivity is included in the dynamics model and also guaranteed in our algorithm. In the conventional SMC design, a precise dynamics model is employed and the perturbation is viewed as an extra input. An example is given in Equation (22) in [28]. The change rate of s is formulated by

$$\dot{s} = \sigma + M\Delta x + O(\Delta x)^2, \tag{29}$$

and the robustness is discussed based on this formulation. In our approach, however, the perturbations are already included in the parameter k for v_{sw} . Recall that $k = \tilde{F} + \eta$, where $\tilde{F}$ is the bound of the model variation, satisfying $\tilde{f} = \hat{f} + \Delta f$ and

$$|\Delta f| \leq \tilde{F}.$$

Let $\tilde{\mathbf{X}} = \tilde{\mathbf{X}}_n + \Delta\mathbf{X}$, where $\tilde{\mathbf{X}}_n$ is the nominal states. Then,

$$\begin{aligned}
 \tilde{f}(\tilde{\mathbf{X}}) &= \tilde{f}(\tilde{\mathbf{X}}_n + \Delta\mathbf{X}) \\
 &= \tilde{f}(\tilde{\mathbf{X}}_n) + \left. \frac{\partial \tilde{f}}{\partial \tilde{\mathbf{X}}} \right|_{\tilde{\mathbf{X}}_n} + \text{H.O.T} \\
 &= \hat{f}(\tilde{\mathbf{X}}) + \Delta f,
 \end{aligned} \tag{30}$$

where $\hat{f}(\tilde{\mathbf{X}}) = \tilde{f}(\tilde{\mathbf{X}}_n)$ and Δf includes all the remainder terms. By properly investigating and choosing the bound of Δf . The influence of perturbations is included and considered in the control. The advantage of our approach is that the robustness of the control is guaranteed with a proper choice of $\tilde{F}$, and we do not have to worry about specific perturbations. The drawback, of course, it that the control might be inefficient if $\tilde{F}$ is defined too large for the purpose of robustness.

2.5. Parameter Tuning

In the proposed algorithm, there are four parameters to tune in total: λ and α to construct s -surface, η to construct v_{sw} , and Φ to determine the saturation function. We have tried to analytically understand the effect of each parameter. However, this is too difficult due to the complexity of nonlinear systems and mathematics. Instead, approximated properties in a 2nd order system are analyzed, and numerical simulations are presented to demonstrate the potential effect of each parameter, so that readers will have some ideas on our tuning procedures. Notably, numerical examples are not solid proofs. An effect may vary in different cases.

The example from [19] is employed to study the effect of each parameter, and how to tune these parameters. The example, given as follows, was originally from [27] and employed to demonstrate the validity of the our proposed algorithm:

$$\ddot{\tilde{x}} + a(t)\dot{\tilde{x}}^2 \cos 3\tilde{x} = \tilde{u}. \quad (31)$$

The function $a(t)$ is unknown but varies within the range $1 \leq a(t) \leq 2$. Let

$$\hat{f} = -1.5\dot{\tilde{x}}^2 \cos 3\tilde{x},$$

The control input of SMC is

$$\begin{aligned} \tilde{u} = & 1.5\dot{\tilde{x}}^2 \cos 3\tilde{x} + \ddot{\tilde{x}}_d - \lambda \dot{e} \\ & - (0.5\dot{\tilde{x}}^2 |\cos 3\tilde{x}| + \eta) \text{sat}\left(\frac{s}{\Phi}\right). \end{aligned}$$

with $s = \dot{e} + \lambda e$. The control input of PISM is

$$\begin{aligned} \tilde{u} = & 1.5\dot{\tilde{x}}^2 \cos 3\tilde{x} + \ddot{\tilde{x}}_d - \alpha^2 e - (\lambda + 2\alpha)\dot{e} - \ddot{e} \\ & - (0.5\dot{\tilde{x}}^2 |\cos 3\tilde{x}| + \eta) \text{sat}\left(\frac{s}{\Phi}\right) \end{aligned}$$

with

$$\begin{aligned} s = & s_1 + s_2 \\ = & \left(\frac{d}{dt} + \lambda\right)e + \left(\frac{d}{dt} + \alpha\right)^2 \zeta_e \\ = & 2\dot{e} + (\lambda + 2\alpha)e + \alpha^2 \int_0^t e d\tau. \end{aligned} \quad (32)$$

Notably, we employed this example in our work in [19], and showed the validity of our work. In this paper, however, we employ this example to show the properties and influences of parameters. As a result, despite the dynamical equation and parameters of the equation being directly employed from [27], other settings are changed. For example, we used state errors and their derivatives, instead of the output rate in the reference. The parameters of the controller are different from those in the reference. The desired signal $\tilde{x}_d$ is set as unit step for investigating transient properties, which is different from the original $\tilde{x}_d = \sin(\pi t/2)$. Therefore, the comparison occurs between our PISM and SMC to know the parameters better. It did not occur between our PISM and the SMC of the reference.

2.5.1. Effect of Φ

Φ is the parameter to determine the switching method and reduce the chattering phenomenon, as shown in (21). However, the larger the value of Φ , the more sluggish the response. As $\Phi \rightarrow \infty$, $\text{sat}(s, \Phi) = s/\Phi$. As $\Phi \rightarrow 0$, $\text{sat}(s, \Phi) = \text{sgn}(s)$, which usually causes chattering phenomenon. Consider the case $\Phi \rightarrow \infty$, so that $\text{sat}(s, \Phi) = s/\Phi$. Since s is finite in general, $|s|/\Phi < 1$. The smaller the $|s|$, the smaller the $|s|^2/\Phi$. The decreasing rate of the Lyapunov function is influenced by

$$\dot{V} \leq -\eta \frac{s^2}{\Phi} = -\eta \frac{|s|^2}{\Phi} \quad (33)$$

The original saturation function $\text{sat}(s, \Phi) = \text{sgn}(s)$ brings $\dot{V} \leq -\eta|s|$. The condition to have a more greatly decreasing rate of the Lyapunov function is given by

$$\eta \frac{|s|^2}{\Phi} \geq \eta|s| \quad (34)$$

or equivalently, $|s| > \Phi$. Therefore, a large Φ is not helpful in speeding up the converge of the system. Figure 2 demonstrates the comparison. $\Phi = 100$ in Figure 2a, whereas $\Phi = 0.1$ in Figure 2b. Other parameters are set identically. It is obvious that the responses with smaller Φ are faster, regardless of the type of controls.

Figure 2. Comparison of parameter effectiveness. (a) $\Phi = 100$; (b) $\Phi = 0.1$.

2.5.2. Effect of η

It can be expected that the larger the value of η , the more decreasing the rate of the Lyapunov function from $\dot{V} \leq -\eta|s|$. Suppose $|s| > \Phi$ at beginning. The upper bound for $\dot{V}$ is given in (33). Since $V = s^2/2 = |s|^2/2$, we conclude the upper bound of V satisfies

$$\begin{aligned}\dot{V} &= -\eta \frac{|s|^2}{\Phi} \\ &= -\frac{2\eta}{\Phi} V\end{aligned}\quad (35)$$

(35) implies that V converges exponentially with time constant $\tau_\eta = \Phi/2\eta$.

Figure 3 demonstrates the comparison. $\eta = 100$ in Figure 3a, whereas $\eta = 0.01$ in Figure 3b. Other parameters are set identically. It is obvious that the responses with large η are faster, regardless of the type of controls.

Figure 3. Comparison of parameter effectiveness. (a) $\eta = 100$; (b) $\eta = 0.01$.

2.5.3. Effect of λ

The role and influence of λ on a higher order system is unclear. However, we may approach this problem through a second order system like the example. When $n = 2$, our algorithm suggests $s = \dot{e} + \lambda e$ and $\tilde{u} = \tilde{u}_R - \lambda \dot{e}$ under the SMC assumption, where $\tilde{u}_R$ contains the remainder terms of $\tilde{u}$. From the structure of $\tilde{u}$ we conclude that the role of λ is roughly the gain of a conventional P-controller in a negative feedback. A proper choice of λ would stabilize the system. Too large a value of λ may destabilize the system, or have the system converged sluggishly.

On the other hand, the function of SMC is to force $s = 0$ first. Then, the property of $s = 0$ will force $e = 0$ when t approaches infinity. Since $s = \dot{e} + \lambda e$, implying that e converges exponentially with time constant $\tau_\lambda = 1/\lambda$, too small a value of λ will cause the system to converge sluggishly. From our experience, $1 \leq \lambda \leq 10$ is a better choice.

Figure 4 demonstrates the comparison. $\lambda = 100$ in Figure 4a, $\lambda = 1$ in Figure 4b, and $\lambda = 0.01$ in Figure 4c. Other parameters are set identically. It is obvious that the responses with too large or too small λ are sluggish, if the SMC control is considered.

Figure 4. Comparison of parameter effectiveness. (a) $\lambda = 100$; (b) $\lambda = 1$; (c) $\lambda = 0.01$.

2.5.4. Effect of α

Similarly, the role and influence of α on a higher order system are unclear. However, we may approach this problem through a second order system. When $n = 2$, our algorithm suggests $s = 2\dot{e} + (\lambda + 2\alpha)e + \alpha^2 \int e d\tau$. Defining $\dot{z} = e$ yields $s = 2\ddot{z} + (\lambda + 2\alpha)\dot{z} + \alpha^2 z$. Moreover, $e \rightarrow 0$ when $z \rightarrow 0$.

A similar statement can be made for α . The function of our algorithm is to force $s = 0$ first. The property of $s = 0$ will then force $z = 0$ when t approaches infinity. Since $s = 2\ddot{z} + (\lambda + 2\alpha)\dot{z} + \alpha^2 z$, z converges exponentially if the discriminant is greater than zero, and converges oscillatorily if the discriminant is less than zero, where the discriminant is defined as

$$D = (\lambda + 2\alpha)^2 - 8\alpha^2 \quad (36)$$

Defining $\alpha = \rho\lambda$ and $\rho > 0$ yields

$$\begin{aligned} D &= \lambda^2 \left[(1 + 2\rho)^2 - 8\rho^2 \right] \\ &= \lambda^2 (1 + 4\rho - 4\rho^2) \end{aligned} \quad (37)$$

The condition for z to converge exponentially is $0 < \rho \leq 1.2$. This approximation is not accurate enough because the actual behavior of z is also influenced by other parameters. We may conclude z converges oscillatorily if $\rho \geq 1$.

Given $\rho \geq 1$, the overshoot of z is influenced by damping ratio of the system. The damping ratio ζ can be approximated by

$$\begin{aligned}\zeta &= \frac{\lambda + 2\alpha}{\sqrt{2}\alpha} \\ &= \frac{1 + 2\rho}{\sqrt{2}\rho}\end{aligned}\quad (38)$$

when $\rho = 1$, $\zeta = 2.213$. When $\rho \rightarrow \infty$, $\zeta = \sqrt{2}$.

Since $e = \dot{z}$, the oscillation of ζ does not directly equal the oscillation of e . Moreover, the nonlinearity of the system also makes the quantitative prediction inaccurate. However, the trend of the oscillation must be similar. The larger the value α is, the smaller the overshoot e has.

Figure 5 demonstrates the comparison. $\alpha = 10$ in Figure 5a, $\alpha = 1$ in Figure 5b, and $\alpha = 0.1$ in Figure 5c. Other parameters are set identically. It is obvious that a very small $\rho = \alpha/\lambda$ causes no overshooting in the system. This is reasonable. When $\alpha = 0$, the algorithm degenerates to a conventional SMC. Although large ρ causes overshooting, the magnitude is small.

The time constant of convergence for an oscillatory z can be approximated by

$$\begin{aligned}\tau_\alpha &= \frac{1}{(\lambda + 2\alpha)/2} \\ &= \frac{2}{\lambda(1 + 2\rho)}\end{aligned}\quad (39)$$

when $\rho \rightarrow 0$, $\tau_\alpha = 2/\lambda$. When $\rho \rightarrow \infty$, $\tau_\alpha \rightarrow 0$.

Although large ρ suppresses overshooting and speeds up the convergence of outputs, $\alpha = \rho\lambda$ grows exponentially compared to the contribution to control signals. Hence, the trade-off of a fast response is a very large control. A very large control is usually energy consuming or impossible in implementation.

Figure 5. Comparison of parameter effectiveness. (a) $\alpha = 10$; (b) $\alpha = 1$; (c) $\alpha = 0.1$.

2.5.5. Tuning Procedures

From previous analysis and our experience, the tuning procedures applying our algorithm are concluded as follows:

1. Select Φ . $0.1 \leq \Phi \leq 1$ is suggested to speed up the convergence.
2. Select η . Large η is helpful to speed up the convergence. However, the value of η is proportional to $\ddot{u}$. Too large a value of η may cause too large a control signal.
3. Select λ . From our experience, $1 \leq \lambda \leq 10$ is a better choice.
4. Select ρ and $\alpha = \rho\lambda$. If an oscillatory response is allowed, ρ can be selected to be larger than 1. If not, a smaller ρ is suggested.
5. Estimate the transient properties:

- (a) The overshoot can be estimated from damping ratio $\zeta = (1 + 2\rho) / \sqrt{2\rho}$ for $\rho \geq 1$.
 - (b) Estimate the total time constant. Since the convergence procedure is $s \rightarrow 0$ and then $e \rightarrow 0$, the total time constant of convergence, τ_e can be approximated by $\tau_e = \tau_\eta + \tau_\alpha$.
 - (c) The settling time t_s , according to control theory, approximated by $t_s = 4\tau_e$.
6. Build the nonlinear mathematical model and run numerical simulations with initially selected parameters. Tune the parameters according to the transient performance and the effect of parameters.
 7. Remember to keep α as small as possible, because a large α usually indicates a very large control input.

2.6. Effectiveness of the Algorithm

As aforementioned, there are only four parameters to tune in total for an arbitrarily high order system: λ and α to construct s -surface, η to construct v_{sw} , and Φ to determine the saturation function. Usually, a higher order controller has more parameters to tune. Using only four parameters to control an arbitrarily high order system may sacrifice potential performance more parameters may bring, but it simplifies the tuning procedures.

On the other hand, the proposed algorithm has more degrees of freedom to adjust the system's performance, compared to the conventional SMC algorithm. Although those characteristics may result in complexity in selecting parameters, with the aforementioned methods and analysis, one is able to tune parameters systematically and efficiently.

Another concern is the cost of the control. Better performance may result from greater effort, such as a larger control signal or greater energy consumption. From our experiences, though not rigorously proven, in some cases the proposed algorithm actually costs less than the conventional SMC, which only tunes λ to build the sliding surface and Φ to defined the boundary layer.

An example is presented in Figure 6. The example was simulated using the aforementioned nonlinear system. In order to evaluate the control laws, a cost function J is defined as follows:

$$J = \frac{1}{2} \int_0^t \tilde{u}^2 d\tau \quad (40)$$

Two control laws were employed to control the system: the SMC and our proposed PISM. Parameters were tuned so that the two systems had similar settling times. A smaller overshoot occurred in the system using PISM. Table 1 lists the parameters:

Table 1. Parameters used for comparison of cost functions.

Algorithm	λ	α	η	Φ
SMC	30	–	28	0.1
PISM	1	4	0.001	0.1

The simulations in Figure 6 show that the proposed algorithm costs less while keeping similar transient performance. Notably, under the designated parameters, chattering occurs once using the control from SMC. As a result, it can be concluded that our algorithm performs better than the conventional SMC in this specific case.

Figure 6. Comparison of parameter effectiveness. (a) Outputs $\tilde{x}$; (b) control input $\tilde{u}$; (c) cost of controls J .

2.7. PISMC for a Weighted-Control System

Based on the work in [19], we extend our work to a nonlinear system of weighted control, which is usually described by the differential equations

$$\begin{aligned}\tilde{x}^{(n)} &= f(\tilde{x}, \dot{\tilde{x}}, \dots, \tilde{x}^{(n-1)}, t) \\ &+ \tilde{g}(\tilde{x}, \dot{\tilde{x}}, \dots, \tilde{x}^{(n-1)}, t) \tilde{u}(t),\end{aligned}\quad (41)$$

where $\tilde{u}, \tilde{x} \in \mathbb{R}$ are the input and output, respectively, and $f, \tilde{g} : \mathbb{R}^n \rightarrow \mathbb{R}$ are smooth functions. The gain $\tilde{g}$ is unknown, yet assumed to be uniformly bounded.

Assumption 2. Assume $0 < \beta_{\min} \leq \tilde{g}(\tilde{\mathbf{X}}, t) \leq \beta_{\max}$, where $\beta_{\min}$ and $\beta_{\max}$ are given constants, time-varying or state-dependent functions [27].

We assume that the control input may enter multiplicatively in the dynamics, and it is natural to choose our estimate $\hat{g}$ of gain $\tilde{g}$ as the geometric mean of the above bounds.

$$\hat{g} = \sqrt{\beta_{\min} \beta_{\max}}. \quad (42)$$

Assumption 2 can then be written in the form of

$$\beta^{-1} \leq \frac{\hat{g}}{\tilde{g}} \leq \beta, \quad (43)$$

where

$$\beta = \sqrt{\frac{\beta_{\max}}{\beta_{\min}}} > 0. \quad (44)$$

Since the PISMC is designed to be robust to the bounded multiplicative uncertainty, as in (43), we call β the gain margin of our design, by analogy to the terminology used in linear control. Let

$$\hat{G} = \frac{1}{\sqrt{\beta_{\max} \beta_{\min}}}. \quad (45)$$

The PISMC $\tilde{u}^*$ is modified from the aforementioned $\tilde{u}$ and it has the form:

$$\tilde{u}^* = \hat{G} (\hat{u}_{eq} + v_{sw}), \quad (46)$$

where the $\hat{u}_{eq}$ and v_{sw} are defined in (13) and (14), respectively. We summarize the controller design in Theorem 2.

Theorem 2. Consider the system (41), which satisfies Assumptions 1 and 2. Let the function $\tilde{u}^*$ be specified in (46) with $\hat{u}_{eq}$, v_{sw} , and $\hat{w}$ defined in (13), (14), and (15), respectively. Let the gain k^* be designed to satisfy (5). If there exists a k^* such that

$$k^* = \left| (\beta - 1) (\hat{f} + \hat{w}) \right| + \beta(M + \eta), \quad (47)$$

then the tracking error as defined in (3) will tend to the neighborhood of zero within a finite time.

Proof. To prove that PISMC $\tilde{u}^*$ satisfies the sliding conditions, the following Lyapunov function is employed:

$$V = \frac{1}{2}s^2.$$

If we choose $\tilde{u}^*$ as in (46), then follow similar steps as those in (18):

$$\begin{aligned} \dot{V} &= s\dot{s} \\ &= (\hat{f} + \Delta f + \tilde{g}\tilde{u}^* + \hat{w})s \\ &\leq -\left[\tilde{g}\hat{G}k^* - \left|(\hat{f} + \hat{w})(1 - \tilde{g}\hat{G}) - \tilde{F}\right|\right]|s|. \end{aligned}$$

Finally, let

$$k^* \geq \left| (\beta - 1)(\hat{f} + \hat{w}) \right| + \beta(\tilde{F} + \eta). \quad (48)$$

We obtain

$$\tilde{g}\hat{G}k^* - \left| (\hat{f} + \hat{w})(1 - \tilde{g}\hat{G}) - \tilde{F} \right| \geq 0,$$

and hence $\dot{V} < 0$. This ensures that any trajectory $e(t)$ converges to the neighborhood around of origin in the phase plane. $\square$

2.8. Comparison of Sliding Surfaces

The design of sliding surface is inherited from [19], whereas the design of control is extended to a weighted-control system. The design of sliding surface in our algorithm extends to the n th order system, and, thus, is considerably different from conventional SMC and PID sliding surface. Table 2 presents the differences.

Table 2. Conceptual comparison between different control laws.

Algorithm	Proposed PISMC	SMC	PID Sliding Surface
Sliding Surface	$s = \left(\frac{d}{dt} + \lambda\right)^{n-1} e + \left(\frac{d}{dt} + \alpha\right)^n \int e d\tau$	$s = k_d \dot{e} + k_p e$	$s = k_d \dot{e} + k_p e + k_i \int e d\tau$
When $n = 2$	$s = 2\dot{e} + (\lambda + 2\alpha)e + \alpha^2 \int e d\tau$	$s = k_d \dot{e} + k_p e$	$s = k_d \dot{e} + k_p e + k_i \int e d\tau$
Robustness	Through k^*	Through insensitivity	Through insensitivity
Parameters tuning	Systematically	Easier	Complex
Performance	Better	Fair	N/A

3. Implementation in a Two-Wheel Vehicle System

This paper extends the design of PISMC controller to a weighted-control system from the work in [19]. Procedures for tuning parameters are also presented in the proceeding section. The third effort of this paper is to implement the proposed control algorithm in a nonlinear system, which is very

rare in the literature. A TWVS was built for the implementation and verification of the algorithm. The novelty of this experiment is the employment of its original nonlinear dynamics, instead of commonly used linearized dynamics.

3.1. Model of Two-Wheel Vehicle System

The dynamic equations for the TWVS were derived for analysis and for the controller design. Figure 7 shows the free body diagram of the TWVS.

Figure 7. (a) Free body diagram of the cart and the pendulum; (b) free body diagram of the wheels

A simplified version of the TWVS is shown at the top of Figure 7a. The shaft of the two wheels goes through the center of mass (cm) of the cart, and a pendulum is installed in the cm of the cart through a hinge. A motor kit is installed on each wheel to drive the wheel. Assume that there is no slip occurred between the wheels and the ground, and no torque exists between the cart and the pendulum. Moreover, this example experiment only considers the longitudinal motion. Therefore, forces and torques in the lateral motion are assumed balanced and not considered.

In order not to distract readers from the main stream of this paper, the detailed derivation of equations of motion is provided in the Appendix A. Notation is defined either in Figure 7 or in the Appendix A. The dynamics of the TWVS are represented as follows:

$$m l g \sin \theta = (m l^2 + J_p) \ddot{\theta} - m l \cos \theta \ddot{x}, \quad (49)$$

$$\left(M + m + 2M_w + \frac{2J_w}{R^2} \right) \ddot{x} - m l \cos \theta \ddot{\theta} = 2 \frac{\tilde{u}^*}{R} - m l \sin \theta \dot{\theta}^2. \quad (50)$$

Define $M_{\text{total}} = M + m + 2M_w + 2J_w/R^2$ and $J_s = J_p + m l^2$. The equations of motion of the TWVS can be written as

$$\mathbf{A} \begin{bmatrix} \ddot{x} \\ \ddot{\theta} \end{bmatrix} = \mathbf{B}, \quad (51)$$

where

$$\mathbf{A} = \begin{bmatrix} M_{\text{total}} & -m l \cos \theta \\ -m l \cos \theta & J_s \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} 2 \frac{\tilde{u}^*}{R} - m l \sin \theta \dot{\theta}^2 \\ m l g \sin \theta \end{bmatrix}$$

or equivalently,

$$\begin{bmatrix} \ddot{x} \\ \ddot{\theta} \end{bmatrix} = \mathbf{A}^{-1} \mathbf{B} = \frac{1}{\Delta} \begin{bmatrix} -J_s m l \sin \theta \dot{\theta}^2 + \frac{2J_s}{R} \tilde{u}^* + \frac{1}{2} m^2 g l^2 \sin(2\theta) \\ -\frac{1}{2} m^2 l^2 \sin(2\theta) \dot{\theta}^2 + \frac{2m l \cos \theta}{R} \tilde{u}^* + M_{\text{total}} g l \sin \theta \end{bmatrix}, \quad (52)$$

where $\Delta = \det(\mathbf{A}) = M_{\text{total}} J_s - m^2 l^2 \cos^2 \theta$.

3.2. Parameter Set-Up of PISMC System

In this section, a numerical simulation and an experiment of PISMC applying to a TWVS are presented. The dynamics of TWVS are defined in (52), and in the simulation, the parameters were selected as $m = 1.076$ kg, $M = 1.012$ kg, $M_w = 0.064$ kg, $R = 0.85$ m, $l = 0.18$ m, $J_p = 0.00221925$ N-m, and $J_w = 0.00132$ N-m. Let $5.5 = \beta_{\min} \leq \tilde{g}(\tilde{\mathbf{X}}, t) \leq \beta_{\max} = 11$. We have $\hat{G} = 0.1286$ and $\beta = \sqrt{2}$. Set $\lambda = 10$, $\alpha = 1$, $\eta = 5$, and $\Phi = 1$. The PISMC $\tilde{u}^*$ can be computed from (46).

3.3. Numerical Simulations

3.3.1. Architecture of the Simulation and Experiment

Figure 8 shows the architecture of the simulation and the actual hardware implementation of the PISMC. The measured output signal was compared with the reference signal and the calculations were performed to produce the switching signal and the equivalent signal. The overall control signal (switching signal + equivalent signal) was sent to the motor driving board to control the real system. In the experiment, the motor driving board contained two PIC18F258 microcontrollers. One microcontroller computed the control signal based on the PISMC algorithm, whereas the other handled the readings of sensors. The computational capability of a PIC18F258 microcontroller is actually powerful enough to implement the proposed algorithm. However, due to the inherent complexity and difficulty in the TWVS, the preliminary goal was to achieve the autonomous balance by a simplified structure. Therefore, the experiment focused on the tilt angle control only, described by the equation of $\ddot{\theta}$ in (52). A simple flow chart is shown in Figure 9.

Figure 8. A block diagram of the overall system.

Figure 9. The experiment focused on the tilt angle control ($\tilde{x} = \theta$) only, due to the inherent complexity and difficulty in the two-wheel vehicle system (TWVS).

3.3.2. Balance Control Simulation Results

Simulation results are presented in Figures 10–12. For the first simulation, Figure 10 shows the system responses of the TWVS balancing with no external disturbances and the initial condition is $(\tilde{x}_d(0), \dot{\tilde{x}}_d(0)) = (1, 0)$. We note that the settling time to reach $\tilde{x}_d$ and the error of the PISMC are smaller and converge quicker than that of the conventional SMC. It is evident that the PISMC presented in this paper is effective for the TWVS to reach the desired attitudes around the equilibrium point in short a period. The stability and accuracy are both maintained in the balance simulation.

Figure 10. Balance response to PISMC without disturbances: **(top)** angular response; **(middle)** response of angular rate; **(bottom)** applied control signals.

Figure 11. Switching and sliding signal for PISM without disturbances: **(top)** history of sliding surface values; **(bottom)** history of saturation conditions.

Figure 12. Balance response to PISM with disturbances of $\sin(2\pi t)$: **(top)** angular response; **(middle)** response of angular rate; **(bottom)** applied control signals.

For the second simulation, Figure 12 shows the system responses of the TWVS with external disturbances. To examine the disturbance rejection property afforded by the proposed control, we consider to simulate the system with initial condition of $(\tilde{x}_d(0), \dot{\tilde{x}}_d(0)) = (-0.5, 0)$, and the system external disturbance is $\sin(2\pi t)$. Figure 11 shows that $\tilde{x}_d$ settles down to the steady state and the

amplitude of the errors are smaller. The results also verify that the PISMIC is capable of tracking the desired states and the errors converge steadily to the neighborhood of zero.

The performance of the system with the proposed PISMIC is better than the use of the SMC controller such that no overshoot, shorter rise time, and shorter settling time were obtained from the proposed controller. The variations of the control efforts are shown in Figure 10 which indicates that the SMC signal is smaller in magnitude and the system settles down in 1.2 s. Several quantitative performance indicators of the attitude tracking qualities for SMC and PISMIC controllers are summarized in Table 3. Variations of the sliding signal for SMC and PISMIC are illustrated in Figure 10. It shows that magnitudes of the variations are smaller in PISMIC. Variations of the sliding surface during the control are illustrated in Figure 10 as well. One notes that the sliding function is $s(t) \neq 0$ when the error signal is not zero. This means that the sliding mode is in the reaching surface up to about 0.92 s, and then arrives the sliding surface. When it reaches to the sliding surface, theoretically it is expected the sliding function to be zero (i.e., $s(t) = 0$). In practical applications, there are always some small deviations and fluctuations at the output measured variable because of uncertainties and disturbances. Here, the average value of the sliding function is zero $s(t) = 0$. As shown in Figures 10–12, we note that these remarkable results are obtained without involving too much control effort.

Table 3. Time domain specifications.

Controller Type	Rise Time	Settling Time	Overshoot Time
SMC	1.2 s	1.3 s	0 s
PISMIC	0.87 s	0.97 s	0 s

4. Experiments and Discussion

4.1. The TWVS Platform

In this study, an experimental TWVS was developed as the experimental platform for gathering data to simulate and verify the proposed controller design. A picture of the TWVS is shown in Figure 13. The TWVS was equipped with a microchip (PIC 18F258)-based onboard computer module which was responsible for the data processing and the autonomous control. The main onboard electronic sensor was the self-developed IMU which consisted of the gyroscope and the accelerometer circuit. They were used to collect data from system identification and to output commands for automatic balance. The main control system was composed of the Borland C++ Builder.

Figure 13. Onboard computer module of the two-wheel vehicle system.

4.2. Attitude Sensors

In general, the orientation of the inertial measurement unit (IMU) is derived from the inertial sensors, i.e., accelerometer and gyroscope sensors. All these sensors provide analog signals, so an analog-to-digital converter (ADC) is required to acquire the data. Therefore, the PIC18F258 made by Microchip Technology, with 8-channel 10-bit ADC, was used. In order to increase the computational efficiency and to perform the data fusion algorithm, microchips (PIC 18F258) served as the processing units of the low-cost IMU, and they communicated with each other through the built-in RS232 bus. The configuration of this self-developed IMU is depicted in Figure 14. This self-developed, low-cost IMU was used to offer attitudes and angular rates for the onboard computer module. The outputs of IMU were Euler angles, angular rates, and accelerations. Although the IMU performs well in normal conditions, its outputs always suffer from unpredictable errors such as moving due to the motor vibrations. The chassis on the TWVS induced vibrations and affected the onboard hardware to some extent. Without proper installation, the IMU probably gives inapplicable outputs. Therefore, vibration isolation or reduction became an important issue for our IMU on a moving platform. Some vibration reduction materials were tested and finally a kind of sound deadening sheet for cars was chosen. The IMU was surrounded by several layers of the sound deadening sheet and then cased into a box.

Figure 14. Attitude sensors: (left) accelerometer; (right) gyroscope.

4.3. Balance Control Experiment Results

For the first experiment with no external disturbances, Figure 15 shows the TWVS tracking responses using the proposed PISMIC and the conventional SMC. In experimental results, the original tilt angle was approximately ± 2 degrees, and the tilt rate was approximately ± 25 degrees/s. Comparing the top left plot (PISMIC) to the top right plot (SMC), one notes that the tilt angle result of the proposed PISMIC controller had no overshoot, shorter rise time, and shorter settling time than that of the conventional SMC controller. Importantly, the experimental result shows that the PISMIC held balance well for the developed TWVS. The results of the second TWVS experiment with external disturbances are shown in Figure 16, which shows that the output of tilt angle by the PISMIC can be tracked well to follow the desired angle. The robustness of the TWVS with the PISMIC is better than that of the conventional SMC. It also demonstrates that the PISMIC can give reasonable angle output of the tilt motion in experiments. Comparing the PISMIC results with the SMC results, the PISMIC has smaller oscillation, smaller overshoot, shorter rise time, and shorter settling time than the conventional SMC. The results of the TWVS experiments validate the balance control capability of the PISMIC.

Figure 15. Experimental results of TWVS without disturbances: **(left)** SMC is implemented; **(right)** PISM is implemented.

Figure 16. Experimental results of TWVS with disturbances: **(left)** SMC is implemented; **(right)** PISM is implemented.

5. Conclusions

This paper presents a generalized proportional-integral sliding mode control (PISM) method and the design of the associated controls. Our previous work generalized the PISM to an n th order, unweighted-control system with only two parameters to tune. This paper extends the work

to an n th order, weighted-control system. The bounds of dynamic variation, usually including state perturbations and disturbances, were employed to design the associated controller. The advantage of this approach is to avoid measuring the uncertainties. A systematic procedure to tune parameters was analyzed and proposed. Numerical simulations of a nonlinear system are presented to demonstrate the validity of the procedure. Robustness of the proposed algorithm is discussed, and the performance is shown to be better than a conventional SMC under certain parameters through the comparison of costs. These examples also demonstrate that the presence of one more parameter from the integral part brings one more degree of freedom with which to tune the system. In addition, the design of a balance control system for a TWVS is presented to validate the algorithm. The simulation results show that the PISMIC autonomously balances the TWVS effectively in both disturbed and undisturbed circumstances, validating the robustness of the algorithm. The performance of the PISMIC is also better than that of the conventional SMC. The balance control experiments of the developed TWVS yielded encouraging results that validate the feasibility and the effectiveness of the proposed PISMIC controller. The successful implementation of the proposed PISMIC using the original nonlinear dynamics potentially extends the applications of the PISMIC to various nonlinear and emerging systems.

Author Contributions: Formal analysis, C.-H.L.; Investigation, C.-H.L.; Resources, F.-Y.H.; Supervision, F.-Y.H.; Validation, C.-H.L.; Writing—original draft, C.-H.L.; Writing—review & editing, F.-Y.H. All authors have read and agreed to the published version of the manuscript.

Funding: Taiwan Ministry of Technology and Science with grant number NSC 97-2221-E-006-235-MY3; NCSIST with grant number MOST 106-2623-E-032-001-D; National Space Organization with grant number NSPO-S-109133.

Acknowledgments: The authors would like to dedicate this paper to the late Fei-Bin Hsiao for his precious comments on this work.

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

Appendix A. Derivation of Dynamic Model

First, consider the dynamics of the left wheel. All notations denoted by subscripts of L and R are associated with the the left and right wheels, respectively. Let A_L be the contact point with the ground; H_L be the friction; N_L be the normal force by the ground; F_{xL} and F_{yL} be the forces exerted by the cart; M_L be the mass of the wheel; $W_L = M_L g$ be the weight; T_L be the controlling torque provided by the motor; and R be the radius of the two wheels.

The dynamic equations of the wheels in the cart-body-fixed frame are shown as follows

$$\begin{cases} -\ddot{\theta}_L J_L &= -T_L + H_L R \\ \ddot{x}_L M_L &= -F_{xL} + H_L \\ 0 &= F_{yL} + N_L - M_L g \end{cases}, \quad (A1)$$

where J_L is the moment of inertia of the wheel and θ_L is the rolling angle of the wheel. A similar process can be applied to the right wheel and yields

$$\begin{cases} -\ddot{\theta}_R J_R &= -T_R + H_R R \\ \ddot{x}_R M_R &= -F_{xR} + H_R \\ 0 &= F_{yR} + N_R - M_R g \end{cases}. \quad (A2)$$

The assumption of no slipping leads to $\dot{x}_L = R\dot{\theta}_L$ and $\dot{x}_R = R\dot{\theta}_R$. Assume the two wheels are identical. Then $M_L = M_R = M_w$, $J_L = J_R = J_w$, $\dot{x}_L = \dot{x}_R = \dot{x}$, $\theta_L = \theta_R = \theta_w$, $T_L = T_R = \tilde{u}^*$. Notably, θ_w is defined positive to have the TWVS move toward $+\mathbf{i}$

The dynamics of the cart is given as follows:

$$\begin{cases} F_{xL} + F_{xR} - F_p = M\ddot{x} \\ -F_{yL} - F_{yR} - N_p = Mg \end{cases} \quad (A3)$$

where M is the mass of the cart. Since longitudinal forces are assumed to be exerted passing through the cm and lateral forces are balanced, there is no equation for torques. This also implies that the cart-body-fixed frame does not rotate and it is, hence, coincident with the inertial frame.

The equations of the pendulum's translation are derived with respect to the cm of the pendulum, whereas the equation of the rotation is derived with respect to the lower tip of the pendulum for convenience, where the pendulum is linked to the cart. Therefore, the equations of motion are obtained by [29]

$$\sum \mathbf{F} = m\mathbf{a}_G, \quad (A4)$$

$$\sum \mathbf{M} = \mathbf{r}_G \times m\mathbf{a}_G + J_p\ddot{\theta}\mathbf{k}, \quad (A5)$$

where m is the mass of the pendulum; J_p is the moment of inertia about the pendulum cm. Moreover, $\mathbf{r}_G$ is the position vector pointing from the tip to the pendulum cm, and $\mathbf{a}_G$ is the acceleration of the pendulum cm. The two terms can be found by

$$\mathbf{r}_G = l(-\cos\theta\mathbf{i} + \sin\theta\mathbf{j}) \quad (A6)$$

$$\mathbf{a}_G = \mathbf{a}_c + \ddot{\theta}\mathbf{k} \times \mathbf{r}_G + \dot{\theta}\mathbf{k} \times (\dot{\theta}\mathbf{k} \times \mathbf{r}_G), \quad (A7)$$

where l is the half length of the pendulum and $\mathbf{a}_c = \ddot{x}\mathbf{i}$ is the acceleration of the cart. Expanding (A4) through (A7) yields (49) and

$$F_p = ml\sin\theta\ddot{\theta}^2 + m\ddot{x} - ml\cos\theta\ddot{\theta} \quad (A8)$$

$$N_p - mg = -ml\cos\theta\ddot{\theta}^2 - ml\sin\theta\ddot{\theta}. \quad (A9)$$

Substituting (A1), (A2), and (A8) into (A3) yields

$$H_L - 2M_w\ddot{x} + H_R - (ml\sin\theta\ddot{\theta}^2 + m\ddot{x} - ml\cos\theta\ddot{\theta}) = M\ddot{x} \quad (A10)$$

Moreover, $H_L = \tilde{u}^*/R - J_w\ddot{x}/R^2$ and $H_R = \tilde{u}^*/R - J_w\ddot{x}/R^2$. Consequently, (A10) can be re-written as (50), given as

$$\left(M + m + 2M_w + \frac{2J_w}{R^2}\right)\ddot{x} - ml\cos\theta\ddot{\theta} = 2\frac{\tilde{u}^*}{R} - ml\sin\theta\ddot{\theta}^2.$$

References

1. Utkin, V.I. Variable structure with sliding mode: A survey. *IEEE Trans. Autom. Control.* **1977**, *22*, 212–222. [CrossRef]
2. Sira-Ramirez, H.; Zribi, M.; Ahmad, S. Dynamical sliding mode control approach for vertical flight regulation in helicopters. *IEE Proc. Control Theory Appl.* **1994**, *194*, 19–24. [CrossRef]
3. Utkin, V.I.; Jingxin, S. Integral sliding mode in systems operating under uncertainty conditions. In Proceedings of the 35th Conference on Decision and Control, Kobe, Japan, 13 December 1996; pp. 4591–4596.
4. Baik, I.C.; Kim, K.H.; Youn, M.J. Robust nonlinear speed control of PM synchronous motor using boundary layer integral sliding mode control technique. *IEEE Trans. Control Syst. Technol.* **2000**, *8*, 47–54. [CrossRef]
5. Lin, C.J.; Chen, C.K.; Li, K.S. Extended sliding-mode control for the tracking problem of a rotational inverted pendulum. *J. Chin. Soc. Mech. Eng.* **2003**, *24*, 277–282.

6. Hess, R.A.; Wells, S.R. Sliding mode control applied to reconfigurable flight control design. *J. Guid. Control Dyn.* **2003**, *26*, 452–462. [[CrossRef](#)]
7. Rafimanzelat, M.R.; Yazdanpanah, M.J. A novel low chattering sliding mode controller. In Proceedings of the 5th Asian Control Conference, Melbourne, Victoria, Australia, 20–23 July 2004; Volume 3, pp. 1958–1963.
8. Niu, Y.; Ho, D.W.C.; Lam, J. Robust integral sliding mode control for uncertain stochastic systems. *Automatica* **2005**, *41*, 873–880. [[CrossRef](#)]
9. Koshkouei, A.J.; Burnham, K.J.; Zinober, A.S.I. Dynamic sliding mode control design. *IEE Proc. Control Theory Appl.* **2005**, *152*, 392–396. [[CrossRef](#)]
10. Bouabdallah, S.; Siegwart, R. Backstepping and sliding mode techniques applied to an indoor micro quad-rotor. In Proceedings of the IEEE International Conference on Robotics and Automation, Barcelona, Spain, 18–22 April 2005; pp. 2259–2264.
11. Chen, C.A.; Chiang, H.K.; Lin, B.R. The novel adaptive sliding mode position control synchronous reluctance motor drive. *J. Chin. Soc. Mech. Eng.* **2008**, *29*, 241–247.
12. Park, S.; Tan, C.W.; Kim, H.; Hong, S.K. Oscillation control algorithms for resonant sensors with applications to vibratory gyroscopes. *Sensors* **2009**, *9*, 5952–5967. [[CrossRef](#)] [[PubMed](#)]
13. Waslander, S.L.; Hoffmann, G.M.; Jang, J.S.; Tomlin, C.J. Multi-agent quad-rotor test-bed control design: Integral sliding mode vs. reinforcement learning. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Edmonton, AB, Canada, 2–6 August 2005; pp. 468–473.
14. Soylu, S.; Buckham, B.; Podhorodeski, R. MIMO sliding-mode and H^∞ controller design for dynamic coupling reduction in underwater-manipulator systems. *Trans. Can. Soc. Mech. Eng.* **2009**, *33*, 731–744. [[CrossRef](#)]
15. Nawawi, S.; Ahmad, M.; Osman, J.; Husain, A.; Abdollah, M. Controller Design for Two-wheels Inverted Pendulum Mobile Robot Using PISMC. In Proceedings of the 4th Student Conference on Research and Development, Shah Alam, Malaysia, 27–28 June 2006; pp. 194–199.
16. Nawawi, S.; Osman, J.; Ahmad, M. A PI Sliding Mode Tracking Controller with Application to a 3-DOF Direct drive Robot Manipulator. In Proceedings of the TENCON 2004—2004 IEEE Region 10 Conference, Chiang Mai, Thailand, 21–24 November 2004; Volume 4, pp. 455–458.
17. Sam, M.; Osman, J.; Ghani, R. Proportional integral sliding mode control of a quarter car active suspension. In Proceedings of the TENCON 2002—2002 IEEE Region 10 Conference, Beijing, China, 28–31 October 2002; Volume 3, pp. 1630–1633.
18. Sam, Y.; Suaib, N.; Osman, J. Hydraulically Actuated Active Suspension System with PISMC. *WSEAS Trans. Syst. Control* **2008**, *3*, 859–868.
19. Lin, C.H.; Jan, S.S.; Hsiao, F.B. Autonomous hovering of an experimental unmanned helicopter system with proportional-integral sliding mode control. *ASCE J. Aerosp. Eng.* **2011**, *243*, 338–348. [[CrossRef](#)]
20. Tho, H.D.; Tasaki, R.; Kaneshige, A.; Miyoshi, T.; Terashima, K. Robust Sliding Mode Control with Integral Sliding surface of an underactuated rotary hook system. In Proceedings of the 2017 IEEE International Conference on Advanced Intelligent Mechatronics (AIM), Munich, Germany, 3–7 July 2017; pp. 998–1003.
21. Pan, Y.; Yang, C.; Pan, L.; Yu, H. Integral Sliding Mode Control: Performance, Modification, and Improvement. *IEEE Trans. Ind. Inform.* **2018**, *14*, 3087–3096. [[CrossRef](#)]
22. Abbasi, S.; Kallu, K.; Lee, M. Efficient Control of a Non-Linear System Using a Modified Sliding Mode Control. *Appl. Sci.* **2019**, *9*, 1284. [[CrossRef](#)]
23. Monsalve-Rueda, M.; Candelo-Becerra, J.; Hoyos, F. Dynamic Behavior of a Sliding-Mode Control Based on a Washout Filter with Constant Impedance and Nonlinear Constant Power Loads. *Appl. Sci.* **2019**, *9*, 4548. [[CrossRef](#)]
24. Zheng, X.; Jian, X.; Du, W.; Cheng, H. Nonlinear Integral Sliding Mode Control for a Second Order Nonlinear System. *J. Control Sci. Eng.* **2015**, *2015*. [[CrossRef](#)]
25. Cairone, F.; Gagliano, S.; Bucolo, M. Experimental study on the slug flow in a serpentine microchannel. *Exp. Therm. Fluid Sci.* **2016**, *76*, 34–44. [[CrossRef](#)]
26. Schembri, F.; Sapuppo, F.; Bucolo, M. Experimental classification of nonlinear dynamics in microfluidic bubbles' flow. *Nonlinear Dyn.* **2012**, *67*, 2807–2819. [[CrossRef](#)]
27. Slotine, J.J.E.; Li, W. *Applied Nonlinear Control*; Prentice-Hall: Englewood Cliffs, NJ, USA, 1991.

28. Vaidyanathan, S.; Lien, C.H.E. *Applications of Sliding Mode Control in Science and Engineering*; Springer: New York, NY, USA 2017.
29. Hibbeler, R.C. *Engineering Mechanics: Dynamics*, 14 ed.; Pearson: New York, NY, USA, 2016.

© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).