

スライディングモード(Slidingmode)の基本的な考え方

(参考文献: 野波, 田, スライディングモード制御 (非線形ロバスト制御の設計理論), コロナ社, 1994)

2 次システム

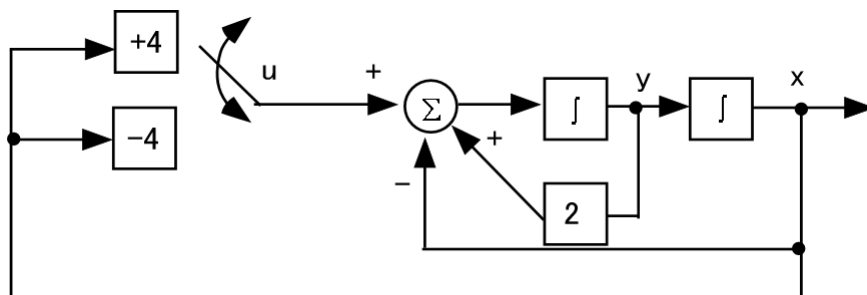
$$\begin{cases} \dot{x} = y \\ \dot{y} = 2y - x + u \\ u = -\varphi x \end{cases} \quad (1-1)$$

とし, フィードバックゲインを次のように選ぶ.

$$\varphi = \begin{cases} 4 & \sigma(x,y) > 0 \\ -4 & \sigma(x,y) < 0 \end{cases} \quad (1-2)$$

ここで, $\sigma(x,y)$ は切換関数とよばれ, 次のように設計する.

$$\sigma(x,y) = xS \quad S = 0.5x + y \quad (1-3)$$



ところで, 切換関数 $\sigma(x,y)$ は 2 つの関数 x と S の積で,

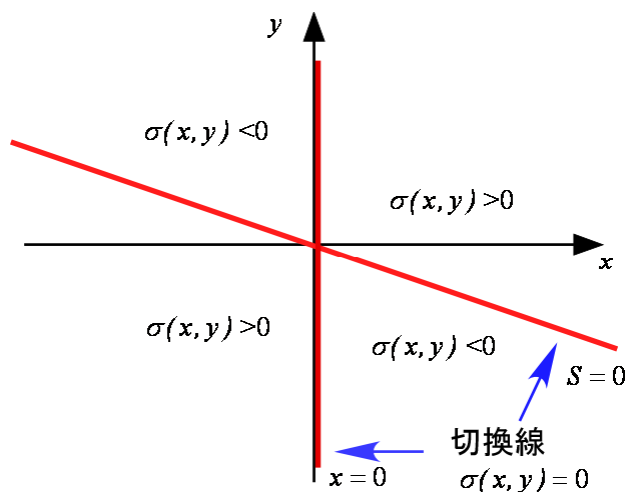
$$\sigma(x,y) = 0 \quad (1-4)$$

を満たすとき

$$x = 0 \quad \text{あるいは} \quad S = 0.5x + y = 0 \quad (1-5)$$

である. なお, 式(1-5)は切換線と呼ばれている.

また, 式(1-5)の 2 つの関数は, 図に示すように位相平面を $\sigma(x,y)$ の符号が異なるように分割している.



フィードバックゲインは式(1-2)に示しているように $\sigma(x,y)$ の符号によって切り替えられ、次の微分方程式によって、2つの位相平面の領域が定義される。すなわち、

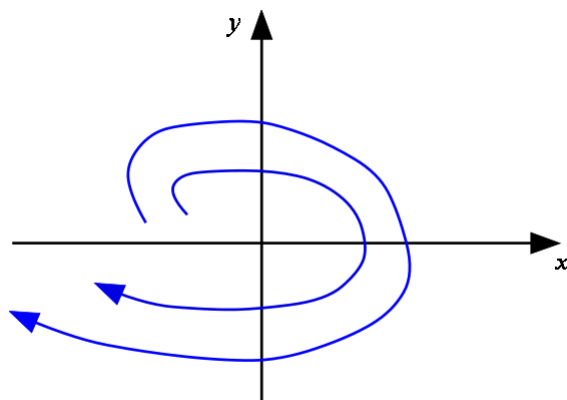
領域Ⅰ $\sigma(x,y) = xS > 0$ のとき

$$\begin{cases} \dot{x} = y \\ \dot{y} = 2y - x - 4x = 2y - 5x \end{cases} \quad (1-6)$$

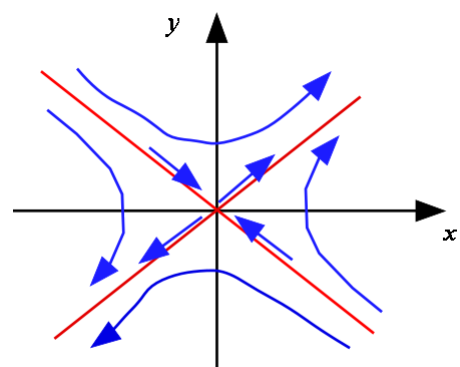
領域Ⅱ $\sigma(x,y) = xS < 0$ のとき

$$\begin{cases} \dot{x} = y \\ \dot{y} = 2y - x + 4x = 2y + 3x \end{cases} \quad (1-7)$$

であり、式(1-6)、式(1-7)それぞれの位相平面軌跡で表せば次のようになる。



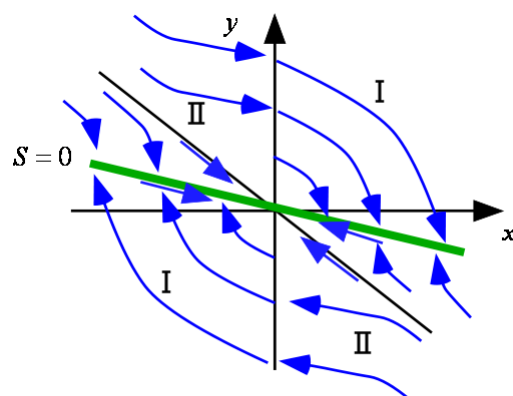
領域Ⅰの位相平面軌跡



領域Ⅱの位相平面軌跡

領域Ⅰの位相平面軌跡は、任意の初期値から出発して平衡点に留まることなく、不安定渦状点(unstable spiral point)の挙動を示しており、領域Ⅱでは原点において平衡点がみられるが、状態軌道は鞍形点(saddle point)であり、不安定な軌跡である。

上図をもとに $\sigma(x,y)$ を考慮して、領域Ⅰと領域Ⅱを併せて軌跡を描くと下図となる。



領域ⅠとⅡの位相平面軌跡

ここで,

$$S = 0.5x + y = 0.5x + \dot{x} = 0 \quad (1-8)$$

は, 切換線つまり微分方程式そのものであるが, 最終的な軌跡はこの切換線に沿った軌跡になっていることがわかる.

また, この図の意味は, 一般にシステムの位相空間上での挙動は 2 つのモードに分けられ, 1 つは, 到達モード呼ばれ, 非スライディングモードであり, 位相平面上の任意の場所から出発した軌跡は切換線 $S = 0$ に向かって動き, 有限時間で切換線に到達する. もう 1 つはスライディングモードで, 式(1-8)で定義した微分方程式に従い, 位相平面上の原点に漸近的に近づいていく. 要するに任意の場所からこの直線上に拘束されて, 平衡点つまり原点に滑っていくこの挙動が滑り状態(sliding mode)であり, これをスライディングモードとよぶ.

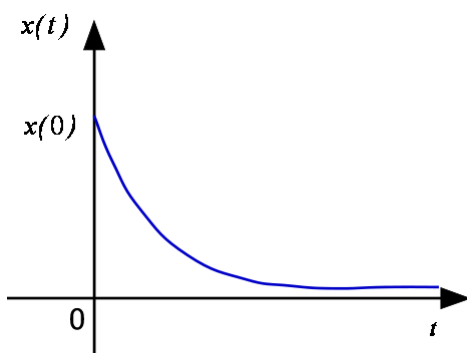
この切換線に乗せればなぜ, 安定な平衡点 (原点) に滑っていくのだろうか. それは, この切換線, 式(1-8)の微分方程式の解を求めてみれば明らかとなる. ちなみに

$$0.5x + \dot{x} = 0 \quad \text{あるいは} \quad \dot{x} = -0.5x \quad (1-9)$$

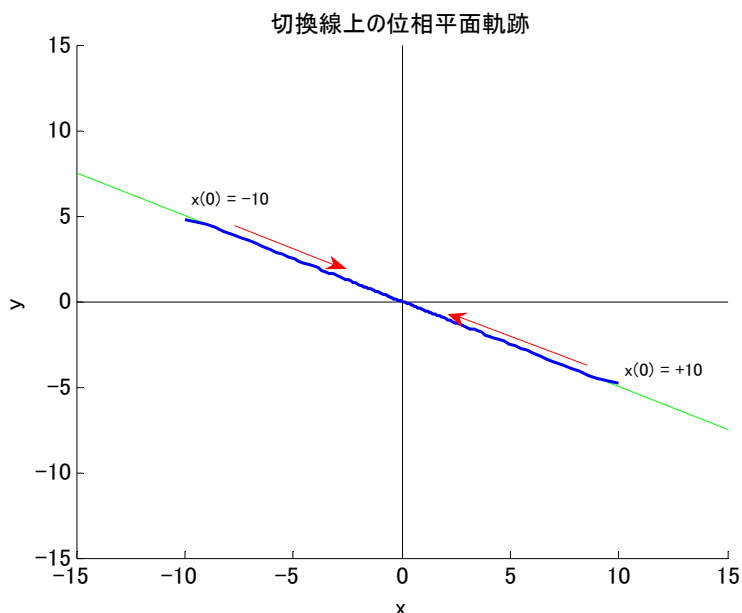
の解は

$$x(t) = x(0)e^{-0.5t} \quad (1-10)$$

である. この関数の挙動は下図から, 0 に近づくことより安定であることがわかる.



また, 式(1-9)の切換線上の位相平面において, この線上に拘束された任意の点は常に原点に収束していくことが下図より明らかである.



以下に MATLAB のプログラムを示す.

```
%-----  
%      スライディングモード制御(コロナ社)  
%      非線形ロバスト制御の設計理論(野波健蔵)  
%      P3  
%      切換線  $x' = -0.5x$   
%      Notice!  
%      call slidingmode_ren2_eq2.m  
%      2006. 6. 13                                produced by kizawa  
%-----  
  
clc  
clear all  
format long e  
global ni;  
global dy; %xの微分値  
%----setup----%  
sampling=0.1;%sampling time of simulation  
%kaisu=1000;  
%timespan=sampling*kaisu;% simulation time  
%--- initial state ----  
x0=[-10 ]; %initial condition  
disp('It is under calculation now!!')  
%===== 初期値条件1=====  
ni=1;  
tend=1000; %6000 応答時間に合わせて変える  
while(ni <= tend)  
    [Tn,Yn]=ode45(@slidingmode_ren2_eq2,[0 sampling],x0,[]); %変えること  
    if ni==1  
        YY(1,:)=Yn(1,1:1);  
        DY(1,:)=dy;  
        TT(1,:)=Tn(1,1);  
    else  
        YY(ni,:)=Yn(size(Yn,1),1:1);  
        DY(ni,:)=dy;  
        TT(ni,1)=sampling*ni;  
    end  
    %---- Up data-----  
    size(Yn,1);  
    x0=Yn(size(Yn,1),1:1);  
    ni=ni+1;  
end
```

```

end
hold on
figure(1)
plot([-15; 15], [-0.5*(-15); -0.5*15], 'g');
plot([-15 15], [0 0], 'k')
plot([0 0], [15 -15], 'k')
plot(YY(:, 1), DY, 'b', 'linewidth', 2);
ax1=findobj(gca, 'Type', 'Axes', 'Visible', 'on');
set(ax1, 'fontsize', 12); %目盛りを大きく
axis([-15 +15 -15 +15])
xlabel('x', 'fontsize', 12);
ylabel('y', 'fontsize', 12)
title('切換線上の位相平面軌跡', 'fontsize', 12)
text(-10, 6, 'x(0) = -10');
%===== 初期値条件2=====
%--- initial state ----
x0=[ +10 ]; %initial condition
ni=1;
tend=1000; %6000 応答時間に合わせて変える
while(ni <= tend)
    [Tn, Yn]=ode45(@slidingmode_ren2_eq2, [0 sampling], x0, []); %変えること
    if ni==1
        YY(1, :)=Yn(1, 1:1);
        DY(1, :)=dy;
        TT(1, :)=Tn(1, 1);
    else
        YY(ni, :)=Yn(size(Yn, 1), 1:1);
        DY(ni, :)=dy;
        TT(ni, 1)=sampling*ni;
    end
    %---- Up data-----
    size(Yn, 1);
    x0=Yn(size(Yn, 1), 1:1);
    ni=ni+1;
end
plot(YY(:, 1), DY, 'b', 'linewidth', 2);
text(+10, -4, 'x(0) = +10');
hold off

```

```
function dy=slidingmode_ren2_eq2(t,y);  
%---2006.6.13---  
global ni  
global dy  
dy=zeros(1,1);  
dy(1,1) = -0.5*y(1,1);
```