# 非線性控制
# Nonlinear Control

## 第八章作業 (滑動模式控制)


學　　號：P46071204

研 究 生：蔡 昱 哲

授課教授：楊 憲 東

Department of Aeronautics and Astronautics

National Cheng Kung University

Tainan, Taiwan, R.O.C.


中 華 民 國 109 年 12 月 26 日

# 目錄

# 第1題. 滑動模式控制設計

考慮下列之二階非線性系統

$$\begin{aligned}\dot{x}_1 &= x_2 + \theta_1 x_1 \sin(x_2) \\ \dot{x}_2 &= \theta_2 x_2^2 + x_1 + u\end{aligned} \tag{1.1}$$

其中 $u$ 是控制訊號; $\theta_1$ 和 $\theta_2$ 是不確定的參數，但滿足

$$|\theta_1| \le a,\ |\theta_2| \le b \tag{1.2}$$

本題的目的是要設計滑動控制使得系統(1.1)為 Lyapunov 穩定。假設所採取的滑動曲面(Sliding Surface)為

$$S = (1+a)x_1 + x_2 = 0 \tag{1.3}$$

**Question (a)**

試證明在不確定性參數 $\theta_1$ 、$\theta_2$ 的作用下，能確保 $S\dot{S} < 0$ 的滑動控制 $u$ 為

$$u = u_{eq} - \beta(\mathbf{x})\operatorname{sgn}(S) \tag{1.4}$$

其中

$$u_{eq} = -x_1 - (1+a)x_2 \tag{1.5}$$

$$\beta(\mathbf{x}) = a(1+a)|x_1| + bx_2^2 + b_0,\ b_0 > 0 \tag{1.6}$$

Answer

考慮非線性系統(1.1)，首先，先證明若所有狀態位於本題所選定的滑動面上(1.3)，則此滑動面可以使得降階系統的為穩定。考慮所有狀態位於滑動面(1.3)上，整理移項，則此時滿足

$$x_2 = -(1+a)x_1 \tag{1.7}$$

將(1.7)的關係代回(1.1)的第一式中，因此本題的降階系統可以被推導為

$$\dot{x}_1 = -\left[1+\alpha+\theta_1 \sin\left((1+a)x_1\right)\right]x_1 \tag{1.8}$$

針對降階系統(1.8)，選擇 Lyapunov 候選函數為

$$V = \frac{1}{2}x_1^2 \tag{1.9}$$

將(1.9)沿著(1.8)對時間取一次導數，推導如下

$$\begin{aligned}\dot{V} &= x_1\dot{x}_1 \\ &= -(1+\alpha)x_1^2 - \theta_1 \sin\left((1+a)x_1\right)x_1^2 \\ &\le -(1+\alpha)x_1^2 + |\theta_1|x_1^2\end{aligned} \tag{1.10}$$

又將本題系統參數不確定性(1.2)的假設關係 $a \ge |\theta_1|$ 代入(1.10)當中，可以進一步推導為

$$\begin{aligned}\dot{V} &\le -\left(1+\alpha-|\theta_1|\right)x_1^2 \\ &\le -x_1^2\end{aligned} \tag{1.11}$$

(1.11)式的結果表示，當降階系統狀態 $x_1$ 不為 0 時，則 $\dot{V}$ 會恆小於 0，由此我們可以知道，本題所選擇的滑動面(1.3)，將可以使得系統在結構化不確定性(1.2)的假設之下，降階系統狀態 $x_1$ 會以漸近穩定的方式收斂於原點，而當 $x_1$ 到達原點時，又由滑動面(1.3)可以知道，$x_2$ 亦會等於 0，以上，我們可以得出結論，當系統所有狀態軌跡進入我們所選定的滑動面時，則這樣的滑動面可以使得系統所有狀態在滑動面上，皆以漸近穩定的方式到達原點，以上的探討為假設系統狀態在滑動面上所發生面運動行為稱為順滑運動(Sliding Motion)，接下來所要探討的是，要如何設計(1.1)中的控制律 $u$，使得所有系統狀態軌跡在有限時間(Finite Time)內進入滑動面，並且保證其不會離開滑動面。

系統在滑動面上的控制律稱之為等效控制，此時的運動微分方程式滿足

$$S=0,\ \dot{S}=0 \tag{1.12}$$

因此，將(1.3)沿著(1.1)對時間進行一次導數為

$$\begin{aligned}\dot{S}&=(1+a)\dot{x}_1+\dot{x}_2\\&=(1+a)\left[x_2+\theta_1x_1\sin(x_2)\right]+\theta_2x_2^2+x_1+u=0\end{aligned} \tag{1.13}$$

然而，根據題意，在真實情況下我們只知道系統的參數 $\theta_1$、$\theta_2$ 存在不確定性的條件為(1.2)，因此我們只知道系統參數 $\theta_1$、$\theta_2$ 的平均為 0，將此已知代入(1.13)的關係當中，可以推出系統的等效控制律為

$$u_{eq}=-x_1-(1+a)x_2 \tag{1.14}$$

然而由於在真實的情況下，系統的參數 $\theta_1$、$\theta_2$ 會在 0 的附近變化，並且系統狀態並不是一開始就為在滑動面上，因此為了保證系統狀態軌跡在有限時間內進入到滑動面，並且一直保持在滑動面上，透過滑動模式控制的概念，真實的控制律必須加入一些不連續的控制切換，根據題目所給定，真實的控制律被設計為

$$u=u_{eq}-\beta(\mathbf{x})\operatorname{sgn}(S) \tag{1.15}$$

其中

$$u_{eq}=-x_1-(1+a)x_2 \tag{1.16}$$

$$\beta(\mathbf{x})=a(1+a)|x_1|+bx_2^2+b_0,\ b_0>0 \tag{1.17}$$

接著我們必須進一步驗證此控制律是否可以使得系統滿足迫近條件(Approaching Condition)。

考慮本題所選擇的滑動變數(Sliding Variable)

$$S=(1+a)x_1+x_2 \tag{1.18}$$

將其沿著(1.1)對時間進行一次導數為

$$\dot{S}=(1+a)\left[x_2+\theta_1x_1\sin(x_2)\right]+\theta_2x_2^2+x_1+u \tag{1.19}$$

由於滑動變數對時間進行一次導數即會有控制律 $u$ 的出現，因此我們稱此滑動模式控制系統為相對階數 1 的系統，接著利用 Lyapunov 直接定理，分析滑動變數的穩定性，考慮由滑動變數所組成的 Lyapunov 候選函數

$$V_S=\frac{1}{2}S^2 \tag{1.20}$$

將其沿著(1.1)對時間進行一次導數，並將以上所設計的控制律(1.15)、切換增益(Switching Gain)(1.17)，以及不確定性假設(1.2)代入其中，得出

$$
\begin{aligned}
\dot{V}_S &= S\dot{S} \\
&= S\left\{(1+a)\left[x_2+\theta_1 x_1 \sin(x_2)\right]+\theta_2 x_2^2+x_1+u_{eq}-\beta(\mathbf{x})\operatorname{sgn}(S)\right\} \\
&= S\left\{(1+a)\theta_1 x_1 \sin(x_2)+\theta_2 x_2^2-\beta(\mathbf{x})\operatorname{sgn}(S)\right\} \\
&= S\left\{(1+a)\theta_1 x_1 \sin(x_2)+\theta_2 x_2^2-\left[a(1+a)|x_1|+bx_2^2+b_0\right]\operatorname{sgn}(S)\right\} \\
&= \left[(1+a)\theta_1 x_1 \sin(x_2)+\theta_2 x_2^2\right]S-\left[a(1+a)|x_1|+bx_2^2+b_0\right]|S| \\
&\le \left|(1+a)\theta_1 x_1+\theta_2 x_2^2\right||S|-\left[a(1+a)|x_1|+bx_2^2+b_0\right]|S| \\
&\le -b_0|S|
\end{aligned}
\tag{1.21}
$$

由(1.21)的穩定性分析推導結果可以得知,滑動變數不管初始值是為$S>0$或者$S<0$，最終皆會收斂到達滑動面$S=0$上，又從(1.21)不等式結果

$$\dot{V}_S \le -b_0|S| \tag{1.22}$$

的形式說明，滑動變數勢必會在有限的時間收斂到達滑動面$S=0$，接著，進一步分析切換控制律在整個滑動模式運動的行為，在切換增益(1.17)中的前 2 項，負責抑制系統參數不確定性，而第 3 項$b_0$為提供順滑變數在有限時間的收斂力道。

**Question (b)**

用 Matlab 模擬以上滑動控制律的正確性。設定 $a=b=1$，並使 $\theta_1$ 和 $\theta_2$ 在區間 $[-1,1]$ 內任意變化，每次模擬均取不一樣的 $\theta_1$ 和 $\theta_2$，例如 $\theta_1=\sin t$，$\theta_2=\cos t$，或是取成 $\pm1$ 之間的任意隨機亂數(利用 Matlab 的隨機亂數產生器)。用數值模擬驗證，當 $\theta_1$ 和 $\theta_2$ 在區間 $[-1,1]$ 內任意變化時，滑動控制律(1.4)都可確保相平面軌跡進入滑動曲面 $S=0$，同時觀察是否有顫動現象伴隨發生。

<u>Answer</u>

本題假設系統存在結構化不確定性 $|\theta_1|\leq 1,\ |\theta_2|\leq 1$，利用 Matlab 的 rand 函式、cos 以及 sin 函式，製造出隨機的系統不確定性，並且在區間 $[-1,1]$ 內，其時間變化如 圖 1.1 所示

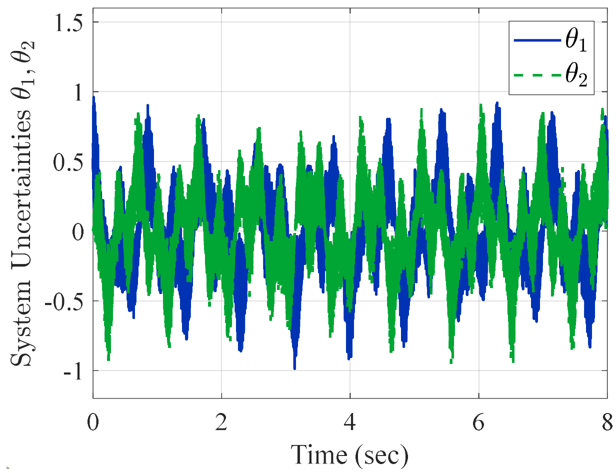



圖 1.1、系統 $a=b=1$ 之參數不確定性時間變化圖

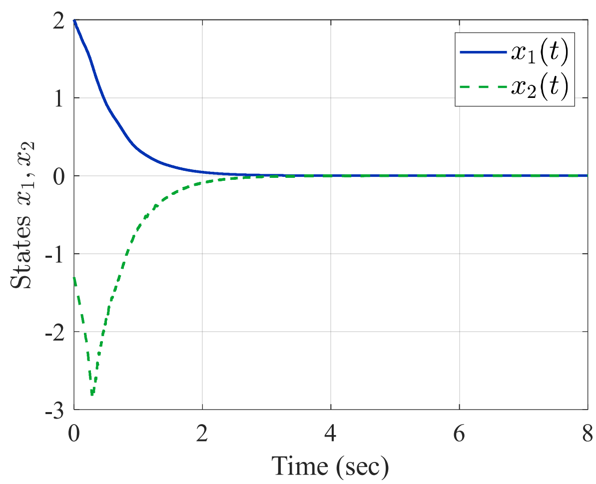



圖 1.2、系統 $a=b=1$ 之狀態收斂時間響應

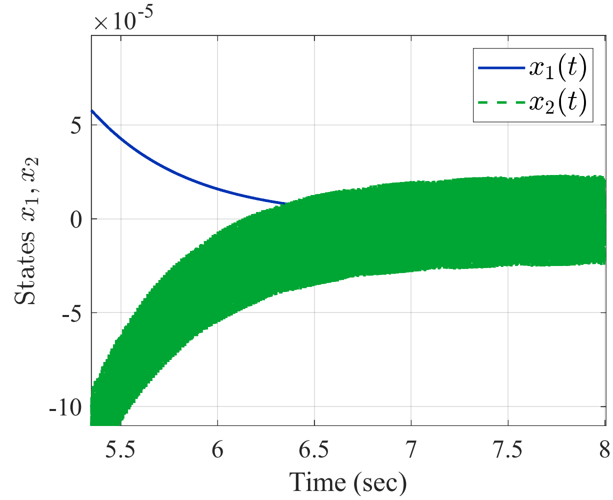



圖 1.3、局部放大之系統狀態收斂時間響應

並且，根據(a)小題的設計滑動模態控制設計，在$a=b=1$的系統參數不確定性設置下，其受控系統的狀態收斂時間響應如 圖 1.2，從圖中可以看出，這樣的控制律設計，使得系統的兩個狀態$x_1$、$x_2$皆能夠收斂到原點，而在這裡特別提到，由於這樣的設計下，只可保證狀態在有限時間內收斂到滑動面上，而在滑動面上，降階系統一樣是透過漸進穩定的方式收斂到原點。

根據穩定性分析(1.22)，滑動變數受到一連串不連續的控制訊號控制後，會在有限時間內收斂到達滑動面上，其時間響應如 圖 1.4 所示，在圖中可以看到由於系統存在著一個固定界限不確定性，因此滑動變數在滑動面上，會出現顫振(Chattering)的情形，這樣的情形是由於不連續的控制訊號所導致 圖 1.5，在 圖 1.5 當中可以看到，輸入訊號不連續的切換增益是由狀態以及一個常數項$b_0$所決定，因此一開始的切換增益因為狀態的關係大，逐漸轉為小，而由於本題的固定切換增益設計為$b_0=1$，因此不連續控制訊號在經過一段時間後，會在正負 1 之間做切換。

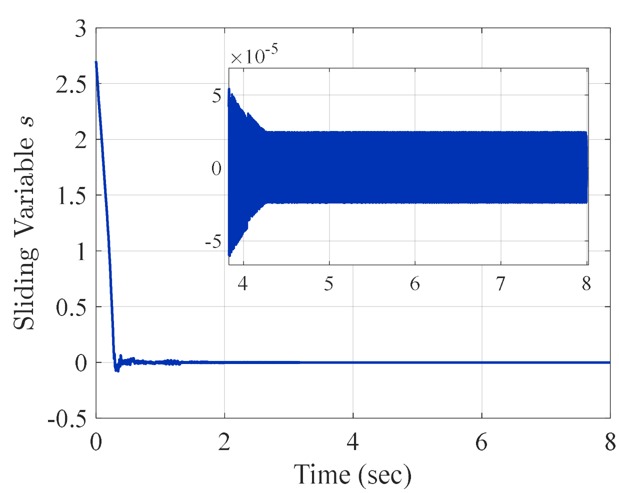



圖 1.4、系統$a=b=1$之滑動變數有限時間收斂響應

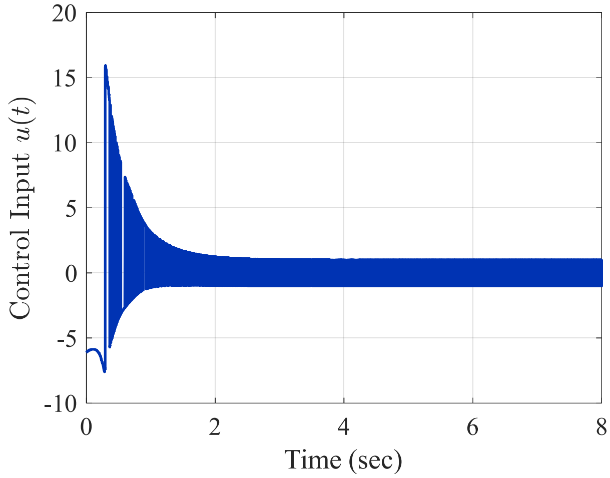


圖 1.5、系統$a=b=1$之滑動模式不連續控制輸入訊號

根據本題所設計的受控系統，以下也繪出了系統狀態在相平面的運動軌跡 圖 1.6 ，在圖中，紅色的虛線代表在相平面中的滑動面$S=0$，此滑動面可以將相平面分為左側$S<0$以及右側$S>0$，在圖中可以很清楚看出本題非線性系統受到控制後，其狀態運動軌跡先從$S>0$以有限時間收斂到達滑動$S=0$，並且一直待在滑動面上，而在滑動面上的軌跡再以漸近穩定的方式滑向原點。

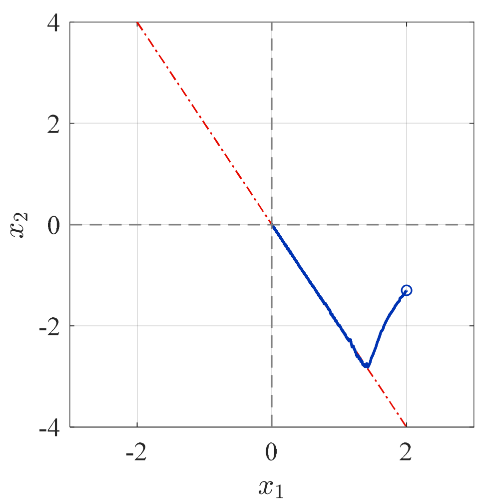


圖 1.6、系統$a=b=1$之相平面狀態運動軌跡

為了驗證滑動模式控制在參數不確定性$|\theta_1|\le 1,\ |\theta_2|\le 1$的假設下，其控制律對於系統皆存在強健性，因此進一步模擬測試在 6 種不同初始值情況，且任意的有界不確定性條件下，其受控系統的響應行為，圖 1.7 為系統滑動變數在 6 次測試下，各組測試的收斂情況，從圖中可以看出，不論初始值為何以及不論參數不確定性為多少，只要參數不確定性滿足本題所設計的有界條件，則滑動變數皆會以有限時間收斂到達滑動

面且不再離開，也因此從狀態相平面 圖 1.8 的運動軌跡來看，也會有相同的結果，從 圖 1.8 中我們可以整理出以下結論，本題所設計的滑動模式控制律對於任意有界的參數不確定性$|\theta_1| \leq 1,\ |\theta_2| \leq 1$，不論初始值為何，皆會收斂到達原點，而其狀態收斂運動軌跡分為兩個階段: 1)迫近階段(Approaching Phase): 系統狀態軌跡以有限時間收斂到達滑動面，2)滑動階段(Sliding Phase): 系統狀態被限制在滑動面上，且在滑動面上以漸進穩定方式收斂到達原點。

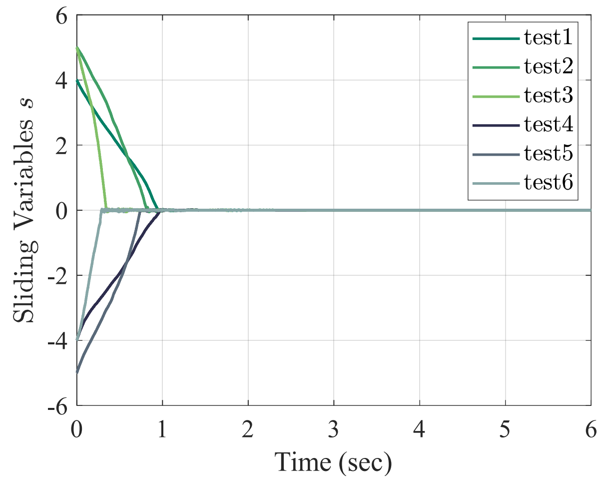


圖 1.7、測試不同初始值之滑動變數有限時間收斂響應

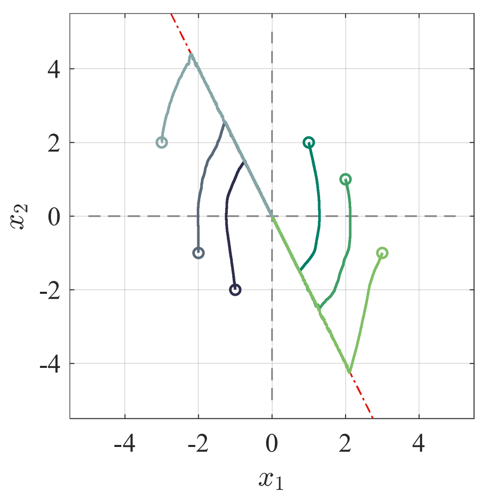


圖 1.8、測試不同初始值之系統相平面狀態運動軌跡

**Question (c)**

設定$a=b=2$，重複上面步驟的模擬，並觀察顫振的情況有何改變。

Answer

在(b)小題當中，假設系統不確定性假設有界$|\theta_1|\le 1,\ |\theta_2|\le 1$，而在本題將系統不確定性的假設為有界$|\theta_1|\le 2,\ |\theta_2|\le 2$，其餘控制律固定切換增益$b_0=1$不變，系統初始值也不變，而不確定性的時間變化如 圖 1.9 所示。

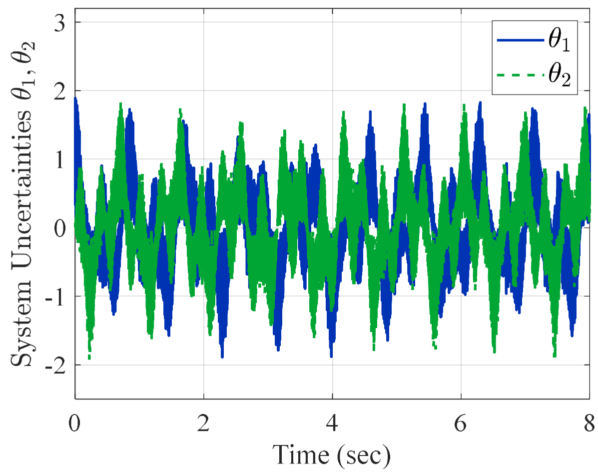



圖 1.9、系統$a=b=2$之參數不確定性時間變化圖

重複以上設計步驟，不確定性界限的改變，不連續的控制律一樣能使得系統狀態收斂到達原點，其時間響應如 圖 1.10。

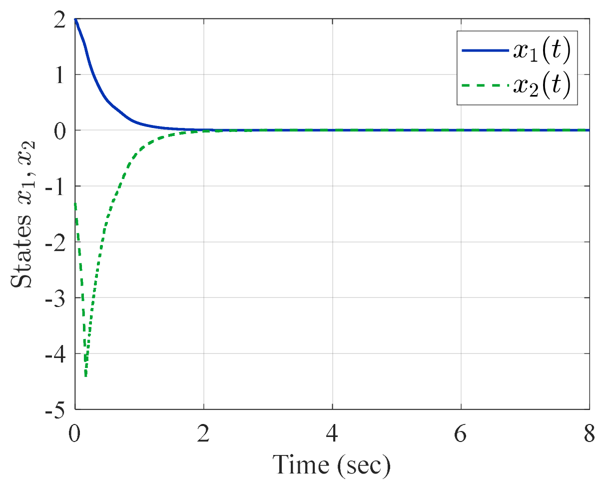



圖 1.10、系統$a=b=2$之狀態收斂時間響應

一樣透過穩定性分析(1.22)，滑動變數受到一連串不連續的控制訊號 圖 1.12 控制後，會在有限時間內收斂到達滑動面上，其時間響應如 圖 1.11 所示，在圖中一樣可以看到顫振的現象，且由於本題的固定切換增益設計相同為$b_0 = 1$，因此不連續控制訊號在經過一段時間後，會在正負 1 之間做切換。

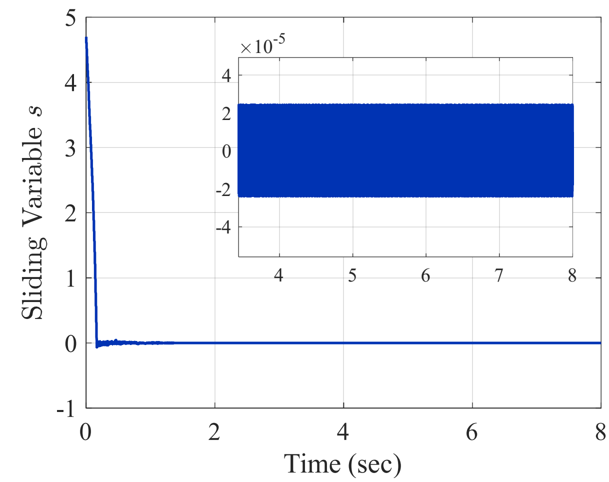



圖 1.11、系統$a = b = 2$之滑動變數於有限時間收斂響應

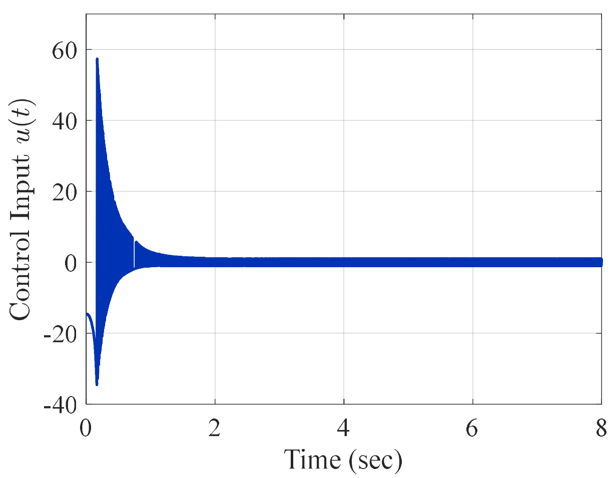



圖 1.12、系統$a = b = 2$之滑動模式不連續控制輸入訊號

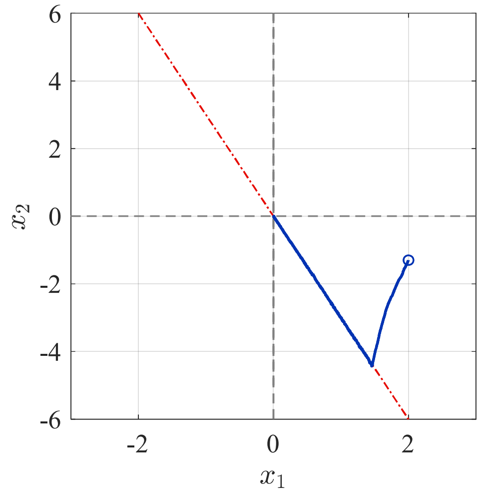


圖 1.13、系統$a=b=2$之相平面狀態運動軌跡

圖 1.13 為在參數不確定界限假設為$|\theta_1|\le 2,\ |\theta_2|\le 2$，設計滑動模式控制，系統在相平面的運動軌跡，在圖中紅色的虛線為滑動面$S=0$，值得注意的是由於滑動面被設計為$S=(1+a)x_1+x_2=0$，也因此在(c)小題的滑動面必須代入$a=2$，與(b)小題的滑動面代入$a=1$不為相同，然而系統的狀態運動軌跡一樣可以在有限時間內進入滑動面且維持在上面。

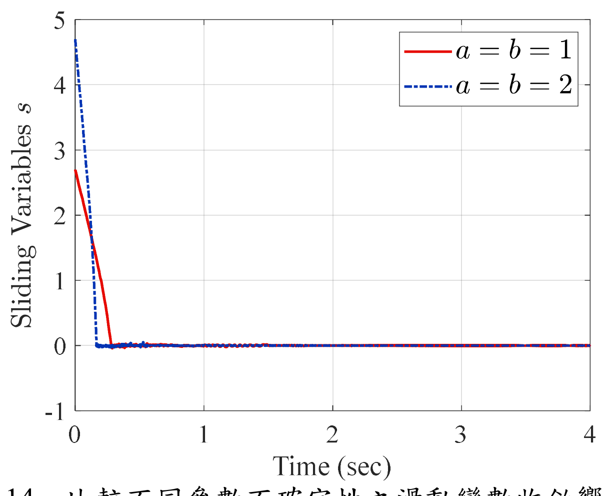
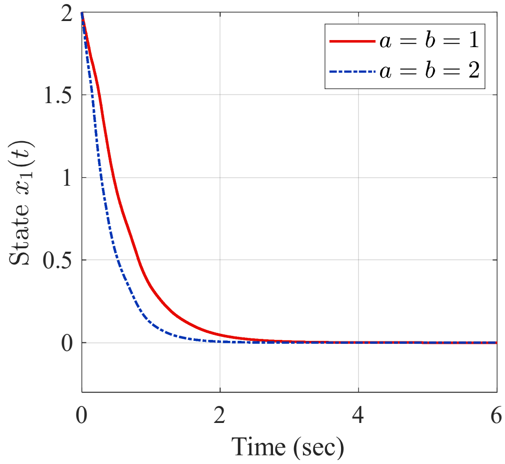
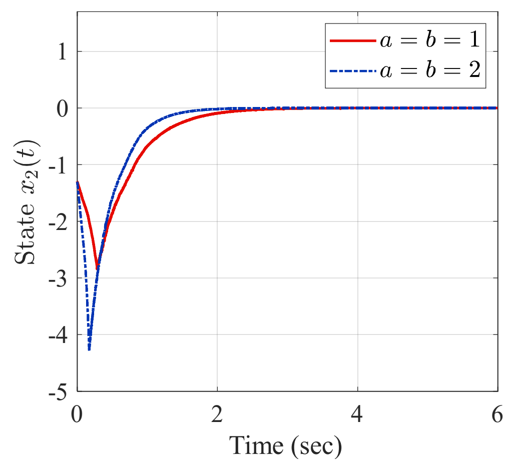


圖 1.14、比較不同參數不確定性之滑動變數收斂響應

圖 1.14 為(c)小題假設$|\theta_1|\le 2,\ |\theta_2|\le 2$以及(b)小題假設$|\theta_1|\le 1,\ |\theta_2|\le 1$，兩者的滑動變數有限時間收斂情形之比較，可以由圖中看出在不確定性界限設置為$a=b=2$時，滑動變數進入滑動面的有限時間較短，而仔細比較兩者在滑動面的顫振情形，差異並不明顯，因為顫振的差異主要是由固定切換增益所影響，然而這兩題的固定切換增益皆設計為$b_0=1$。




a) 狀態 $x_1$ 收斂時間響應




b) 狀態 $x_2$ 收斂時間響應

圖 1.15、比較不同參數不確定性之系統狀態收斂時間響應

圖 1.15 為(c)小題假設$|\theta_1| \le 2,\ |\theta_2| \le 2$以及(b)小題假設$|\theta_1| \le 1,\ |\theta_2| \le 1$，兩者分別對於狀態時間響應收斂情形比較，由圖中看出在不確定性界限設置為$a = b = 2$時，狀態響應的收斂速度較快。

# Matlab Code

| 第一題 - (b) |
|---|

```
%% Nonlinear Control HW8 - Q1 - (b-1)
clear ; clc ; close all
hz = 2000 ;
dt = 1/hz ; t_final = 8 ;
t = 0 : dt : t_final ;
x1_0 = 2 ;    % Initial of x1
x2_0 =    -1.3 ;
X_0 = [ x1_0 ; x2_0 ] ;
%% Plot 1
LW1 = 1.6 ;
LW2 = 1 ;
FS1 = 16 ;
FS_lg = 18 ;

[ ts , Xt ] = ode45( @(t,X) NonlinearSystem(X,t) , t, X_0) ;
% [ ts , Xt ] = RK4( @(t,X) NonlinearSystem(X,t) , [0 t_final], X_0,dt) ;
Xt = Xt' ;

for i =1:length(ts)
    [dX(:,i) , theta1(i) , theta2(i) , uc(i) , s(i) ] = NonlinearSystem(Xt(:,i),t(i)) ;
end

f1 = figure ;
plot(ts,Xt(1,:),'-','Color',[0 0.2 0.7],'LineWidth',LW1) ; hold on ;
plot(ts,Xt(2,:),'--','Color',[0 0.65 0.2],'LineWidth',LW1) ;
hs1(1)=legend({'$x_1(t)$','$x_2(t)$'},'Interpreter','latex');
ax1(1) = gca ;
xlabel('Time (sec)') % x label
ylabel('States        $x_1, x_2 $','Interpreter','Latex') % y label
axis normal
grid on

f2 = figure ;
plot(ts,uc,'-','Color',[0 0.2 0.7],'LineWidth',LW1) ; hold on ;
```

```
ax1(2) = gca ;
xlabel('Time (sec)') % x label
ylabel('Control Input $u(t)$','Interpreter','Latex') % y label
axis normal
grid on

f3 = figure ;
plot(ts,s,'-','Color',[0 0.2 0.7],'LineWidth',LW1) ; hold on ;
ax1(3) = gca ;
xlabel('Time (sec)') % x label
ylabel('Sliding Variable $s$','Interpreter','Latex') % y label
axis normal
grid on

f4 = figure ;
plot(ts,theta1,'-','Color',[0 0.2 0.7],'LineWidth',LW1) ; hold on ;
plot(ts,theta2,'--','Color',[0 0.65 0.2],'LineWidth',LW1) ;
hs1(2)=legend({'$\theta_1$','$\theta_2$'},'Interpreter','latex');
ax1(4) = gca ;
xlabel('Time (sec)') % x label
ylabel('System Uncertainties $\theta_1, \theta_2 $','Interpreter','Latex') % y label
axis([0 t_final -1.2 1.6])
axis normal
grid on

f5 = figure ;
x1s = -3:0.1:3 ;
x2s = -(1+1).*x1s ;
plot(x1s,x2s,'-.','Color',[0.9 0.04 0],'LineWidth',LW2) ; hold on ;
plot(Xt(1,:),Xt(2,:),'Color',[0 0.2 0.7],'LineWidth',LW1)
plot(Xt(1,1),Xt(2,1),'o','Color',[0 0.2 0.7],'LineWidth',LW2) ;
plot([-5 5],[0 0],'--','Color',[0.5 0.5 0.5],'LineWidth',LW2) ;
plot([0 0],[-5 55],'--','Color',[0.5 0.5 0.5],'LineWidth',LW2) ;
ax1(5) = gca ;
xlabel('$x_1$','Interpreter','Latex') % x label
ylabel('$x_2$','Interpreter','Latex') % y label
axis([ -3 3 -4 4 ])
axis square
```

```
grid on

for i = 1:length(ax1)
    set(ax1(i),'FontSize',FS1,'FontName','Times New Roman')
end

for i = 1:length(hs1)
    set(hs1(i),'FontSize',FS_lg,'FontName','Times New Roman')
end

%% Nonlinear System Function
function [dX , theta1 , theta2 , uc , s ] = NonlinearSystem(X,t)
    a = 1 ; b = 1 ;
    b0 = 1 ;

    x1 = X(1) ; x2 = X(2) ;
    s = (1+a)*x1 + x2 ;      % Sliding variable
    %===== Control Input
    beta = a*(1+a)*abs(x1) + b*(x2^2) + b0 ;
    uc = -x1 - (1+a)*x2 - beta*(sign(s)) ;

    %===== Nonlinear System
    theta1 = 1/2*(rand(1)*cos(15*t)+rand(1)*cos(7*t)) ;     % System uncertainty
    theta2 = 1/2*(-rand(1)*sin(7*t)+rand(1)*sin(20*t)) ;    % System uncertainty
    dx1 = x2 + theta1*x1*sin(x2) ;
    dx2 = theta2*(x2^2) + x1 + uc ;
    dX = [ dx1 ; dx2 ] ;
end

%% Nonlinear Control HW8 - Q1 - (b-2)
clear ; clc ; close all
hz = 2000 ;
dt = 1/hz ; t_final = 6 ;
t = 0 : dt : t_final ;
X_0_1 = [1   ; 2] ; X_0_2 = [2   ; 1] ; X_0_3 = [3   ; -1] ;
X_0_4 = [-1   ; -2] ; X_0_5 = [-2   ; -1] ; X_0_6 = [-3   ; 2] ;
X_0_t = [ X_0_1 , X_0_2 , X_0_3 , X_0_4 , X_0_5 , X_0_6 ] ;
%% Plot 1
```

```
LW1 = 1.6 ;
LW2 = 1 ;
FS1 = 16 ;
FS_lg = 15 ;
Dfcolor1 = summer(5) ;
Dfcolor1 = Dfcolor1(1:3,:) ;
Dfcolor2 = bone(6) ;
Dfcolor2 = Dfcolor2(2:4,:) ;
Dfcolor = [Dfcolor1;Dfcolor2] ;

for i = 1:length(X_0_t(1,:))
%       [ ts , Xt ] = ode45( @(t,X) NonlinearSystem(X,t) , t, X_0_t(:,i)) ;
    [ ts , Xt ] = RK4( @(t,X) NonlinearSystem(X,t) , [0 t_final], X_0_t(:,i),dt) ;
    Xt = Xt' ;
    x1 = Xt(1,:) ;
    x2 = Xt(2,:) ;
    X1(i,:) = x1 ;
    X2(i,:) = x2 ;

    for k =1:length(ts)
        [dX(:,k) , theta1(k) , theta2(k) , uc(k) , s(:,k) ] = NonlinearSystem(Xt(:,k),t(k)) ;
    end
    st(i,:) = s ;
end

figure(1)
for i = 1:length(X_0_t(1,:))
    plot(ts,X1(i,:),'-','Color',Dfcolor(i,:),'LineWidth',LW1) ; hold on ;
end
hs1(1)=legend({'test1','test2','test3','test4','test5','test6'},'Interpreter','latex');
ax1(1) = gca ;
xlabel('Time (sec)') % x label
ylabel('States      $x_1$','Interpreter','Latex') % y label
axis normal
grid on

figure(2)
for i = 1:length(X_0_t(1,:))
```

```
    plot(ts,X2(i,:),'-','Color',Dfcolor(i,:),'LineWidth',LW1) ; hold on ;
end
hs1(2)=legend({'test1','test2','test3','test4','test5','test6'},'Interpreter','latex');
ax1(2) = gca ;
xlabel('Time (sec)') % x label
ylabel('States        $x_2$','Interpreter','Latex') % y label
axis normal
grid on

figure(3) ;
for i = 1:length(X_0_t(1,:))
    plot(ts,st(i,:),'-','Color',Dfcolor(i,:),'LineWidth',LW1) ; hold on ;
end
hs1(3)=legend({'test1','test2','test3','test4','test5','test6'},'Interpreter','latex');
ax1(3) = gca ;
axis([0 t_final -6 6])
xlabel('Time (sec)') % x label
ylabel('Sliding Variables    $s$','Interpreter','Latex') % y label
axis normal
grid on

figure(4) ;
x1s = -4:0.1:4 ;
x2s = -(1+1).*x1s ;
plot(x1s,x2s,'-.','Color',[0.9 0.04 0],'LineWidth',LW2) ; hold on ;
plot([-5 5],[0 0],'--','Color',[0.5 0.5 0.5],'LineWidth',LW2) ;
plot([0 0],[-5 5],'--','Color',[0.5 0.5 0.5],'LineWidth',LW2) ;
for i = 1:length(X_0_t(1,:))
    plot(X1(i,:),X2(i,:),'-','Color',Dfcolor(i,:),'LineWidth',LW1) ;
    plot(X1(i,1),X2(i,1),'o','Color',Dfcolor(i,:),'LineWidth',LW1) ;
end
ax1(4) = gca ;
xlabel('$x_1$','Interpreter','Latex') % x label
ylabel('$x_2$','Interpreter','Latex') % y label
axis([ -5.5 5.5 -5.5 5.5 ])
ax1(4).XTick = [-4 -2 0 2 4] ;
ax1(4).YTick = [-4 -2 0 2 4] ;
axis square
```

```
grid on

for i = 1:length(ax1)
    set(ax1(i),'FontSize',FS1,'FontName','Times New Roman')
end

for i = 1:length(hs1)
    set(hs1(i),'FontSize',FS_lg,'FontName','Times New Roman')
end

%% Nonlinear System Function
function [dX , theta1 , theta2 , uc , s ] = NonlinearSystem(X,t)
    a = 1 ; b = 1 ;
    b0 = 1 ;

    x1 = X(1) ; x2 = X(2) ;
    s = (1+a)*x1 + x2 ;     % Sliding variable
    %===== Control Input
    beta = a*(1+a)*abs(x1) + b*(x2^2) + b0 ;
    uc = -x1 - (1+a)*x2 - beta*(sign(s)) ;

    %===== Nonlinear System
    theta1 = 1/2*(rand(1)*cos(15*t)+rand(1)*cos(7*t)) ;     % System uncertainty
    theta2 = 1/2*(-rand(1)*sin(7*t)+rand(1)*sin(20*t)) ;    % System uncertainty
    dx1 = x2 + theta1*x1*sin(x2) ;
    dx2 = theta2*(x2^2) + x1 + uc ;
    dX = [ dx1 ; dx2 ] ;
end
```

| 第一題 - (c) |
| --- |

```
%% Nonlinear Control HW8 - Q1 - (c)
clear ; clc ; close all
hz = 1000 ;
dt = 1/hz ; t_final =8;
t = 0 : dt : t_final ;
x1_0 = 2 ;   % Initial of x1
```

```
x2_0 =    -1.3 ;
X_0 = [ x1_0 ; x2_0 ] ;
%% Plot 1
LW1 = 1.6 ;
LW2 = 1 ;
FS1 = 16 ;
FS_lg = 18 ;

[ ts , Xt1 ] = ode45( @(t,X) NonlinearSystem1(X,t) , t, X_0) ;
[ ts , Xt2 ] = ode45( @(t,X) NonlinearSystem2(X,t) , t, X_0) ;
% [ ts , Xt1 ] = RK4( @(t,X) NonlinearSystem(X,t) , [0 t_final], X_0,dt) ;
% [ ts , Xt2 ] = RK4( @(t,X) NonlinearSystem(X,t) , [0 t_final], X_0,dt) ;
Xt1 = Xt1' ;
Xt2 = Xt2' ;

for i =1:length(ts)
    [dX1(:,i) , theta1_1(i) , theta1_2(i) , uc1(i) , s1(i) ] = NonlinearSystem1(Xt1(:,i),t(i)) ;
end
for i =1:length(ts)
    [dX2(:,i) , theta2_1(i) , theta2_2(i) , uc2(i) , s2(i) ] = NonlinearSystem2(Xt2(:,i),t(i)) ;
end

f1 = figure ;
plot(ts,Xt1(1,:),'-','Color',[0 0.2 0.7],'LineWidth',LW1) ; hold on ;
plot(ts,Xt1(2,:),'--','Color',[0 0.65 0.2],'LineWidth',LW1) ;
hs1(1)=legend({'$x_1(t)$','$x_2(t)$'},'Interpreter','latex');
ax1(1) = gca ;
xlabel('Time (sec)') % x label
ylabel('States      $x_1, x_2 $','Interpreter','Latex') % y label
axis normal
grid on

f2 = figure ;
plot(ts,uc1,'-','Color',[0 0.2 0.7],'LineWidth',LW1) ; hold on ;
ax1(2) = gca ;
xlabel('Time (sec)') % x label
ylabel('Control Input $u(t)$','Interpreter','Latex') % y label
axis normal
```

```
grid on

f3 = figure ;
plot(ts,s1,'-','Color',[0 0.2 0.7],'LineWidth',LW1) ; hold on ;
ax1(3) = gca ;
xlabel('Time (sec)') % x label
ylabel('Sliding Variable $s$','Interpreter','Latex') % y label
axis normal
grid on

f4 = figure ;
plot(ts,theta1_1,'-','Color',[0 0.2 0.7],'LineWidth',LW1) ; hold on ;
plot(ts,theta1_2,'--','Color',[0 0.65 0.2],'LineWidth',LW1) ;
hs1(2)=legend({'$\theta_1$','$\theta_2$'},'Interpreter','latex');
ax1(4) = gca ;
xlabel('Time (sec)') % x label
ylabel('System Uncertainties $\theta_1, \theta_2 $','Interpreter','Latex') % y label
axis([0 t_final -2.2 2.6])
axis normal
grid on

f5 = figure ;
x1_1s = -4:0.1:4 ;
x1_2s = -(1+2).*x1_1s ;
plot(x1_1s,x1_2s,'-.','Color',[0.9 0.04 0],'LineWidth',LW2) ; hold on ;
plot(Xt1(1,:),Xt1(2,:),'Color',[0 0.2 0.7],'LineWidth',LW1)
plot(Xt1(1,1),Xt1(2,1),'o','Color',[0 0.2 0.7],'LineWidth',LW2) ;
plot([-5 5],[0 0],'--','Color',[0.5 0.5 0.5],'LineWidth',LW2) ;
plot([0 0],[-6 6],'--','Color',[0.5 0.5 0.5],'LineWidth',LW2) ;
ax1(5) = gca ;
xlabel('$x_1$','Interpreter','Latex') % x label
ylabel('$x_2$','Interpreter','Latex') % y label
axis([ -3 3 -5 5 ])
axis square
grid on

f6 = figure ;
plot(ts,s2,'-','Color',[0.9 0.04 0],'LineWidth',LW1)    ; hold on ;
```

```
plot(ts,s1,'-.','Color',[0 0.2 0.7],'LineWidth',LW1) ;
hs1(3)=legend({'$a=b=1$','$a=b=2$'},'Interpreter','latex');
ax1(6) = gca ;
xlabel('Time (sec)') % x label
ylabel('Sliding Variables $s$','Interpreter','Latex') % y label
axis normal
grid on

f7 = figure ;
plot(ts,Xt2(1,:),'-','Color',[0.9 0.04 0],'LineWidth',LW1)    ; hold on ;
plot(ts,Xt1(1,:),'-.','Color',[0 0.2 0.7],'LineWidth',LW1) ;
hs1(4)=legend({'$a=b=1$','$a=b=2$'},'Interpreter','latex');
ax1(7) = gca ;
xlabel('Time (sec)') % x label
ylabel('State    $x_1(t)$','Interpreter','Latex') % y label
axis normal
grid on

f8 = figure ;
plot(ts,Xt2(2,:),'-','Color',[0.9 0.04 0],'LineWidth',LW1)    ; hold on ;
plot(ts,Xt1(2,:),'-.','Color',[0 0.2 0.7],'LineWidth',LW1) ;
hs1(5)=legend({'$a=b=1$','$a=b=2$'},'Interpreter','latex');
ax1(8) = gca ;
xlabel('Time (sec)') % x label
ylabel('State    $x_2(t)$','Interpreter','Latex') % y label
axis normal
grid on

for i = 1:length(ax1)
    set(ax1(i),'FontSize',FS1,'FontName','Times New Roman')
end

for i = 1:length(hs1)
    set(hs1(i),'FontSize',FS_lg,'FontName','Times New Roman')
end

%% Nonlinear System Function
function [dX , theta1 , theta2 , uc , s ] = NonlinearSystem1(X,t)
```

```
    a = 2 ; b = 2 ;
    b0 = 1 ;

    x1 = X(1) ; x2 = X(2) ;
    s = (1+a)*x1 + x2 ;       % Sliding variable
    %===== Control Input
    beta = a*(1+a)*abs(x1) + b*(x2^2) + b0 ;
    uc = -x1 - (1+a)*x2 - beta*(sign(s)) ;

    %===== Nonlinear System
    theta1 = (rand(1)*cos(15*t)+rand(1)*cos(7*t)) ;      % System uncertainty
    theta2 = (-rand(1)*sin(7*t)+rand(1)*sin(20*t)) ;     % System uncertainty
    dx1 = x2 + theta1*x1*sin(x2) ;
    dx2 = theta2*(x2^2) + x1 + uc ;
    dX = [ dx1 ; dx2 ] ;
end

function [dX , theta1 , theta2 , uc , s ] = NonlinearSystem2(X,t)
    a = 1 ; b = 1 ;
    b0 = 1 ;

    x1 = X(1) ; x2 = X(2) ;
    s = (1+a)*x1 + x2 ;       % Sliding variable
    %===== Control Input
    beta = a*(1+a)*abs(x1) + b*(x2^2) + b0 ;
    uc = -x1 - (1+a)*x2 - beta*(sign(s)) ;

    %===== Nonlinear System
    theta1 = 1/2*(rand(1)*cos(15*t)+rand(1)*cos(7*t)) ;      % System uncertainty
    theta2 = 1/2*(-rand(1)*sin(7*t)+rand(1)*sin(20*t)) ;     % System uncertainty
    dx1 = x2 + theta1*x1*sin(x2) ;
    dx2 = theta2*(x2^2) + x1 + uc ;
    dX = [ dx1 ; dx2 ] ;
end
```