High order sliding mode observer for fault actuator estimation and its application to the three tanks benchmark

Matthias Rolink, Taha Boukhobza, Dominique Sauter

To cite this version:

Matthias Rolink, Taha Boukhobza, Dominique Sauter. High order sliding mode observer for fault actuator estimation and its application to the three tanks benchmark. Nov 2006, German-French Institut for Automation and Robotics, pp.CDROM, 2006. <hal-00121029>

HAL Id: hal-00121029
https://hal.archives-ouvertes.fr/hal-00121029
Submitted on 19 Dec 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers. L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
HIGH ORDER SLIDING MODE OBSERVER FOR
FAULT ACTUATOR ESTIMATION AND ITS APPLICATION
TO THE THREE TANKS BENCHMARK

M.Rolink**, T. Boukhobza* and D.Sauter*

* Research Centre in Automatic Control (CRAN - CNRS UMR 7039), Nancy University, BP 239, 54506 Vandoeuvre Cedex, France, Phone: 33 383 684 464, Fax: 33 383 684 462
and
** Institute for Automatic Control and Complex Systems (AKS)
Faculty of Engineering, University of Duisburg-Essen
Bismarckstr. 81 BB, 47057 Duisburg, Germany
email: ernie99@gmx.de

Abstract: This work deals with the design of a FDI scheme on the basis of sliding mode techniques for a 3 three tanks system. More precisely, we focus on the design of high-order sliding mode observer to estimate the magnitude of the actuator faults. The main advantage of the use of high order sliding mode techniques is that it allows to avoid the chattering phenomenon which is inherent to the classical first order sliding mode observers and controllers. The efficiency of the proposed observers is illustrated on a Matlab/Simulink simulator.

Keywords: High-order sliding mode techniques, Fault detection and Isolation, Fault estimation, 3-tanks-system.

1. INTRODUCTION

Nowadays, there exists a high demand by industry for the implementation of plant-wide networking systems. This requires the use of distributed control and maintenance in industrial processes. The use of networked control system (NeCS) can provide many advantages that will contribute to meet these requirements. In this context, the NeCST project [8] was started in September 2004 with the aim to explore research opportunities of distributed control systems in order to enhance the performance of diagnostics and fault tolerant control systems.

The fundamental purpose of a fault detection and isolation (FDI) scheme is to generate an alarm when a fault occurs and also, if it is possible, to locate the fault or to estimate its magnitude [2].

In this paper, a FDI scheme for the benchmark of the NeCST project was developed. This benchmark platform, consisting in a 3-tanks-system, contains of several actuators, like pumps and valves. The corresponding FDI scheme was realized on the basis of sliding mode techniques.

The concept of sliding modes was emerged from the Soviet Union in the sixties, where the effects of introducing discontinuous control action into dynamical systems were explored [2,4,9,13]. Beside the control context, sliding mode techniques are also used for observation and fault detection aims [1,3,12,15]. With the help of sliding mode techniques, it is possible to estimate the magnitude of a fault [2]. This may very helpful for the design of Fault Tolerant Control strategies.

More precisely, in this paper, we discuss some algorithms related to sliding mode techniques. Particularly, our interest lies here in the elimination of the well-known chattering phenomenon which can have a big negative influence in fault reconstruction. To avoid this phenomenon high-order sliding modes can be used. The application of such techniques to the FDI problem constitutes the main contribution of this paper.

The structure of the paper is as follows: In section 2, we recall briefly some useful principles of the sliding mode theory. In section 3, after the description of the 3-tanks-system, we present a general overview of first order and second order sliding mode observers design for the 3-tanks-system. In section 4 some simulation results are presented and compared. Finally, some concluding remarks end the paper.
2. RECALLS ON SLIDING MODE THEORY

The sliding mode technique is related to the variable structure systems (VSS) theory. It is based essentially on the resolution of differential equations with discontinuous right hand side and is introduced by Filippov in 1960 [5]. This technique is used in control theory since the works of Emelyanov [4], Itkis [9] and Utkin [13]. The control community was interested in sliding modes because the latter can be easily implemented and because of their well-known robustness against perturbations and model uncertainties [12,14,15]. In this section we will recall briefly some essential parts of this technique.

2.1 FIRST ORDER SLIDING MODE

Let us consider the following non-linear system [13]:

\[
\dot{x} = f(x, u) \\
\text{s}(x) = 0
\]  

(2.1)

where \(x \in \mathbb{R}^n\) is the state and \(f\) is a smooth vector field. \(u \in \mathbb{R}^m\) represents the input of the system with

\[
\begin{cases} 
  u^+(x) & \text{if } s(x) > 0 \\
  u^-(x) & \text{if } s(x) < 0
\end{cases}
\]  

(2.2)

where \(s(x)\) is a smooth function which is called sliding surface. It separates the space in two separated parts \(s^+ = s(x) > 0\) and \(s^- = s(x) < 0\).

In the case where the system state is in \(s^+\) or \(s^-\) it will converge towards the surface with the “speeds” \(f^+\) and \(f^-\) which are defined as follows:

\[
\begin{cases} 
  \dot{x} = f^+(x) & \text{if } s(x) > 0 \\
  \dot{x} = f^-(x) & \text{if } s(x) < 0
\end{cases}
\]  

(2.3)

We say that the surface is attractive if some conditions are satisfied. These conditions are detailed in [13] but we can roughly summarize them by the inequality \(s \dot{s} < 0, \forall s \neq 0\).

When the surface \((s(x) = 0)\) is reached, the state will stay on it and sliding along. This movement along the discontinues surface will be called sliding mode.

The equivalent control method [15] allows to calculate the dynamic of system (2.1) on the sliding surface. It has been proved in [13], that the dynamics of the system, when the sliding mode is achieved, is equivalent to the dynamic of the system when it is submitted to an equivalent continuous control \((u_{eq})\) which is equal to the control \(u\) allowing to maintain the system on \(s(x) = 0\). Indeed, this equivalent control consists in a control which will be obtained by the solution of the equation

\[
\dot{s}(x) = 0
\]  

(2.4)

The previous equation gives us directly the dynamics of the system on the sliding surface.

In spite of the different advantages of sliding modes control, its use has an important disadvantage which is called chattering (Fig. 1). Indeed, an ideal sliding mode has a control which commutes with an infinity frequency around the surface (Fig. 2). Nevertheless, there do not exist a computer or a control circuit to realize this.

Fig. 1 – Non ideal Sliding Mode with chattering

Fig. 2 – Ideal Sliding Mode

The chattering is a natural consequence of the real dynamic behaviour of the controlled actuators systems. These control commutations can degrade the control performance or can initiate high undesirable dynamic frequencies in the system. Thus different methods to avoid this phenomena are studied in the literature.

One easy solution to avoid chattering is to replace the discontinuous \(\text{sign}\) function by a continuous approximation with a high gain in the boundary layer like a saturation function (Fig 3).

\[
sat(s) = \begin{cases} 
  \frac{s}{\varepsilon} & \text{if } |s| \leq \varepsilon \\
  \text{sign}(s) & \text{if } |s| > \varepsilon
\end{cases}
\]  

(2.6)

Fig. 3 – Saturation function type

Chattering can be avoided with this function but unfortunately sliding mode performances will be compromised. In the next section we will present another solutions based on the theory of higher-order sliding modes[10].
2.2 HIGH-ORDER SLIDING MODES

To reduce or to avoid the chattering phenomenon High-Order Sliding Modes (HOSM) can be used. They generalize the basic sliding mode idea, acting on the higher order time derivatives of the system deviation from the constraint, instead of influencing the first deviation derivative as it happens in standard sliding modes [7].

HOSM have the advantage that they can totally remove the chattering effect whereas they are keeping the main advantages of the original approach (robustness).

To realize HOSM there exist several algorithms. We focus in this paper on 2 specific algorithms which are the so-called twisting and super-twisting algorithms. Indeed, for our system, it is sufficient to use just 2nd order sliding mode algorithms.

Considering the system

\[ \dot{x} = f(t, x, u) \]
\[ s = s(t, x) \]  

where \( x \in \mathbb{R}^n, u = U(t, x) \in \mathbb{R} \) is the control, \( f \) and \( s \) are smooth functions.

The aim of 2nd order sliding algorithm is to achieve the 2nd order sliding mode at a chosen surface and so to obtain after a finite time \( s = \dot{s} = 0 \).

To achieve this, there must exist constants \( s_0, K_m, K_M \) and \( C_0 \) such that system (2.7) satisfy [10]:

1) \( 0 < K_m \leq |\frac{\partial f}{\partial x}| \leq K_M \)  
2) \( |\frac{\partial s}{\partial x} f(t, x, u) + \frac{\partial s}{\partial x} \dot{s}(t, x, u)| \leq C_0 \)  
3) \(|s| < s_0\)

\[ \dot{s} = s(t, x, u) \]  

\[ s = s(t, x) \]  

2.2.1 TWISTING ALGORITHM

Historically the twisting algorithm is the first known 2nd order sliding algorithm [6]. It features twisting around the origin of the 2nd order sliding plane (Fig. 4). The trajectories perform an infinite number of rotations while converging in finite time to the origin.

Let the relative degree of the system be one. Considering the system (2.7), the control algorithm is defined by the following control law [6,10], whereby the area around the surface is restricted by \( u \).

\[ u = \begin{cases} 
-u & \text{when } |u| > |u| \\
-\lambda_m \text{sign}(s) & \text{when } s \dot{s} \leq 0, |u| \leq |u| \\
-\lambda_M \text{sign}(s) & \text{when } s \dot{s} > 0, |u| \leq |u| 
\end{cases} \]  

(2.11)

where \( \lambda_m \) and \( \lambda_M \) satisfy the conditions

\[ \lambda_M > \lambda_m \]  

(2.12)

\[ \lambda_m > 4 \frac{K_m}{s_0} \]  

(2.13)

\[ \lambda_m > \frac{C_0}{K_m} \]  

(2.14)

Ideally, the control law (2.11) ensures that the trajectory converges in finite time towards the sliding surface.

\[ K_m \lambda_M - C_0 > K_M \lambda_m + C_0 \]  

(2.15)

2.2.2 SUPER-TWISTING ALGORITHM

In this algorithm the trajectories on the 2nd order sliding plane are also characterized by twisting around the origin (Fig. 5). But this algorithm was just developed to control systems with relative degree one in order to avoid chattering. The continuous control law \( u \) consists of 2 terms. The first is defined by means of its discontinuous time derivative, while the other is a continuous function of the available sliding variable. The super-twisting algorithm has the advantage that it needs not the time derivative of the sliding surface.

The algorithm is defined by the following control law [6,10]:

\[ u = u_1(t) + u_2(t) \]

\[ u_1 = \begin{cases} 
-u & \text{when } |u| > |u| \\
-W \text{sign}(s) & \text{when } |u| \leq |u| 
\end{cases} \]  

(2.16)

\[ u_2 = \begin{cases} 
-\lambda s \text{sign}(s) & \text{when } |s| > s_0 \\
-\lambda |s| \text{sign}(s) & \text{when } |s| \leq s_0 
\end{cases} \]  

Beside the conditions (2.12),(2.13),(2.14),(2.15), system (2.7) must satisfy:

\[ W > \frac{C_0}{\lambda_m} \]  

(2.17)

\[ \lambda^2 \geq \frac{44 C_{K_M} W + C_0}{\lambda_m \lambda M (W + C_0)} \]  

(2.18)

\[ 0 < \rho \leq 0.5 \]  

(2.19)

An exponentially stable 2nd order sliding mode is achieved if the control law (2.16) with \( \rho = 1 \) is used. The maximal value for \( \rho \) is equal to 0.5. This choice ensures that a 2nd order sliding mode occurs [7].
The aim of this work is to design sliding mode observers for the estimation of actuator faults on a 3-tanks-system. More details on this system will be provided in section 3.

Let us consider the system

\[ \dot{x} = Ax + Bu + Ef \]
\[ y = Cx \]  \hspace{1cm} (2.20)

where \( f \) is a possible actuator fault with its gain matrix \( E \) such that \( E = B \).

To this system is the following observer associated:

\[ \dot{\hat{x}} = A \hat{x} + Bu + L\dot{\omega} \]
\[ \dot{\hat{y}} = C \hat{x} \]
\[ \dot{\omega} = \omega \]  \hspace{1cm} (2.21)

The variable \( \omega \) will be replaced with a sliding mode control law like (2.11) or (2.16).

With (2.20) and (2.21) the surface of the sliding mode algorithms becomes

\[ s = e = \dot{y} - y = C(\dot{x} - x) \]  \hspace{1cm} (2.22)

With (2.20),(2.21),(2.22), we can rewrite (2.8) and (2.9) as:

\[ 0 < K_n \leq \left| \frac{\partial s}{\partial \omega} \right| \leq K_M \]
\[ \Rightarrow 0 < K_n \leq |CL| \leq K_M \]  \hspace{1cm} (2.21)

\[ |CA(Ae + L\omega + Ef) + CE\dot{f}| < C_0 \]  \hspace{1cm} (2.22)

At Figure 6, we represent the implementation of the observer. The function \( \omega = f(s) \) represents the sliding mode control law depending on \( s \) when we use a first order sliding mode observer or \( \omega = \int f(s) \) for a second order sliding mode observer.

3 OBSERVER DESIGN

3.1 THE 3-TANKS-SYSTEM

The 3-tanks-system (Fig. 7) consists of 3 cylindrical tanks [11]. The control objective for the platform is to merge a liquid in tank 2 with the help of the other tanks and the corresponding pumps and valves. In order to satisfy to the
distribution of FDI schemes constraints, we have 4 control/FDI nodes. Each node controlsa subsystem, which consists of one pump and for every pump a observer is designed. According to this, each observer corresponds to an actuator.

To simulate the designed observers we use the “NeCST Benchmark Platform In Open Loop simulator” from April 2006 of the CRAN.

3.2 SYSTEM MODELLING

For the 3-tanks-system we have the following equations [11]:

Tank 1: 
\[ S_1 \cdot \dot{L}_1 = Q_{01} - Q_{10} - Q_{12} \]  
(3.1)

Tank 2: 
\[ S_2 \cdot \dot{L}_2 = Q_{02} - Q_{20} + Q_{12} + Q_{32} \]  
(3.2)

Tank 3: 
\[ S_3 \cdot \dot{L}_3 = Q_{03} - Q_{30} - Q_{32} \]  
(3.3)

Thereby, 
\[ Q_{10} = S_{10} \cdot \sqrt{2 \cdot g \cdot L_1} \]
\[ Q_{32} = S_{32} \cdot \sqrt{2 \cdot g \cdot (L_3 - L_2)} \]

Note that \( Q_{10} \) and \( Q_{32} \) have non-linear dynamics. However, fortunately these flowrates are measured and therefore no modelling of \( Q_{10} \) and \( Q_{32} \) is needed. So we obtain the following linear state space representation.

\[
\begin{bmatrix}
\dot{L}_1 \\
\dot{L}_2 \\
\dot{L}_3
\end{bmatrix} =
\begin{bmatrix}
\frac{1}{S_1} & 0 & 0 & 0 & 0 & Q_{01} \\
0 & \frac{1}{S_2} & \frac{1}{S_2} & 0 & 0 & Q_{12} \\
0 & 0 & 0 & \frac{1}{S_3} & \frac{1}{S_3} & Q_{32}
\end{bmatrix}
\begin{bmatrix}
Q_{01} \\
Q_{12} \\
Q_{02} \\
Q_{20} \\
Q_{03} \\
Q_{30}
\end{bmatrix}
\]  
(3.4)

where, \( S_1 = 0.03 \text{ m}^2 \) and \( S_2 = S_3 = 0.017 \text{ m}^2 \).

We have obtained now a system in the form

\[
\begin{align*}
\dot{x} &= Bu \\
y &= Cx
\end{align*}
\]  
(3.5)

3.3 REALIZATION WITH SIMULINK

For the following sections we consider only pump 1 as an example because the other pumps with their corresponding observers are acting equivalently.

With respect to (3.5) and (2.21) we obtain the following observer.

\[
\begin{align*}
\dot{x} &= Bu + L \omega \\
\dot{y} &= C \dot{x} \\
\dot{f} &= \omega
\end{align*}
\]  
(3.6)

This system is realized with Simulink as it is depicted in the following scheme.

According to the scheme (Fig. 8) we have:

\[ Q_{01} : \text{Flowrate which enters Tank 1. Controlled by Pump 1} \]
\[ Q_{12} : \text{Flowrate which leaves Tank 1 and enters Tank 2. Controlled by Pump 2} \]
\[ Q_{10} : \text{Perturbation of Tank 1. Controlled by a manual valve} \]
\[ L_1 : \text{Level of Tank 1 with a possibly fault} \]
\[ L_{1}' : \text{Level of Tank 1 without fault obtained with the observer} \]
\[ f_{01} : \text{The output with the estimated fault} \]

The function \( \omega = \int f(s) \) represents a 2nd order sliding mode control law like twisting- or super-twisting algorithm.

To simulate a fault the output signal of one pump in the simulator must be multiplied with a value between 0 and 1. So we obtain the following modified system

\[
\dot{x} = (1 - \alpha) Bu \\
y = Cx
\]  
(3.7)

4. SIMULATION RESULTS

We implement the previous sliding mode algorithms using a MATLAB/Simulink simulator. Moreover, we have considered that the 3-tanks-system is a discrete system. This lead to high frequency perturbations on the estimated fault signal \( f \). To reduce these perturbations a low pass filter is realized with the following PT1 element.

\[
\dot{f} = \frac{1}{0.5\text{s} + 1} \omega
\]  
(3.8)

All the simulations are done under the following conditions (for Tank 1):

- \( L_1 = 75 \text{ cm} \)
- \( Q_{01} = 4 \text{ l/min} \)
- \( Q_{12} = 4 \text{ l/min} \)
- \( 0 < \alpha < 1 \)
- Fault appears between \( 400 \text{s} < t < 600 \text{s} \).

4.1. SIMULATION RESULTS FOR THE FIRST ORDER SLIDING MODE

Figure 9 highlights the chattering phenomenon which appears, in the case of a first order sliding mode with
respect to the conditions (3.9) and $\alpha=0.5$.

As we can see, the chattering phenomenon makes the estimation error unacceptable to be used in an efficient FDI or FTC schemes.

To avoid this chattering phenomenon, a 2nd order sliding modes is used.

4.2. SIMULATION RESULTS OF TWISTING ALGORITHM

To simulate the twisting algorithm the following scheme (Fig. 10) with respect to (2.11) was implemented as the function $\omega=\int f(s)$.

![Twisting algorithm in Simulink](image1)

Fig. 10 – Twisting algorithm in Simulink

The results of the simulation considering (3.9) and $\alpha=0.5$ can be seen at Figure 11:

![Fault of Pump 1](image2)

Fig 11 – Fault of Pump 1

We can see that the fault is sufficiently well estimated by the observer and with the help of the filter one obtains an estimated fault with reduced perturbations.

At Figure 12, we can see the estimation errors in function of $\alpha$ (3.9).

![Estimation error for the actuator fault of Pump 1 with twisting algorithm](image3)

Fig. 12 – Estimation error for the actuator fault of Pump 1 with twisting algorithm

For example we have for an actuator fault of 20% a difference between reference and actual debit of $0.8$ l/min (flowrate error). The corresponding estimation error which is computed using the formula (3.10) is equal to $2.5\%$.

$$e = \left| \frac{\hat{f}_1 - f_0}{f_1} \right| \times 100$$ (3.10)

whereby $f_0l$ is the estimated fault and $f1$ is the real value of the fault.

4.3. SIMULATION RESULTS OF SUPER-TWISTING ALGORITHM

To simulate the super-twisting algorithm the scheme depicted in Figure 15 with respect to (2.16) was implemented as the function $\omega=\int f(s)$.

![Super-twisting algorithm in Simulink](image4)

Fig.15 – Super-twisting algorithm in Simulink

The results of the simulation considering (3.9) and $\alpha=0.5$ are shown in Figure 13:

![Fault of Pump 1 with super-twisting algorithm](image5)

Fig. 13 – Fault of Pump 1 with super-twisting algorithm
We can see that the fault is sufficiently well estimated by the observer. Using with filter (3.8), we obtain an estimated fault with reduced perturbations.

In Figure 14, we represent the estimation errors for several values of α (3.9).

The estimation error are computed as in the section before with formula (3.10).

5 CONCLUSION

In this paper sliding mode observers are designed in order to estimate fault actuators magnitude. More precisely, we have considered different known algorithms to implement sliding modes techniques for a 3-tanks-system system. Especially attention lied here in techniques to reduce or avoid the chattering phenomenon. The easiest way is to use a first order sliding mode with a saturation function. But unfortunately this will compromise the performance and therefore it wouldn't be the best way to avoid chattering. The chosen solution is to use 2nd order sliding techniques. They are quite more complex to design than first order sliding modes but they can reduce significantly the chattering phenomenon without any performance loss. Particularly, we have compared the twisting and the super-twisting algorithm. Both algorithms allows a very good fault estimation and can be used in practise. Indeed, the estimation error lies under 5 % for an actuator fault for the twisting algorithm and is less than 2% for the super-twisting algorithm. Because the super-twisting algorithm has also less perturbations in the estimated fault signal it works better than the twisting algorithm.

The next step of this work will be the implementation of the designed observers on the 3-tanks-system.

REFERENCES


