Table of Contents

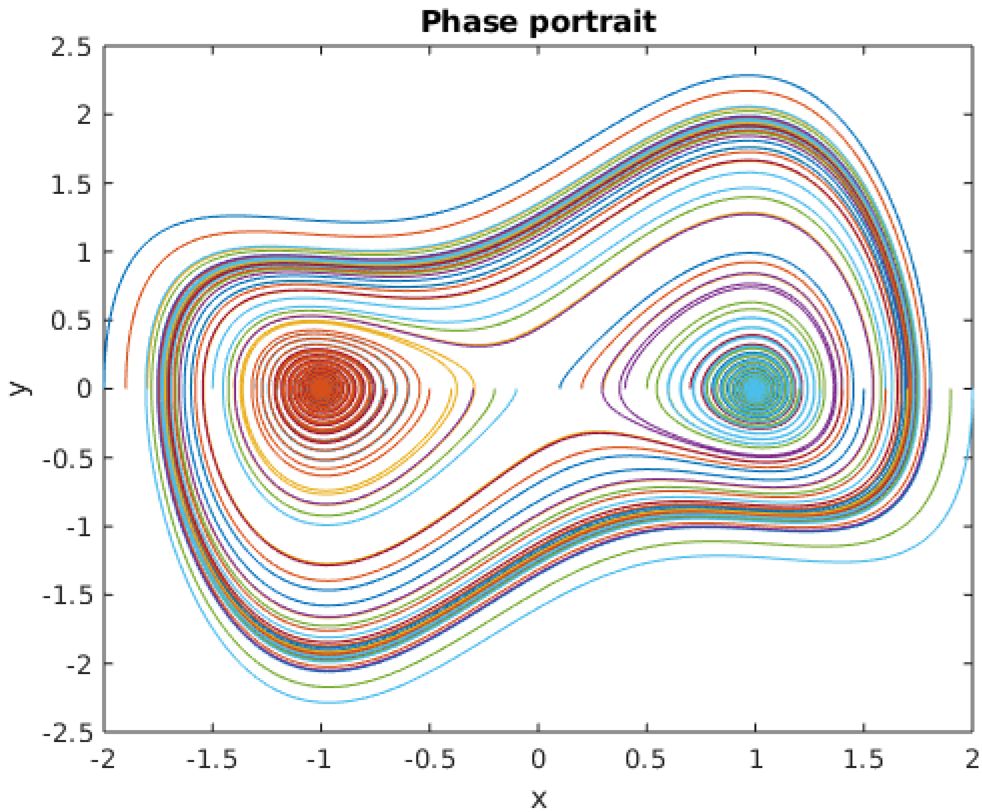
Section A

```
%The analytical results for the equilibrium points are (0,0), (1,0),
 (-1,0)
```

Section B

```
close all;
clc;
format long g;

h = 1e-2;
desiredPoints = 100000;
figure;

% Using the analytical results of section A, we have chosen the
 following
% intial guesses for the time stepper, with which we want to obtain
 the
% trajectories to see the phase portrait of the system.

for guessX = -2:0.1:2%inital points on the x axis, we keep y=0
    guess = [guessX; 0];
    result = RK4(guess, h, @functionODE, desiredPoints);
    plot(result(1, :), result(2, :))
    hold on
end

title('Phase portrait')
xlabel('x')
ylabel('y')

% From the results above it is clear that the system has three orbits.
 It
% seems like the tragectories tend tot he external one if the intial
 point
% is outside the small orbits, and it tends to the focus of the small
% orbits if the intial point is inside. This makes us think that the
% external one will be stable atractor and the other ones will be
 unstable
% repulsor.

% For the x points -1,0,1 there are no results obatined, this
```

```
% result was expected as in section A we discovered that those points
 were
% equilibrium points.

% We will analize the stabilities in section C.
```

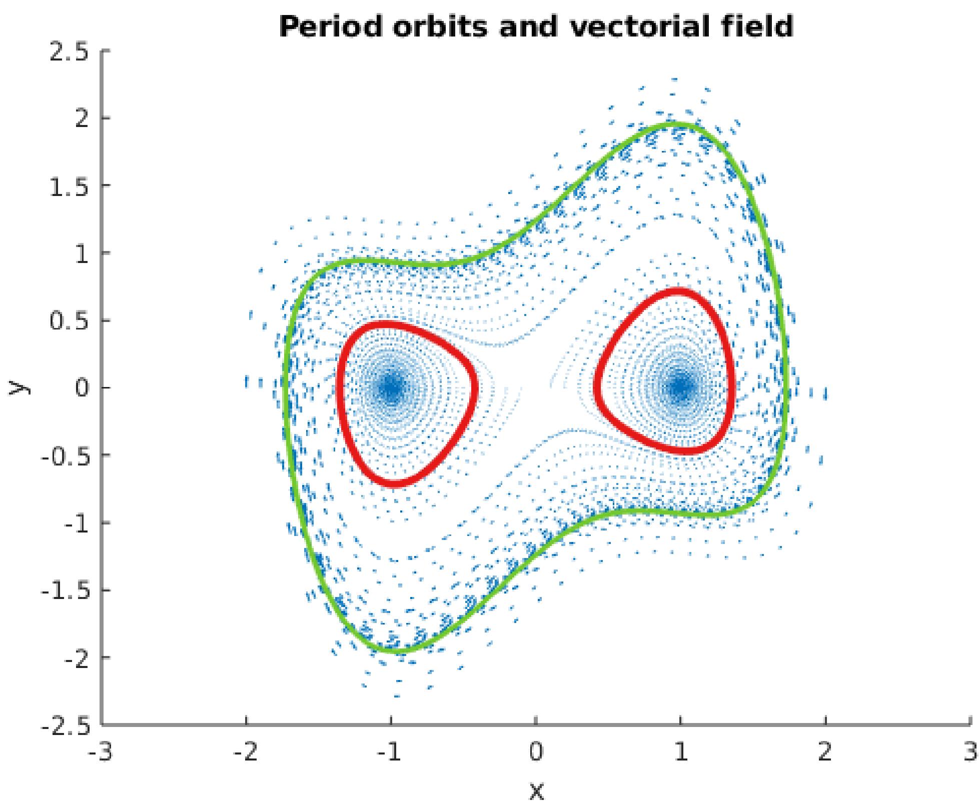


Section C

Now we are looking for the stable and unstable equilibrium orbits mentioned in B.

```
% Following the guesses in section B about the orbit's stabulity, to
 obtain
% the unstables we will integrate backward in time and use intial
 guesses
% close to it that we will get form the plot in section B.
% To get the stable one we will integrate forward in time, and the
 intial
% guess can be any point outside the two unstabel orbits, as all
% trajectories that stat in that region will get to it, however we'll
 use a
% close one.

figure;
desiredPoints = 1000;
inestableGuess1 = [-1.3530; -0.0262];
inestableGuess2 = [1.3530; 0.0262];
```

```
stableGuess = [-1.7100; -0.3825];
h = 0.1;
X = [];
Y = [];

for guessX = -2:0.1:2%inital points on the x axis, we keep y=0
    guess = [guessX; 0];
    result = RK4(guess, h, @functionODE, desiredPoints);
    X = [X; result(1, :)];
    Y = [Y; result(2, :)];
    hold on
end

vx = Y';
vy = X' + 0.9 * Y' - X'.^3 - X'.^2 .* Y';
quiver(X', Y', vx, vy)
hold on

h = 1e-2;
result = RK4(inestableGuess1, h, @functionODEBack, desiredPoints);
plot(result(1, :), result(2, :), 'LineWidth', 3, 'Color', [0.9 0.1
 0.10]);
hold on;

result = RK4(inestableGuess2, h, @functionODEBack, desiredPoints);
plot(result(1, :), result(2, :), 'LineWidth', 3, 'Color', [0.9 0.1
 0.10]);
hold on;

% In this case we integrate forwards to find the stable orbit:
result = RK4(stableGuess, h, @functionODE, desiredPoints);
plot(result(1, :), result(2, :), 'LineWidth', 2, 'Color', [0.4660 0.8
 0.1880]);
hold on;

hold off
title('Period orbits and vectorial field')
xlabel('x')
ylabel('y')

% Using quiver and ploting the periodic orbits it is easy to see the
% stabilties. The equilibrium points (-1,0) and (1,0) are stable and
% atractor as the velocity arrows of trajectories inside the red
 orbits
% point directly to them. So thei 'get away' from the red orbits,
 which
% also do the ones startic outside the orbit, this tells us that those
% orbits are repulsor. Meanwhile for the points outside the red orbits
 all
% the vecolicty arrows point directly to the green curve, so this is
 an
% stable atractor orbit.
```



Section D:

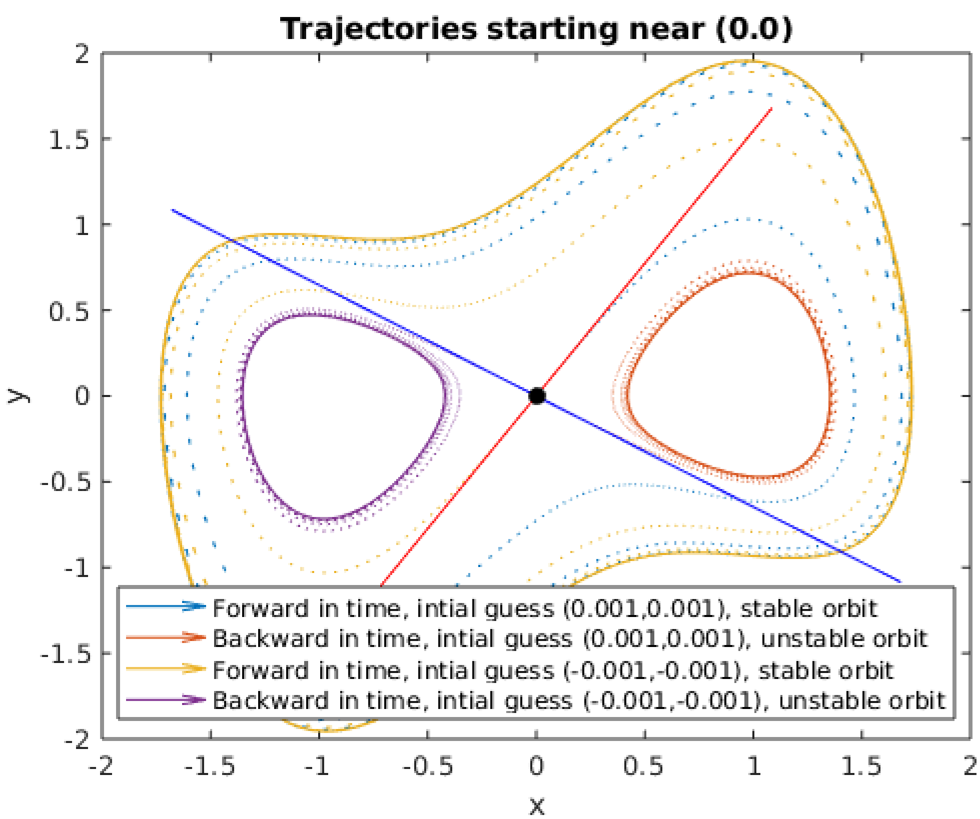
Now let's see which kind of stabulty has the origin. From the plot of section B we know that it is unstable, as nothing goes to it, however is it repulor?

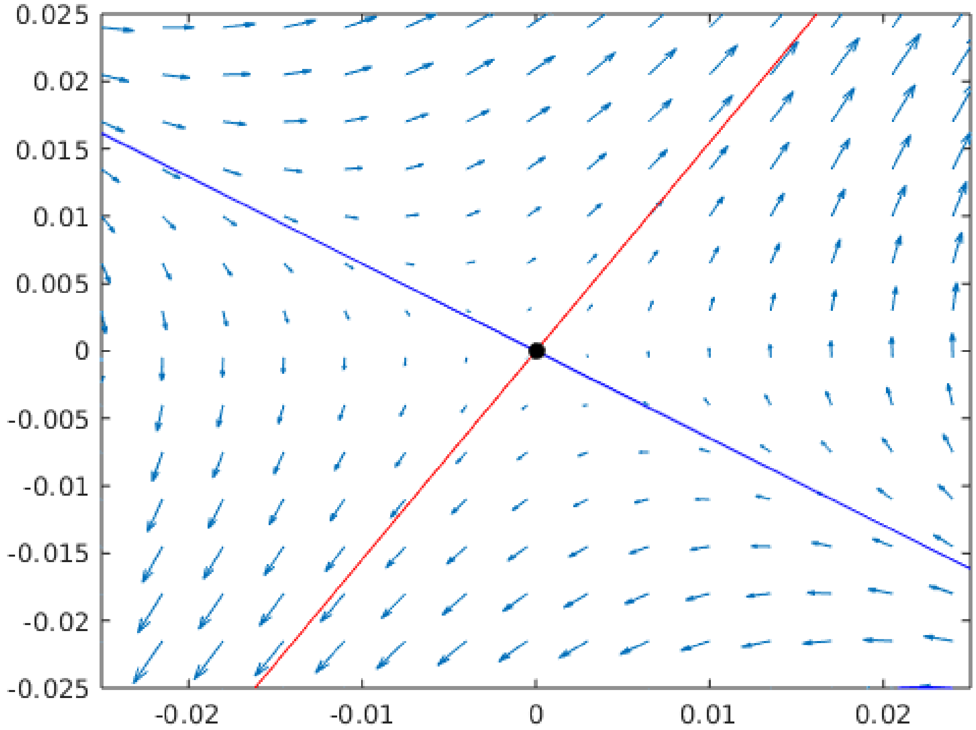
```
% We compute the Jacobian of f at the point 0,0.
desiredPoints = 10000;

DF = jaco(@functionODE, [0; 0]);
[evec, eval] = eig(DF); % evec: matriu amb els vectors propis per
 columna, eval: matriu amb els valors propis a la diagonal
disp('The eigenvalues for the (0,0) point are')
disp(eig(DF))

% As the problem says, we see that the origin is unstable because
 there is
% an eigenvalue with a real positive part.

% Now if we plot the invariant lines corresponding to the eigenvalues
% obtained we see how the tragectories approax or get away of the
 origin.
figure;


guess = [0.001; 0.001];
h = 0.05;
```

```
result = RK4(guess, h, @functionODE, desiredPoints);
quiver(result(1, :), result(2, :), result(2, :), result(1, :) + 0.9 *
 result(2, :) - result(1, :).^3 - result(1, :).^2 .* result(2, :));
hold on;

result = RK4(guess, h, @functionODEBack, desiredPoints);
quiver(result(1, :), result(2, :), result(2, :), result(1, :) + 0.9 *
 result(2, :) - result(1, :).^3 - result(1, :).^2 .* result(2, :));
hold on

% We try on another point near to the origin:%
guess = [-0.001; -0.001];
result = RK4(guess, h, @functionODE, desiredPoints);
quiver(result(1, :), result(2, :), result(2, :), result(1, :) + 0.9 *
 result(2, :) - result(1, :).^3 - result(1, :).^2 .* result(2, :));
hold on;

result = RK4(guess, h, @functionODEBack, desiredPoints);
quiver(result(1, :), result(2, :), result(2, :), result(1, :) + 0.9 *
 result(2, :) - result(1, :).^3 - result(1, :).^2 .* result(2, :));

plot(linspace(-2, 2, 10) * evec(1, 1), linspace(-2, 2, 10) * evec(2,
 1), 'b')
hold on
plot(linspace(-2, 2, 10) * evec(1, 2), linspace(-2, 2, 10) * evec(2,
 2), 'r')
hold on
plot(0, 0, 'o', 'MarkerEdgeColor', 'k', 'MarkerFaceColor', 'k')
hold on
title('Trajectories starting near (0.0)')
xlabel('x')
ylabel('y')
legend('Forward in time, intial guess (0.001,0.001), stable
 orbit', ...
    'Backward in time, intial guess (0.001,0.001), unstable
 orbit', ...
    'Forward in time, intial guess (-0.001,-0.001), stable orbit', ...
    'Backward in time, intial guess (-0.001,-0.001), unstable
 orbit', 'Location', 'best');

%Equal spaced vectors cuadricula

[x, y] = meshgrid(-0.025:0.0035:0.025, -0.025:0.0035:0.025);
vx = y;
vy = x + 0.9 * y - x.^3 - x.^2 .* y;


%As expected when we integrate forward the tragectories tend to the
 stable
%orbit meanwhile integrating backwards we get the unstable ones that
 go to
%the repulsor points.
```

```
% Now focusing on the origin, We also see that it is unstable because
 the
% trajectories do not tend to it.

zoomX = 0.025;
zoomY = 0.025;
figure
quiver(x, y, vx, vy)
startx = -zoomX:0.001:zoomX;
starty = -zoomY.*ones(1,length(startx));
streamline(x, y, vx, vy, startx, starty);
hold on;
plot([-0.5:0.01:0.5] * evec(1, 1), [-0.5:0.01:0.5] * evec(2, 1), 'b')
hold on
plot([-0.5:0.01:0.5] * evec(1, 2), [-0.5:0.01:0.5] * evec(2, 2), 'r')
hold on
plot(0, 0, 'o', 'MarkerEdgeColor', 'k', 'MarkerFaceColor', 'k')
axis([-zoomX, zoomX, -zoomY, zoomY]);
```

The eigenvalues for the (0,0) point are
-0.646585609973065
1.54658560997307





Published with MATLAB® R2019b