

```
clear all
close all

priklad = 114;

% matice soustavy
switch (priklad)
    case 111
        A=[0 1;6 -1]; % sedlo
    case 121
        A=[0 1;-2 3]; % uzel bikriticky
    case 112
        A=[-2 2; -2 -2]; % ohnisko
    case 114
        A=[1 -5;2 -1]; % stred
    case 115
        A=[1 -1;1 3]; % uzel monokriticky
    otherwise
        disp('takovy prikklad neni pripraven');
        return
end

%% vypocet smeroveho pole T %%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% volba hustoty site
h = 0.2;
% volba rozsahu site
xh =-3 : h : 3 ;
yh =-3 : h : 3 ;
% souradnice vseh bodu site
[xx,yy] = meshgrid(xh,yh);
% zmena formatu pro nasobeni matici:
% XX ma dva radky,
% XX(:,k) jsou souradnice k-teho bodu site
XX = [xx(:), yy(:)]';
% vypocet smeroveho pole
T = A*XX;

%% nakreseni pole T %%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% zmena formatu pro funkci quiver
tx = reshape(T(1,:),size(xx));
ty = reshape(T(2,:),size(yy));
% normovani vektoru (tx, ty)
p = sqrt(tx.*tx + ty.*ty);
z = find(p<0.0001); % aby se nedelilo 0 v BR
p(z) = 1;
ttx = 0.8*tx./p; % 0.8 je empiricka konst., vypada to nejlip
tty = 0.8*ty./p;
% v kazdem bode site xh,yh se nakresli vektor [ttx, tty]
quiver(xh, yh, ttx, tty);
hold on

%% nakreseni vybraneho reseni %%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

h=0.05; % krok pro kresleni krivky

switch (priklad)
    case 111
        % v1. vektory
        plot([-1.5 1.5], [-3 3],'k','linewidth',2); % , 'linewidth',2
        plot([-1 1], [3 -3],'k','linewidth',2);
        % reseni C.u.
        t = 0 : h : 1.35;
        X = -0.1*exp(2*t).*[1;2] +0.9*exp(-3*t).*[1;-3];
        Xi=X;
    case 121
        % v1. vektory
        plot([-1.5 1.5], [-3 3],'k','linewidth',2); % , 'linewidth',2
        plot([-3 3], [-3 3],'k','linewidth',2);
        % reseni C.u. - pro trajektorii
        t = -0.1 : h : 2;
        X = 1*exp(t).*[1;1] - 0.1*exp(2*t).*[1;2];
        % reseni C.u. - pro integ. krivku
        t=-3:h:2;
        Xi = 1*exp(t).*[1;1] - 0.1*exp(2*t).*[1;2];
    case 112
        t = 0 : h : 6;
        X = exp(-2*t).*[2*cos(2*t)-3*sin(2*t); -3*cos(2*t)-2*sin(2*t)];
        Xi=X;
    case 114
        t = 0 : h : 6*pi/3;
        X = [-cos(3*t)-2*sin(3*t); cos(3*t)-sin(3*t)];
        Xi=X;
        Xi1= [2*cos(3*t)-sin(3*t); cos(3*t)+sin(3*t)];
        % ezplot('2*x^2-2*x*y+5*y^2-9',[-3, 3]) % kontrola
    case 115
        % v1. vektor
        plot([-3 3], [3 -3],'k','linewidth',2); % , 'linewidth',2
        t = -1.2 : h : 0.3;
        X = exp(2*t).*[3*t-2; -3*t-1];
        Xi=X;
    otherwise
        disp('takovy prikklad neni pripraven');
end

plot(X(1,:),X(2,:), 'r','linewidth',2) % trajktorie reseni
plot(X(1,1),X(2,1), 'r*', 'linewidth',4) % poc. podm.
axis tight
axis equal

figure
plot3(t,Xi(1,:),Xi(2,:), 'r','linewidth',2) % integ. krivka
axis equal
xlabel('t', "fontsize", 16,'fontweight','bold');
ylabel('x', "fontsize", 16,'fontweight','bold');
zlabel('y', "fontsize", 16,'fontweight','bold');
grid on

if priklad==114
    hold on
    plot3(t,Xi1(1,:),Xi1(2,:), 'k','linewidth',2) % dalsi integ. krivka
    h=legend('X(\pi/2)=[2;1]', 'X(0)=[2;1]');
    set (h, "fontsize", 16);
    hold off
end
end
```