Stochastic Extension to Viability Theory

Rüştü Erciyes Karakaya

August 2022, 18

1 Introduction

Basically, the problem explained before was based on finite time observation of non-linear dynamical system and observing the process in terms of the concept of *viability*. Actually, there was a reward-penalty system to create color coded map during observing the process and accordingly, when we consider the problem with different notions[1] then we can say that *Viability Kernel* is like *Safe Value Function*-SFV which consist of rewards and the remaining part of the map is just a penalized case.

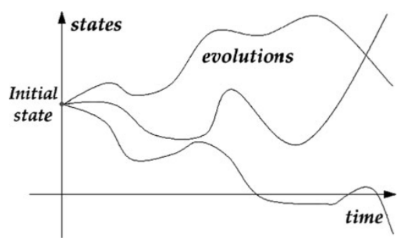


Figure 1: Evolutions (or Trajectories), this the most fundamental summary of our problem actually.

In previous problem, Car on the Hill, was working in the deterministic framework; there was no any probabilistic parameters which affects also our finite time observations. In the book[2], they summarized that:

- "regulatory parameters" in those natural systems where no identified or consensual agent acts on the system.
- "tyches" or disturbances, perturbations under which nobody has any

[1] It can be considered like we revisit the Set-Valued Analysis by using Reinforcement Learning concepts; for further studies: *Massiani, Pierre-François and Heim, Steve and Solowjow, Friedrich and Trimpe, Sebastian. (2021). Safe Value Functions.*

[2] Fundamental Book of Viability Theory: *Aubin, Jean-Pierre and Bayen, Alexandre and Saint-Pierre, Patrick. (2011). Viability Theory: New Directions.*

control.
• “controls”, whenever a controller or a decision maker “pilots” the system by choosing the controls, as in engineering.

Actually, for the basic visualization of Viability Kernel, we omitted the disturbances and control parameters before, but now we are working on these a bit more.

2 Nondeterministic Systems in Viability Theory

In the book, it is stated that there are three different types of uncertainties in Viability Theory framework which are contingent, tychastic or stochastic. In the stochastic framework, dynamical system can be defined as follows:[3]

Dynamical System:

$\frac{dx(t)}{dt} = G(x(t), u(t), \omega(t))$, $\omega \,\epsilon\, \Omega$ and $u(t) \,\epsilon\, U(x(t)) \subset R^m$

$$Parameters = \begin{cases} x(t) := \text{regulatory parameter} \\ u(t) := \text{control parameter} \\ \omega(t) := \text{probabilistic parameter} \end{cases}$$

Viable Scenario: $(\omega(\cdot) \,\epsilon\, \Omega)$

$$\Omega_{\hat{u},x_0} = \begin{cases} \frac{dx(t)}{dt} = G(x(t), u(t), \omega(t)), & x(0) = x_0 \\ u(t) = \hat{u}(t, x(t)) & \\ I(t, x(t), u(t)) \geq \eta_k, & k = 1, 2, 3..q \\ t \epsilon R^+ & \end{cases}$$

Viability Probability: $\Pi(\hat{u}, \eta)$=$\mathbb{P}(\{\omega(\cdot) \,\epsilon\, \Omega_{\hat{u},x_0}\})$

To make job easier, we will start with “Probability” concept as measure in Viability by investigating the *most viable initial states* in stochastic framework. The framework defined above is working with the set of constraints K which consist of random constraints. However, in the following example, you will see that there is a constant K working with regulatory, control and probabilistic parameter within the discrete dynamical system and we observed that which initial state are most viable probabilistically. To do that we will use the concept of *stochastic viability value function* $V(x_0)$:[4]

$$V(x_0) = max_{u \epsilon U}\ \mathbb{E}_{\mathbb{P}}[\pi(t_0, \omega(\cdot), x(\cdot), u(\cdot))]$$

[3] *Optimal Control Approaches to Sustainability under Uncertainty*, DEOS Working Papers 2215, Athens University of Economics and Business.

[4] For further investigation about this concept: *Luc, Doyen and De Lara, Michel. (2010). Stochastic viability and dynamic programming. Systems and Control Letters.*

3 Basic Example-I, Discrete Time Stochastic Viability Case

Discrete Dynamical System:

$$x(\text{t+1})=x(\text{t})+u(\text{t})+\omega(\text{t})$$

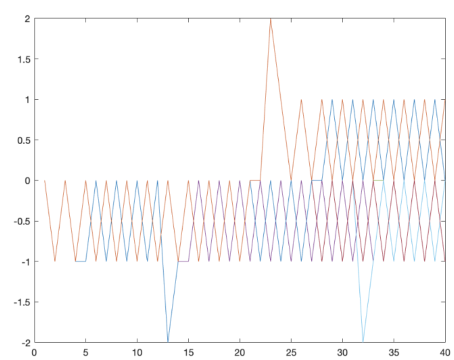
$$Parameters = \begin{cases} x(t)\epsilon\{-1,0,1\}, \text{ state constraints;} \\ u(t)\epsilon\{-1,1\}, \text{ control parameter;} \\ \omega(t)\epsilon\{-1,0,1\}, \text{ perturbance.} \end{cases}$$

Control decision: if x(t)=1, take u(t)=1 so that x(t)+u(t)=0 (the same with x(t) = 1 and u(t) = 1)
Distribution of perturbance: it's simulated from;
$\mu(\omega(\text{t})=1)=\mu(\omega(\text{t})=\text{-}1)=\text{p}$; $\mu(\omega(\text{t})=0)=1\text{-}2\text{p}$, with parameter p.

Aim: $\mathbb{P}[\ x(t)\epsilon\{-1,0,1\},\ \text{t}=t_0,...,\text{T}\ ] \geq \beta$, i.e determine the most viable case within the confidence level β.

Simulation:[5] For the any specific initial state we started the 9 different trajectories and observe which them are viable or penalized. To do that, we can determine the stochastic viability value function $V(x_0)$.

Outcomes:




```
V = 0.3333
```

Figure 2: Three of the trajectories are penalized out of 9 at finite time, so stochastic viability value function $V(x_0 = 0)$ is roughly 0.66 for this case.

[5] Matlab codes are available for this example at the end of the document, please check them!

4 Example-II: Car on the Hill[6]

$$f(x(t),u)=\dot{x}=\begin{pmatrix}\dot{x_1}\\ \dot{x_2}\end{pmatrix}=\begin{pmatrix}x_2+v_1\\ -9.81sin(\frac{dg(x_1)}{dx_1}+v_2)-0.7x_2+u+v_3\end{pmatrix},$$

$$Parameters=\begin{cases}x_1(t)\epsilon(0,12),\text{ state constraint;}\\ u(t)\epsilon(2,2),\text{ control parameter;}\\ v_1(t)\epsilon(0.4,0.4),\text{ 1st uncertainty parameter;}\\ v_2(t)\epsilon(0.07,0.07),\text{ 2nd uncertainty parameter;}\\ v_3(t)\epsilon(0.2,0.2),\text{ 3rd uncertainty parameter.}\end{cases}$$

$v_1(t)$:= uncertainty of the longitudinal velocity;
$v_2(t)$:= uncertainty of the vertical position on the hill;
$v_3(t)$:= uncertainty of the force on the car and all them are iid.
Control decision:

```
if abs(12-y(1))<y(2)&&y(2)>0&&(0<=y(1))<=12
    %set u
    if y(2)+y(1)<=9
        u=-1;
    elseif  y(2)+y(1)>9
        u=-2;
    end
elseif abs(0-y(1))<abs(y(2))&&y(2)<0&&(0<=y(1))<=12
    %set u
    if y(2)+y(1)<=3
        u=2;
    elseif  y(2)+y(1)>3
        u=1;
    end
end
```

Figure 3: y(1)=x_1=s and y(2)=$x_2 = \dot{s}$; and the methodology is based on comparison of distance to terminal points, s=0,12 and velocity, $\dot{s}$.

Outcomes:[7]

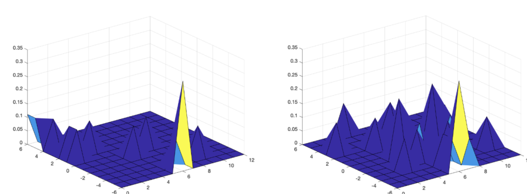


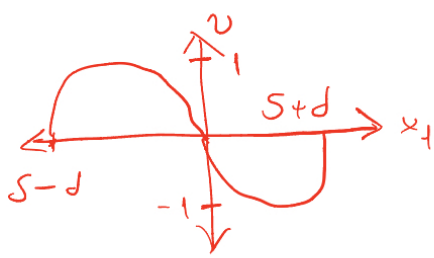
Figure 4: LHS is when u=0, RHS is when control decision is included. In the plane; x,y represent the x_1=s and $x_2 = \dot{s}$ respectively and z refer to the stochastic viability value function.

[6]Aim is still same with previous example, so some details are skipped in this case!
[7]MATLAB codes are available at end of the document, please check there!

5 APPENDIX

How to define more realistic and reasonable control?

If you investigate the Example-II above, the control in there is not defined continuous which is nonrealistic situation actually. Instead these, we stated the continuous and reasonable control as can seen the figure below.




```
%continuous control
%our critical points:=eq_points=[9.5,6.69,3.91];
d=0.3;
if 3.91-d<=y(1)&&3.91+d>=y(1)
   u=-sin((pi/d)*(y(1)-3.91));
elseif 6.69-d<=y(1)&&6.69+d>=y(1)
   u=-sin((pi/d)*(y(1)-6.69));
elseif 9.5-d<=y(1)&&9.5+d>=y(1)
   u=-sin((pi/d)*(y(1)-9.5));
end
```

Figure 5: We try to stabilize the car between longitudinal position S+d and S-d by using the control, u, and this control is started with zero and reached the max during the process which is reasonable and realistic.

In this case, we have three control station on the landscape which is marked the Figure-7 below. The marked points below are actually the equilibrium points, namely our dynamical system is:

$f(x(t), u) = \dot{x} = \begin{pmatrix} \dot{x_1} \\ \dot{x_2} \end{pmatrix} = 0$

These points roughly overlaps with the points of x_1={$9.5, 6.69, 3.91$}, that's why we defined the control around them with tolerance d.

Stability Analysis of Deterministic Case of Car on the Hill with continuous control

The stability is the case that states of trajectories remains the same region like the rhs of the figure below. As stated previously, we have three equilibrium points and we expect that stable regions will constitute around them, which can be seen lhs of the figure below as we expected.

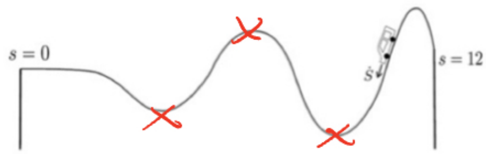


Figure 6: Equilibrium points of the our dynamical system.

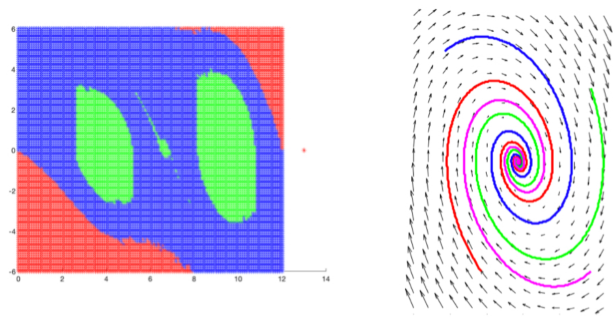


Figure 7: In the lhs, Green region is viable and stable; Blue region is viable but not stable; Red region is not viable and not stable. The lhs is just a basic representation of meaning of stability.

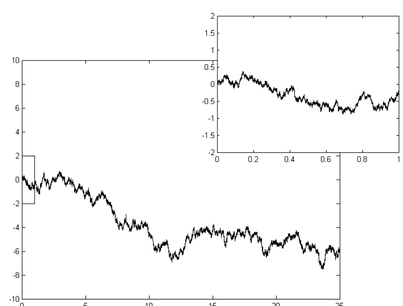


Figure 8: A single realization of a one-dimensional Wiener process, This is what we suggest for relation between dynamical systems and stochastic process in a rigorous way.

At the end of the day, we found the locally stable regions like capture basins. Actually, the problem is about the topology of the Figure-4; in the realistic framework, we expect that it has more smooth topology but it's not so here. We also have to make a more realistic and reasonable definitions for stochastic aspects of the these analysis, the probabilistic parameters defined in the Example-II is not like that so we have to work on more rigorous way. Wiener Process can be showed up at this point actually, so that we may obtain more smooth topology in Figure-4.

Literature suggestions:

• Dynamical Theories of Brownian Motion by Edward Nelson.

• An Introduction to Mathematical Optimal Control Theory by Lawrence C. Evans.

• Set-theoretic Methods in Control by Franco Blanchini.

• Tanaka, Takashi and Sandberg, Henrik and Skoglund, Mikael. (2021). Transfer-Entropy-Regularized Markov Decision Processes. IEEE Transactions on Automatic Control.

• Aubin, Jean-Pierre and Bayen, Alexandre and Saint-Pierre, Patrick. (2011). Viability Theory: New Directions.

• Optimal Control Approaches to Sustainability under Uncertainty, DEOS Working Papers 2215, Athens University of Economics and Business.

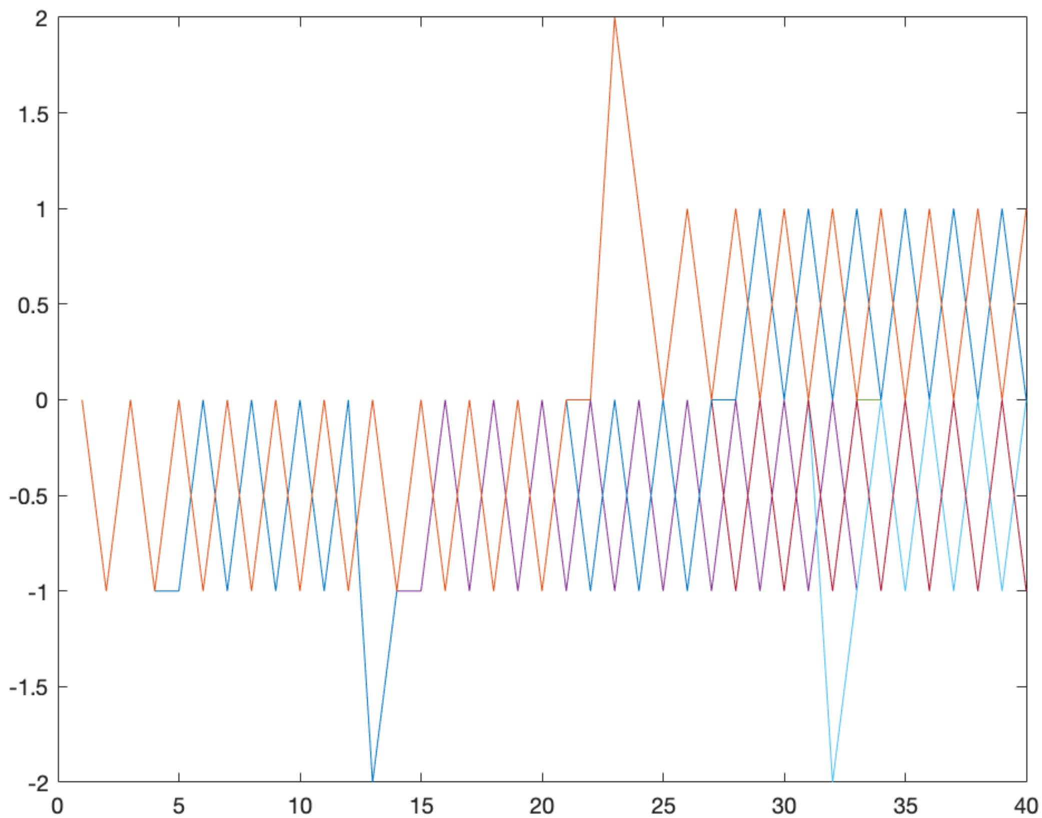
• Luc, Doyen and De Lara, Michel. (2010). Stochastic viability and dynamic programming. Systems and Control Letters.

• Massiani, Pierre-Francois and Heim, Steve and Solowjow, Friedrich and Trimpe, Sebastian. (2021). Safe Value Functions.

For the MATLAB part of the APPENDIX, please visit the following pages!

```matlab
%Basic Example-I, Discrete Time Stochastic Viability Case
clear all;
%we start with # of 9 Trajectories(= Finite Time Observation, Evolution)
%t=40, finite time
trajectories=zeros(9,40);
number_of_penalty=0;number_of_trajectories=9;
for j=1:number_of_trajectories
    %initial_settings
    x(1)=0;reward=true;
    trajectories(j,1)=x(1);
    u=1;p=0.01;
    P=[p p 1-2*p];
    W=[1 -1 0];

    for i=2:40

    %assign the control
    if u+x(i-1)>0
        u=-1;
    elseif u+x(i-1)<0
        u=1;
    end

    %uncertainty
    w=randsample(W,1,true,P);

    %The evolution of a scalar x(t), discrete-time dynamics
    x(i)=x(i-1)+u+w;
    trajectories(j,i)=x(i);

    %check viable or not (penalized or reward)
    if x(i)>=-1 && x(i)<=1
        %reward
    else
        %penalty
        reward=false;
    end

    end

    if reward==false
       number_of_penalty=number_of_penalty+1;
    end

end
```

```
t=1:1:40;
for m=1:number_of_trajectories
   plot(t,trajectories(m,:))
   hold on
end
```

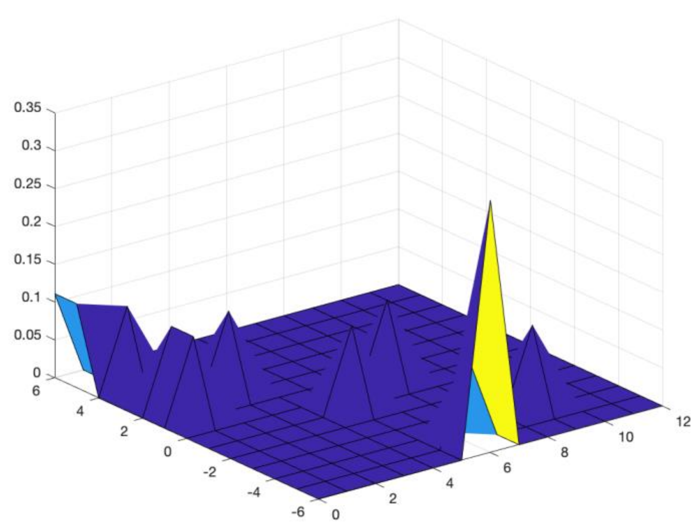



```
%Viability probability value function:= V(0=x)
V=number_of_penalty/number_of_trajectories
```

V = 0.3333

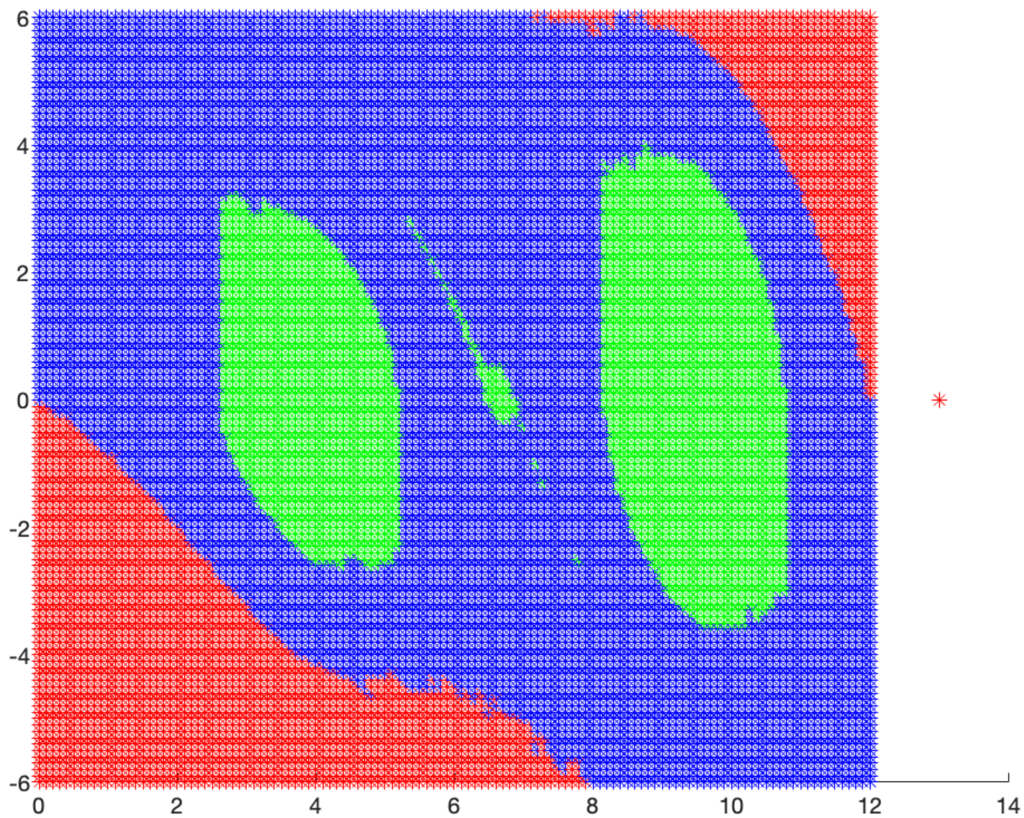
```
```

```
%stochastic car on the hill with control parameter
tspan= 0:1:30;
number_of_penalty=0;number_of_trajectories=9;
m=1;
for i=0:1:12
    n=1;
    for j=-6:1:6
        number_of_penalty=0;
        for k=1:number_of_trajectories
            [t,y] = ode15s(@vdp_Car,tspan,[i;j]);
            A=(0>y(:,1)); B=(y(:,1)>12);
            if A+B==0
                %reward
            else
                %penalty
                number_of_penalty=number_of_penalty+1;
            end
        end
        table(m,n)=1-(number_of_penalty/number_of_trajectories);
        n=n+1;
    end
    m=m+1;
end

x=0:1:12;y=-6:1:6;
z = table(x+1,y+7);
surf(x,y,z)
```



```
function dydt = vdp_Car(t,y)
v1 = unifrnd(-0.4,0.4);v2 = unifrnd(-0.07,0.07);v3 = unifrnd(-0.2,0.2);u=0;
y1=y(2)+v1;
%{
%control
if abs(12-y(1))<y(2)&&y(2)>0&&(0<=y(1))<=12
    %set u
    if y(2)+y(1)<=9
        u=-1;
    elseif  y(2)+y(1)>9
        u=-2;
    end
elseif abs(0-y(1))<abs(y(2))&&y(2)<0&&(0<=y(1))<=12
    %set u
    if y(2)+y(1)<=3
        u=2;
    elseif  y(2)+y(1)>3
        u=1;
    end
end
%}
y2 = -9.81*sin(0.55*sin(1.2*y(1))-0.6*sin(1.1*y(1))+v2)-0.7*y(2)+v3+u;
dydt=[y1;y2];
end
```

```
%Stability Analysis of Deterministic Case of Car on the Hill with continuous control
tspan= 0:0.1:30;
%%{
hold all
bool=false;
m=1;
eq_points=[9.5,6.69,3.91];
d=1.3;
for i=0:0.1:12
    n=1;
    for j=-6:0.1:6
        bool=false;
        [t,y] = ode45(@vdp_Car,tspan,[i;j]);
        A=(0>y(:,1)); B=(y(:,1)>12);
        if A+B==0
            bool=true;
            if(eq_points(1)-d<=y(:,1))&(y(:,1)<=eq_points(1)+d)
                plot(i,j,'*',"Color",'g');
            elseif (eq_points(2)-d<=y(:,1))&(y(:,1)<=eq_points(2)+d)
                plot(i,j,'*',"Color",'g');
            elseif (eq_points(3)-d<=y(:,1))&(y(:,1)<=eq_points(3)+d)
                plot(i,j,'*',"Color",'g');
            else
                plot(i,j,'*',"Color",'b');
            end
        else
            plot(i,j,'*',"Color",'r');
        end
        table(m,n)=bool;
        n=n+1;
    end
    m=m+1;
end
%table
plot(13,0,'*',"Color",'r')
hold off
```




```
function dydt = vdp_Car(t,y)
%v1 = unifrnd(-0.4,0.4);v2 = unifrnd(-0.07,0.07);
%v3 = unifrnd(-0.2,0.2);
u=0;
y1=y(2);
%{
%control
if abs(12-y(1))<y(2)&&y(2)>0&&(0<=y(1))<=12
    %set u
    if y(2)+y(1)<=9
        u=-1;
    elseif  y(2)+y(1)>9
        u=-2;
    end
elseif abs(0-y(1))<abs(y(2))&&y(2)<0&&(0<=y(1))<=12
    %set u
    if y(2)+y(1)<=3
        u=2;
    elseif  y(2)+y(1)>3
        u=1;
```

```
    end
end
%}

%continuous control
%our critical points:=eq_points=[9.5,6.69,3.91];
d=0.3;
if 3.91-d<=y(1)&&3.91+d>=y(1)
   u=-sin((pi/d)*(y(1)-3.91));
elseif 6.69-d<=y(1)&&6.69+d>=y(1)
   u=-sin((pi/d)*(y(1)-6.69));
elseif 9.5-d<=y(1)&&9.5+d>=y(1)
   u=-sin((pi/d)*(y(1)-9.5));
end

y2 = -9.81*sin(0.55*sin(1.2*y(1))-0.6*sin(1.1*y(1)))-0.7*y(2)+u;
dydt=[y1;y2];
end
```