Practica_11_Casas_MercadÃ©

Table of Contents

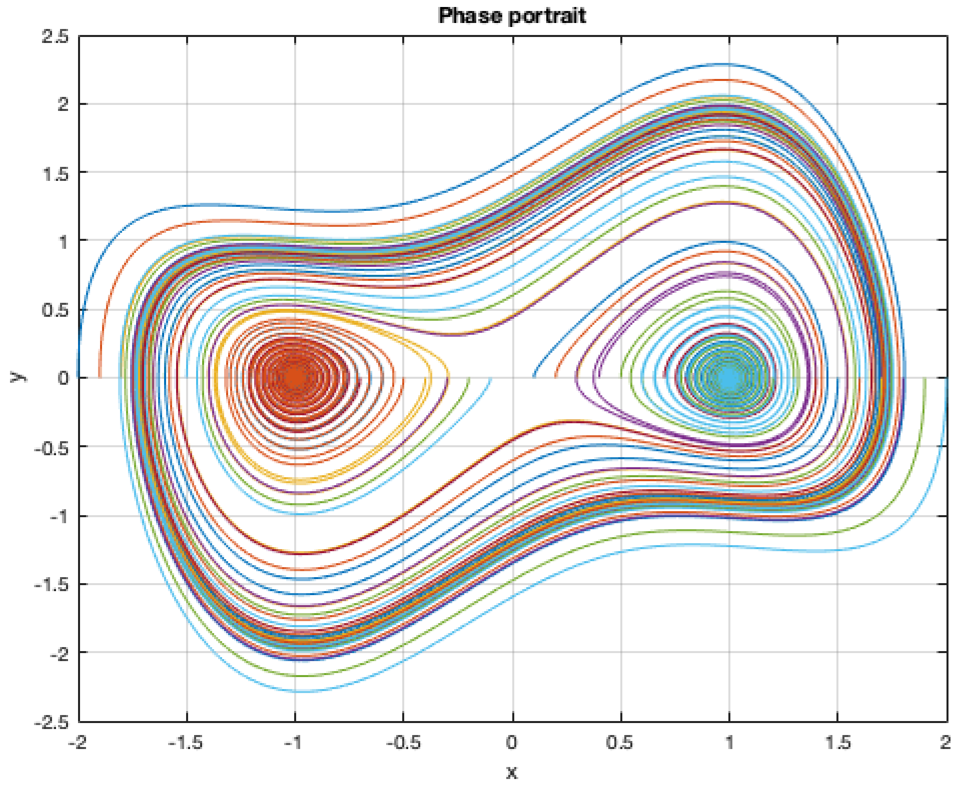
Section A

The analytical results for the equilibrium points are (0,0), (1,0), (-1,0). The analytical derivation will be attached to the end of this pdf.

Section B

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

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


% Using equispaced intial guesses for the time stepper, we 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
    grid on
end

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

% From the results above it is clear that the system has three
 invariant
% orbits. It seems like the trajectories tend to the 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.

% 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.
```



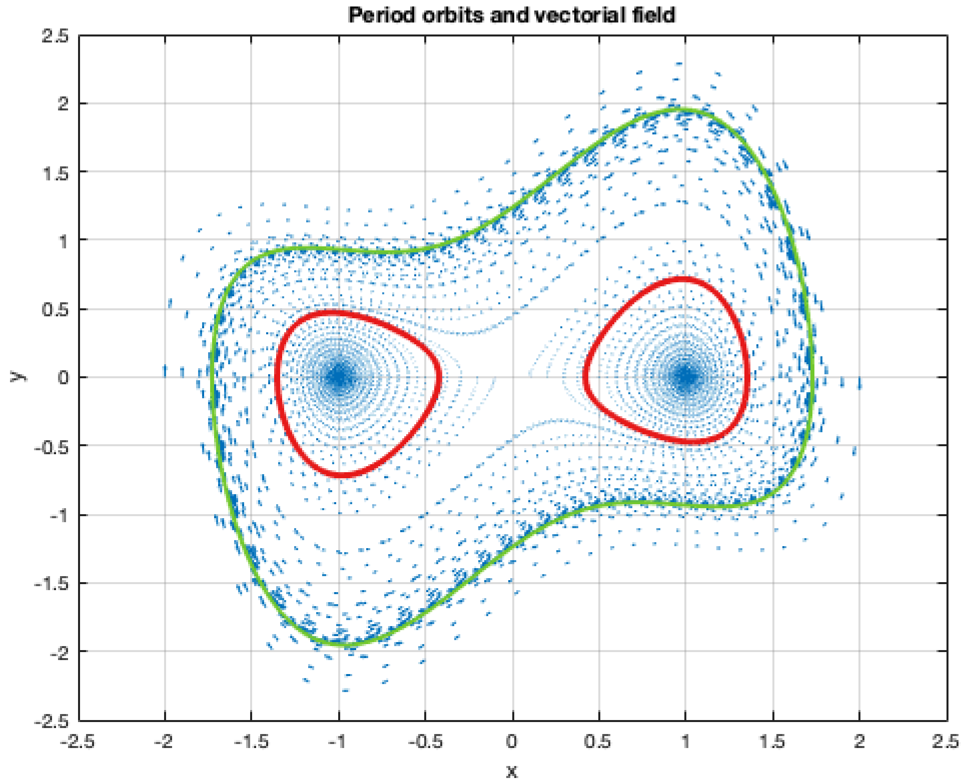
Section C

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

```
% Following the guesses in section B about the orbit's stability, 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, :)];
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;
grid 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
% stabiluties. 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 they '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 velocity arrows point directly to the green curve, so this is an
```

```
% stable atractor orbit.
```



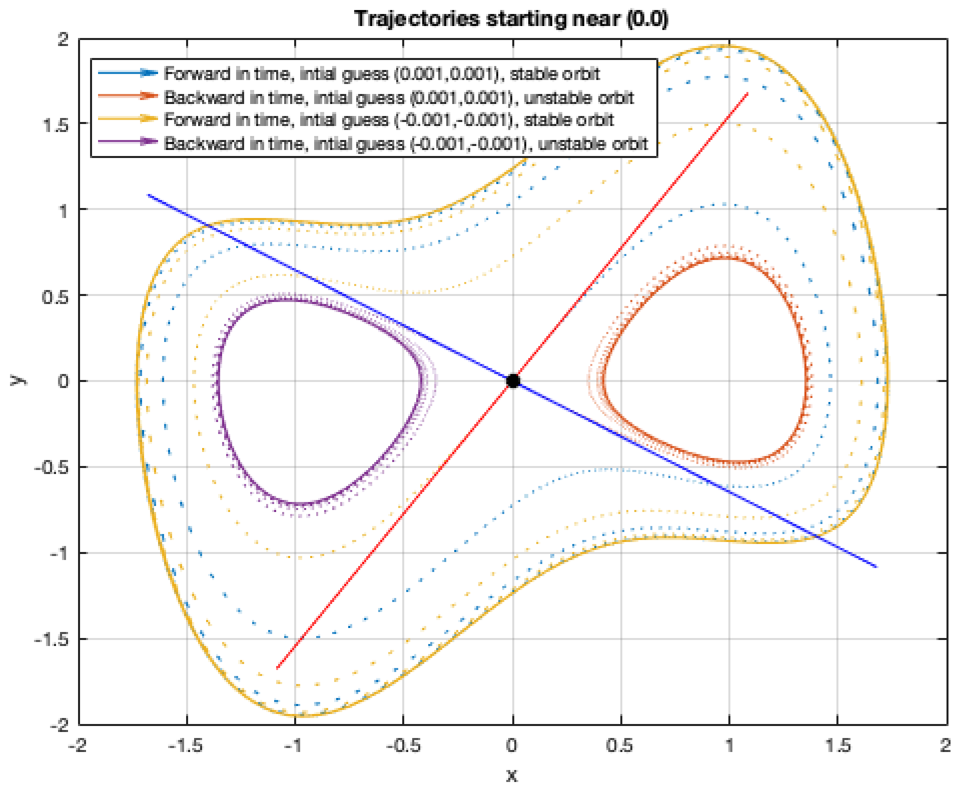
Section D

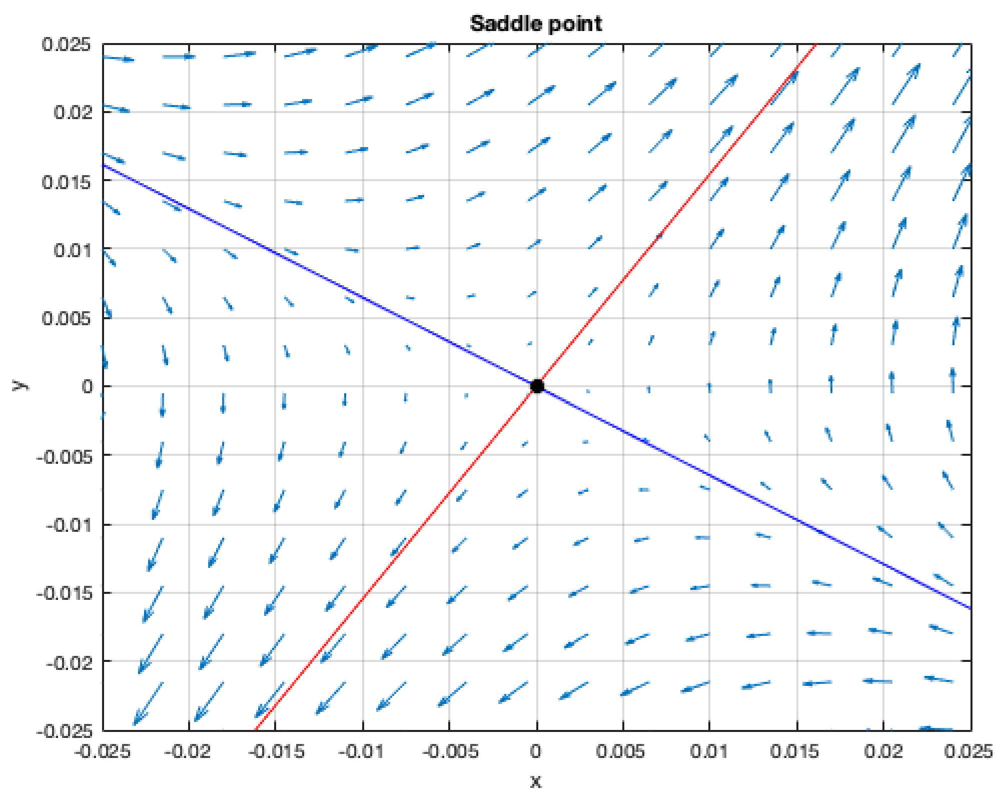
```
% Now let's see the stabilty of the origin.
%From the plot of
% section B we know that it is unstable, as nothing goes to it,
 however is
% it repulor? More specifically it is a saddle point because it has a
% positve and a negative real part eigenvalue. As a saddle point, it
 has
% an invariant inestable vector (red) and a inverant stable vector
 (blue)
% passing through it.

% 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 has a real positive
```

```
% eigenvalue and anotherone negative, so the origin is a saddle point.
 Now
% if we plot the two invariant lines corresponding to the two
 eigenvalues
% we se how the system behaves near the point. We plot in blue the
% atractive one corresponding to the neagtive eigenvalue, and in red
 the
% repulsor one which is due to the positive eigenvalue.


% We start RK4 near the origin to see how the trajevctories evolve,
 and we
% use both the forward and backawrd integration.

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, :));

% Now we plot the invariant lines to have a more visual concept of how
 the
% system behaves due tot the saddle point in the origin
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
grid 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');


% As expected when we integrate forward the trajectories 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 see how the trajectories approax the
% origin coming from the atractor blue line but when are close to the
% origin they take the repulsor line direcction. In order to see this
% saddle behaviour better we do a 'zoom' centered in the origin and
 plot
% not only the invariant lines but also a vetor field which shows us
 how
% the particles will move near (0.0)

% Equispaced 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;

zoomX = 0.025;
zoomY = 0.025;
figure;
quiver(x, y, vx, vy)
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')
grid on
title('Saddle point')
xlabel('x')
ylabel('y')
axis([-zoomX, zoomX, -zoomY, zoomY]);
```

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


Trajectories starting near (0.0)
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
x
y


Saddle point
x
y

Published with MATLAB® R2018b