Sliding Mode Based Self-Tuning PID Controller for Second Order Systems

Alper BAYRAK∗1

1Abant Izzet Baysal University, Faculty of Engineering and Architecture, Department of Electrical and Electronic Engineering, 14030, Bolu

Abstract: In this paper, a sliding mode based self-tuning PID controller is proposed for uncertain second order systems. While developing the controller, it is assumed that the system model has a part which contains nonlinear terms similar to PID structure which is a new approach in the literature. The controller and update rules for controller parameters are obtained from Lyapunov stability analysis. The proposed controller with update rule is experienced on an experimental 2-DOF helicopter which is also known as Twin-Rotor Multi-Input Multi-Output System (TRMS). From experiments, it was seen that the PID parameter update rules run satisfactorily and, in parallel with this, the controller achieved the control objective by providing the system track the desired trajectory.

Keywords
Self-Tuning PID controller, Sliding mode controller, Second order systems

1. Introduction

PID control is the most preferred control technique in industrial applications since its simple structure and convenience in implementation [1]. However, the effectiveness of the PID controller is based on the accurate selection of its parameters. Despite the good performance results in linear systems, the selection of the parameters might be very difficult and time wasting with the rise of nonlinearities of the system. To deal with this problem many approaches of self-tuning PID controllers have been presented till today. These approaches can be separated into two main categories: i) model based approaches and ii) rule-based approaches. In model based approaches, the tuning mechanism is based on the knowledge of system model. In rule based approaches, the tuning is based on some optimization or estimation rules without model knowledge, which basically mimics an experienced operator’s behavior [2]. A good survey can be found in [2] on this topic.

In the literature, many studies can be found on self-tuning PID controller and its applications. In [3], a self-tuning method for PID controllers based on theory of adaptive interaction for the quadrotor system were presented. In the manuscript it is reported that the determining of the adaption coefficient appear as a problem. In [4], a self-tuning PID control scheme based on support vector machine (SVM) and particle swarm optimization (PSO) was presented. Jiang and Jiang proposed a fuzzy based self-tuning PID controller for temperature control [5]. Zheng et. al. used fuzzy module to tune PID parameters with respect to the error and change in error [6]. In [7] and [8], genetic algorithm was utilized to tune of PID parameters. Na presented a study on water level control of a nuclear steam generator with PID controller of which parameters were tuned by model predictive control (MPC) [9]. Fan et. al., used neural network to tune PID controller to track the position of a pneumatic artificial muscle [10]. Gundogdu and Komurgaz presented a self-tuning algorithm for PID based on an adaptive interaction approach [11]. In [12], Howell and Best used continuous
action reinforcement learning automata (CARLA) method to tune the PID controller parameters while controlling engine idle-speed. Bobal et al. presented a self-tuning PID controller for process control modelled by $\delta$-models [13]. In [14], wavelet neural network based identifier was used to develop an auto tuning adaptive PID controller to prevent wing rock phenomena. In [15], Shih and Tseng designed a self-tuning PID controller by using integral of time-weighted absolute error (ITAE) optimal control principle and the pole-placement approach to control position of a servo-cylinder. Dong and Mo presented model reference adaptive PID controller for motor control system [16]. In [17], Chamsai et al. presented an adaptive PID controller combined with sliding mode controller for uncertain nonlinear systems. Chang and Yan proposed an adaptive PID controller based on sliding mode controller for uncertain chaotic systems [18]. Kuo et al. presented an adaptive sliding mode controller with PID tuning method for a class of uncertain systems [19]. In [20], Huang et al., presented an adaptive control system for online tuning of PID controllers for SISO systems. In [21], a pole assignment self-tuning PID control algorithm was presented.

In this paper, a sliding mode based self-tuning PID controller is proposed for uncertain second order systems. Different from the literature, it is assumed that the model contains nonlinear terms similar to PID structure. The idea lies under this assumption is that the system model should contain a similar structure with PID controller for the controller to be effective. The controller and update rules for PID parameters are obtained from Lyapunov stability analysis. The proposed controller is designed to control an experimental 2-DOF helicopter (i.e. TRMS) which has highly nonlinear behavior and has no accurate dynamic model. The controller has a quite simple structure and also easily applicable for all second order systems.

The rest of the paper is presented as follows; the system model is given in Section 2. Control and parameter update rule design are presented in Section 3. Experimental results are given in Section 4. Finally conclusions are presented in Section 5.

2. System Model

The following second order system is considered in this paper,

$$
\begin{align*}
\dot{x}_1(t) &= x_2(t) \\
\dot{x}_2(t) &= f(x) + u(t)
\end{align*}
$$

(1)

(2)

where $x(t) = [x_1(t), x_2(t)]^T$ is state vector, $u(t) \in \mathbb{R}$ is control signal. The function $f(\cdot) : \mathbb{R}^2 \rightarrow \mathbb{R}$ is assumed in the form of

$$
f(x) = g(x) + k_p x_1(t) + k_d x_2(t) + k_i \int x_1(t) dt.
$$

(3)

where $g(\cdot) : \mathbb{R}^2 \rightarrow \mathbb{R}$ is unknown nonlinear function, $k_p$, $k_d$ and $k_i$ are unknown system parameters.

Assumption 1. It is assumed that the function $g(\cdot)$ is bounded as

$$|g(x)| \leq \rho$$

(4)

where $\rho$ is known.

Assumption 2. It is assumed that the system parameters $k_p$, $k_d$ and $k_i$ are in known bounded regions.

Assumption 3. It is assumed that $x(t)$ is available and continuous.

3. Control and Parameter Update Rule Design

The objective of the controller is to utilize $x_1(t)$ track a desired trajectory while updating PID parameters. To achieve this objective, the error system is designed as follows,

$$
\begin{align*}
\dot{x}_1 &= x_{d1} - x_1 \\
\dot{x}_2 &= x_{d2} - x_2
\end{align*}
$$

(5)

(6)

where $x_{d1}$ and $x_{d2}$ are desired trajectories. To construct sliding mode controller, the filtered error signal is designed as

$$s = \dot{x}_2 + 2\lambda \dot{x}_1 + \lambda^2 \int \dot{x}_1 dt.
$$

(7)

where $\lambda \in \mathbb{R}^+$ is a constant.

The derivative of (7), which will be utilized later, is

$$s = \dot{x}_2 + 2\lambda \dot{x}_1 + \lambda^2 \dot{x}_1
$$

(8)

The control input is designed as

$$u = u_{PID} + u_R
$$

(9)

where $u_R$ is sliding part of the controller and

$$u_{PID} = \ddot{k}_p \dot{x}_1 + \ddot{k}_d \dot{x}_1 + \ddot{k}_i \int \dot{x}_1 dt
$$

(10)

where $\ddot{k}_p$, $\ddot{k}_d$ and $\ddot{k}_i$ are estimates of $k_p$, $k_d$ and $k_i$, respectively.

By substituting the (9) and (10) in (8), it is obtained as

$$
\dot{s} = \dddot{x}_{d2} - k_p x_1 - k_d \dot{x}_1 - k_i \int x_1 dt
$$

$$
- \dddot{k}_p x_{d1} - \ddot{k}_d \dot{x}_{d1} - \dot{k}_i \int x_{d1} - u_R + 2\lambda \dot{x}_{d1}
$$

$$
- 2\lambda \dot{x}_1 + \lambda^2 \int \dot{x}_1 dt
$$

(11)


where

$$
\ddot{k}_p = k_p - \dddot{k}_p, \quad \ddot{k}_d = k_d - \dddot{k}_d, \quad \ddot{k}_i = k_i - \dddot{k}_i.
$$

(12)

The Lyapunov function in (13) is utilized to construct update rules for PID gains and design $u_R$,

$$V = \frac{1}{2} x_1^2 + \frac{1}{2} x_2^2 + \frac{1}{2} \int x_1 dt + \frac{1}{2} \int x_2 dt
$$

(13)
The derivative of (13) is obtained as
\[
V = s\dot{x} + \hat{k}_p \dot{x} + \hat{k}_d \ddot{x} + k_i \dot{k}_i
\]
\[
= s(\dot{x}_d - g - \hat{k}_p x_d - \hat{k}_d \dot{x}_d - \dot{k}_i \int x_d - u_R
\]
\[
+ 2\lambda \dot{x}_d - 2\lambda \dot{x}_1 + \lambda^2 \dot{x}_1 - \hat{k}_p (x_1 + \hat{k}_p)
\]
\[
- \hat{k}_p (x_1 + \hat{k}_d) - k_i (s \int x_1 + \dot{k}_i).
\]
(14)

From (14), the update rules of \(\hat{k}_p, \hat{k}_d\) and \(\dot{k}_i\) are selected as in (15) to eliminate the terms with gain errors.
\[
\hat{k}_p = -sx_1, \quad \hat{k}_d = -sx_1, \quad \dot{k}_i = -s \int x_1.
\]
(15)

After substitution of (15) in (14), \(V\) is obtained as
\[
V = s(\dot{x}_d + 2\lambda \dot{x}_1 + \lambda^2 \dot{x}_1) - sg
\]
\[
- s(\dot{k}_p x_d + \hat{k}_d \dot{x}_d + \dot{k}_i \int x_d) - su_R.
\]
(16)

The input signal \(u_R\) should be designed to make \(V\) negative. To achieve this purpose, \(u_R\) will be investigated by separating into three terms as
\[
u_R = u_1 + u_2 + u_3.
\]
(17)

\(u_1\) is designed as to eliminate first two terms in (16) as
\[
u_1 = \dot{x}_d + 2\lambda \dot{x}_1 + \lambda^2 \dot{x}_1 + ksgn(s), \quad k \in \mathbb{R}^+.
\]
(18)

To eliminate the term \(sg\) in (16), the condition in assumption 1 can be utilized. From (4) the following inequality can be obtained
\[-sg \leq |s| \rho, \quad \rho \in \mathbb{R}^+\]
(19)

By using (19), \(u_2\) is designed as follows
\[
u_2 = \frac{|s|}{s} \rho
\]
(20)

By substituting (18) and (20) in (16), \(V\) is obtained as
\[
V \leq -k |s| - sL - su_3
\]
(21)

where
\[
L = \hat{k}_p x_d + \hat{k}_d \dot{x}_d + \dot{k}_i \int x_d.
\]
(22)

An upper bound for \(L\) can be defined as
\[
L_m > |\hat{k}_p| |x_d| + |\hat{k}_d| |\dot{x}_d| + |\dot{k}_i| \int |x_d|,\]
(23)

where \(\hat{k}_p, \hat{k}_d\) and \(\dot{k}_d\) are upper bounds of \(k_p, k_i\) and \(k_d\), respectively. Hence, the following inequality can be written as
\[-sL < |s| L_m,\]
(24)

Remark 3.1. In (23), \(L_m\) may go to infinity for \(x_d \neq 0\) since integral term. But it should be kept in mind that the main interested term is \(|s| L_m\). So if it can be proven that \(s(t)\) converge to 0, fast enough, then, it can be assumed that the term \(|s| L_m\) stays bounded. So \(L_m\) can be accepted as bounded.

In the rest of the paper, it will be proven that \(s(t)\) converge to 0 with a tunable rate.

From (24),
\[
V < -k |s| + |s| L_m - su_3.
\]
(25)

So, \(u_3\) can be obtained as
\[
u_3 = \frac{|s|}{s} L_m.
\]
(26)

This leads
\[
V < -k |s|.
\]
(27)

From (27), it can be said that \(s, \hat{k}_p, \hat{k}_d\) and \(\dot{k}_i\) are bounded. To show that \(s(t)\) goes to zero with respect to time, it should be investigated in deep by taking the time derivative of \(s^2\) as
\[
\frac{1}{2} \frac{d}{dt} s^2 = ss
\]
\[
= (\dot{x}_d - g - k_p x_1 - k_d \dot{x}_1 - k_i \int x_1 - u
\]
\[
+ 2\lambda \dot{x}_d - 2\lambda \dot{x}_1 + \lambda^2 \dot{x}_1) s.
\]
(28)

By substituting (9) and (15) in (28), it is obtained as,
\[
\frac{1}{2} \frac{d}{dt} s^2 < -k |s| + (-\hat{k}_p x_1 - \hat{k}_d \dot{x}_1 - \dot{k}_i \int x_1) s
\]
(29)

If \(k\) is selected as
\[
k > |\hat{k}_p x_1 + \hat{k}_d \dot{x}_1 + \dot{k}_i \int x_1| + \eta
\]
(30)

where
\[
\hat{k}_p = \overline{k}_p - k_p
\]
(31)

\[
\hat{k}_d = \overline{k}_d - k_d
\]
(32)

\[
\dot{k}_i = \overline{k}_i - \dot{k}_i,
\]
(33)

where \(\hat{k}_p, \hat{k}_d\) and \(\dot{k}_d\) are lower bounds of \(k_p, k_i\) and \(k_d\), respectively. (29) is obtained as
\[
\frac{1}{2} \frac{d}{dt} s^2 < -\eta |s|
\]
(34)

which leads
\[
\dot{s} < -\eta |s|.
\]
(35)

From (35), it is seen that starting from any initial condition, the state trajectory reaches to the surface in a finite time smaller than \(|s(t = 0)| / \eta\) and then converges to \(x_d(t)\) exponentially with a time constant equal to \(1/\lambda\) [22].

4. Experimental Results

The performance of the control law in (9) and update rules in (15) were tested on a 2-DOF helicopter which is known as TRMS. The 2-DOF helicopter, shown in Figure 1, is constructed in our laboratory and controlled via LabView software. During the experiment, the parameter values of input signal were selected as \(\lambda = diag(2, 2), k = diag(10, 10), \rho = [5 5]^T\), \(L_m = [100 50]^T\). The initial values of gain estimates were set to \(k_p = [400 700]^T, k_d = [40 50]^T\), and \(k_i = [10 30]^T\). The initial positions of the axes were 0 and the desired positions were selected as
\( \mathbf{x}_d = [30 \ 30]^T \) in degree. It should be noted that \( \rho \) was selected by means of the limited knowledge about the system.

The position errors and the control inputs of yaw and pitch axes are presented in Figures 2, 3, 4 and 5, respectively. Input signal is duty cycle of the PWM signal with a precision of 1024. The PID gain estimates are given in Figures 6, 7 and 8. As can be seen in the Figures 2 and 3, both the yaw and the pitch errors are driven to the vicinity of zero. So it can be said that the control law performed satisfactorily.

In Figures 6, 7 and 8, it is seen that \( \hat{k}_p \), \( \hat{k}_d \) and \( \hat{k}_i \) went to constant values with respect to error signal and those constant gains provided the stability of the system. This situation shows the accuracy of the obtained gains. The update rule reached the optimal gain values which took the system to the equilibrium point in a short time. The update rule differs from the existing tuning methods with its structure.

The effectiveness of the controller and update rule was tested experimentally on a TRMS. Since TRMS is known as a highly-nonlinear system in the literature, the performance of the control and update rules on this system is an important criterion which indicates the success of the system. From the Figures 2, 3 and Figures 6-8, it can be said that the control and update rules work successfully and update rule finds the optimal gains which take the system to the equilibrium point in a short time. The update rule also provides the control system adapt itself to the changes in the system parameters, quickly. These features make the self-tuning control system more advantageous than the classic controllers in the sense that both obtaining optimal gains and adapting the control system to possible variances in system parameters.
Figure 4. Input signal for yaw axis.

Figure 5. Input signal for pitch axis.

Acknowledgments

The experimental equipment used in this study is developed under the support of Scientific Research Projects Commission of Abant Izzet Baysal University under grant number 2015.09.03.836.

References


