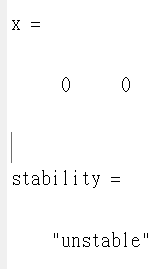
**HW3**

**1.A**

 只有0 0為此題之平衡點，其餘不在坐標系上

syms x1 x2;

x1\_dot = x2 -(4\*x1^2 + x2^2 - 4)\*x1;

x2\_dot = -1\*x1 -(4\*x1^2 + x2^2 - 4)\*x2;

[x1,x2] = solve(x1\_dot,x2\_dot);

x = [double(x1(1)) double(x2(1))]

x\_1 = x(1);

x\_2 = x(2);

%use the first eq. point x = [0 0], and check its stability by linearization

%use Tylor series, take the terns n<2

A = [ -12\*x\_1^2-x\_2^2+4 1-2\*x\_1\*x\_2;

-1-8\*x\_1\*x\_2 -4\*x\_1^2-3\*x\_2^2+4

];

eigen = eig(A)

if vpa(real(eigen),4)<0

stability = "stable";

elseif vpa(real(eigen(1)),4) >0 | vpa(real(eigen(2)),4) >0

plot(x\_1,x\_2,'\*')

hold on

stability = "unstable";

else

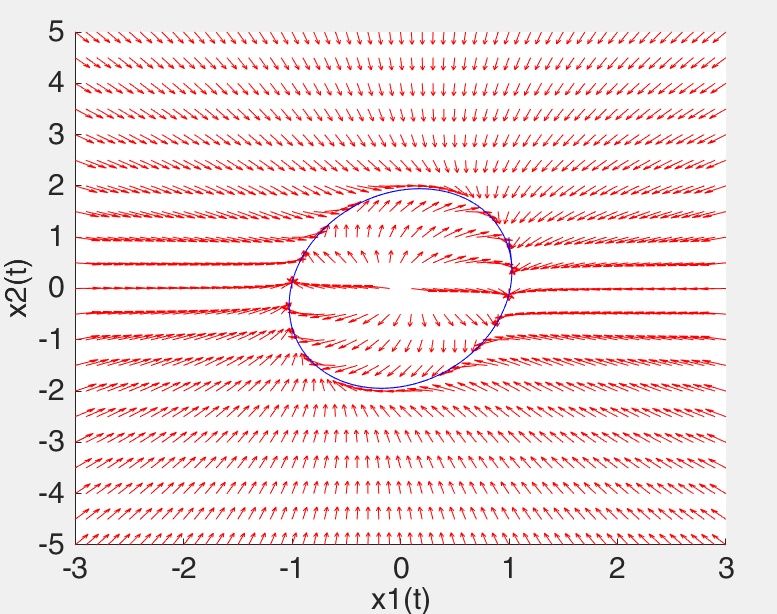
stability = "unknown";

end

x

stability

**1.B**



這個系统求解平衡點只有一个（0,0），不稳定的焦點。

由此圖可以看出|4\*x1^2+x2^2|<4的時候，x的趨勢是向外擴散的，所以此時狀態不穩定；而在|4\*x1^2+x2^2|>4的時候，x的趨勢是向內收斂的，所以此時狀態穩定。而此題不穩定部分被包覆在內，所以皆會傾向於極限環。

1.函式庫

function vectorfield(deqns,xval,yval,t)

if nargin==3;

t=0;

end

m=length(xval);

n=length(yval);

x1=zeros(n,m);

y1=zeros(n,m);

for a=1:m

for b=1:n

pts = feval(deqns,t,[xval(a);yval(b)]);

x1(b,a) = pts(1);

y1(b,a) = pts(2);

end

end

arrow=sqrt(x1.^2+y1.^2);

quiver(xval,yval,x1./arrow,y1./arrow,.5,'r');

axis tight;

2.主程式

clear

hold on

sys=inline('[x(2)-(4\*x(1)^2+x(2)^2-4)\*x(1);-x(1)-(4\*x(1)^2+x(2)^2-4)\*x(2)]','t', 'x');

vectorfield(sys,-3:.1:3,-5:0.5:5);

f=@(t,x)([x(2)-(4\*x(1)^2+x(2)^2-4)\*x(1);-x(1)-(4\*x(1)^2+x(2)^2-4)\*x(2)]);

[t,xs] = ode45(f,[0 30],[1 1]);

plot(xs(:,1),xs(:,2),'b')

hold off

axis([-3 3 -5 5])

fsize=15;

set(gca,'XTick',-3:1:3,'FontSize',fsize)

set(gca,'YTick',-5:1:5,'FontSize',fsize)

xlabel('x1(t)','FontSize',fsize)

ylabel('x2(t)','FontSize',fsize)

hold off

**1.c**

syms x1 x2;

V = 0.5\*(x1^2 + x2^2)

V\_dot = -1\*(4\*x1^2+x2^2-4)\*(x1^2+x2^2)

[x1,x2] = solve(V);

-------------------------------------------------------------------

X1 = 0

X2 = 0

當x1，x2不為0時:

V>0

當4\*x1^2+x2^2-4 <4時:

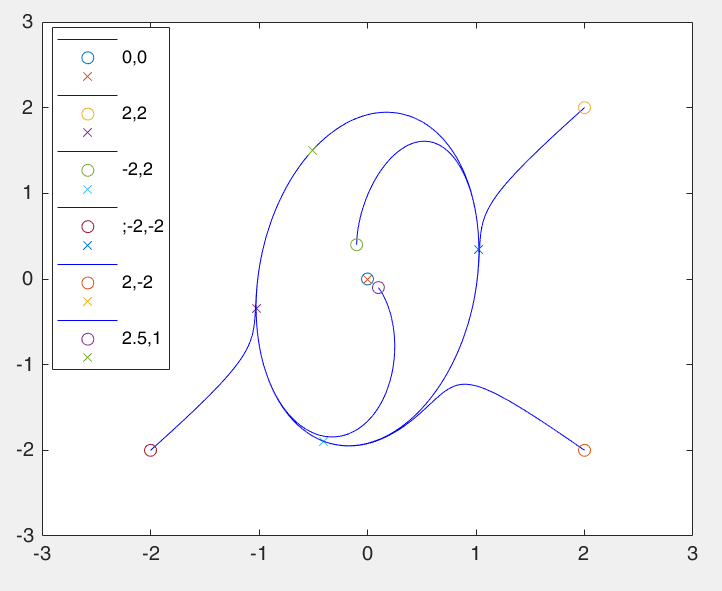
V\_dot>0，系統不穩定

當4\*x1^2+x2^2-4 >4時:

V\_dot<0，系統穩定

所以穩定

**1.d**



%%

1.函式庫

function xNew = myFindNextPos2(A,x,delt)

x\_dot = zeros(size(x))

xNew = x\_dot;

x\_dot = A\*x

xNew = x + x\_dot\*delt;

return;

end

2.主程式

function xNew = myFindNextPos2(A,x,delt)

x\_dot = zeros(size(x))

xNew = x\_dot;

% A = [-2 0;0 -2]; %type 1:star

% A = [-2 -1;2 -6]; %type 2:node

% A = [-2 0;0 1.5];%type 1:saddle point

% A = [0 1;-1 0]; %type 1:center

% A = [-8 50;-50 -8]; %type 1:focus

x\_dot = A\*x

xNew = x + x\_dot\*delt;

return;

end

%% 1-d

r = [0 0;-8 -5;4 -2;-4 1;5 7;0 2]

x1arr1 = [0,0,0,0]';

x1arr2 = x1arr1;

for j=1:6

x=[r(j,1) r(j,2)]'

x1arr1(1) = x(1);

x1arr2(1) = x(2);

delt = 0.0011

x\_1 = x(1)

x\_2 = x(2)

for i=1:2000

A = [ -4\*x\_1^2-x\_2^2+4 1;

-1 -4\*x\_1^2-x\_2^2+4

]

xN1 = myFindNextPos2(A,x,delt);

x1arr1(i+1) = xN1(1);

x1arr2(i+1) = xN1(2);

x = xN1;

x\_1 = xN1(1);

x\_2 = xN1(1);

end

figure(2);

plot(x1arr1,x1arr2,'b');

hold on

plot(x1arr1(1),x1arr2(1),'o',x1arr1(1001),x1arr2(1001),'x')

end

legend('','0,0','','','-8,-5','','','4,-2','','','-4,1','','','5,7','','','0,2','')

**2.i**

syms x1 x2;

V = 0.5\*(x1^2 + x2^2)

V\_dot = -1\*(4\*x1^2+x2^2-4-U)\*(x1^2+x2^2)

-------------------------------------------------------------------

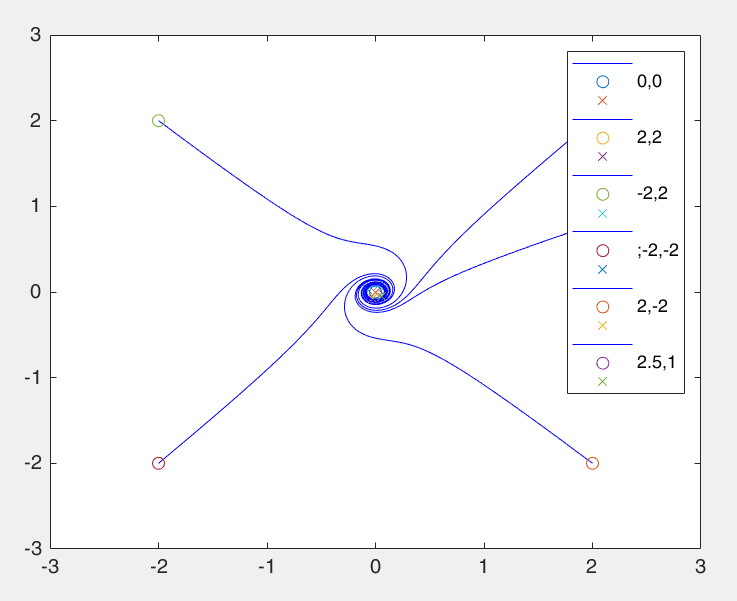
Set U = -4

系統變為V\_dot = -1\*(4\*x1^2+x2^2)\*(x1^2+x2^2)

則V\_dot<=0

系統穩定

**2.ii**



r = [0 0;2 2;-2 2;-2 -2;2 -2;2.5 1]

x1arr1 = [0,0,0,0]';

x1arr2 = x1arr1;

for j=1:6

x=[r(j,1) r(j,2)]'

x1arr1(1) = x(1);

x1arr2(1) = x(2);

delt = 0.005

x\_1 = x(1)

x\_2 = x(2)

for i=1:4000

u = (-4-3\*x\_1^2)

A = [ -4\*x\_1^2-x\_2^2+4+u 1;

-1 -4\*x\_1^2-x\_2^2+4+u

]

xN1 = myFindNextPos2(A,x,delt);

x1arr1(i+1) = xN1(1);

x1arr2(i+1) = xN1(2);

x = xN1;

x\_1 = xN1(1);

x\_2 = xN1(1);

end

figure(2);

max(x1arr2)

plot(x1arr1,x1arr2,'b');

hold on

y(j,:) = [x1arr1(4001),x1arr2(4001)]

plot(x1arr1(1),x1arr2(1),'o',x1arr1(4001),x1arr2(4001),'x')

end

r = [0 0;2 2;-2 2;-2 -2;2 -2;2.5 1]

legend('','0,0','','','2,2','','','-2,2','','',';-2,-2','','','2,-2','','','2.5,1','')

axis([-3,3,-3,3])