Исследование систем первого порядка: поиск особых точек систем и определения их типа

Исследование систем будем проводить с помощью программы написанной в Matlab. Код программы и функции для на нахождения особых точек находятся в приложении.

Решение по каждой системе представлено на отдельной странице и содержит уравнение системы, фазовый портрет, количество и описание состояний равновесия.

**Система 1**

Уравнение системы 1:



В результате использования программы получаем следующий фазовый портрет системы (рис. 1):

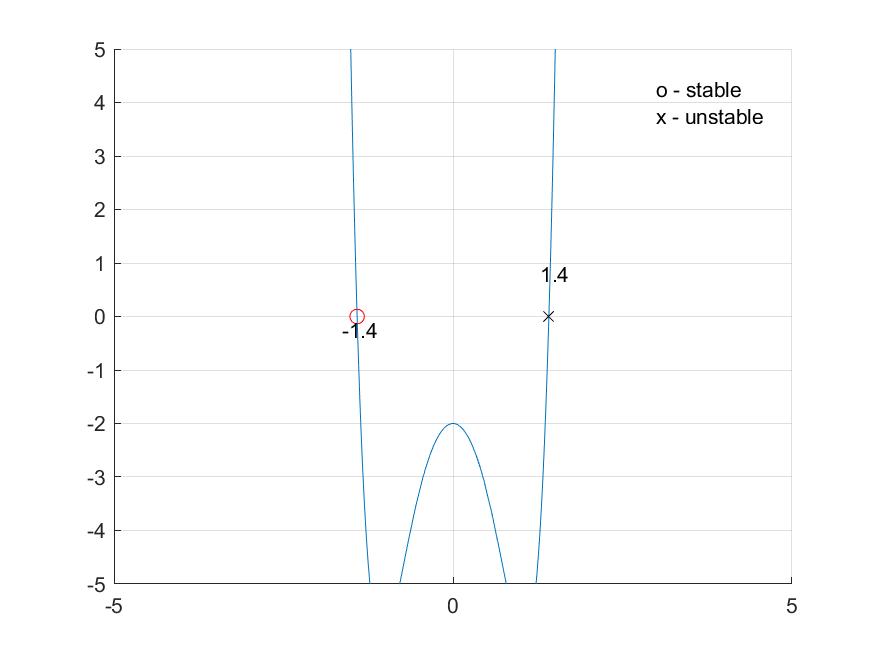


Рисунок - Фазовый портрет системы 1

**Вывод:**

Динамическая система первого порядка имеет два состояния равновесия:

x = 1.4142 – неустойчивое

x = -1.4142 – устойчивое

Гарантированные интервалы устойчивости: x < 1.41

Гарантированные интервалы неустойчивости: x > 1.41

**Система 2**

Уравнение системы 2:



В результате использования программы получаем следующий фазовый портрет системы (рис. 2):

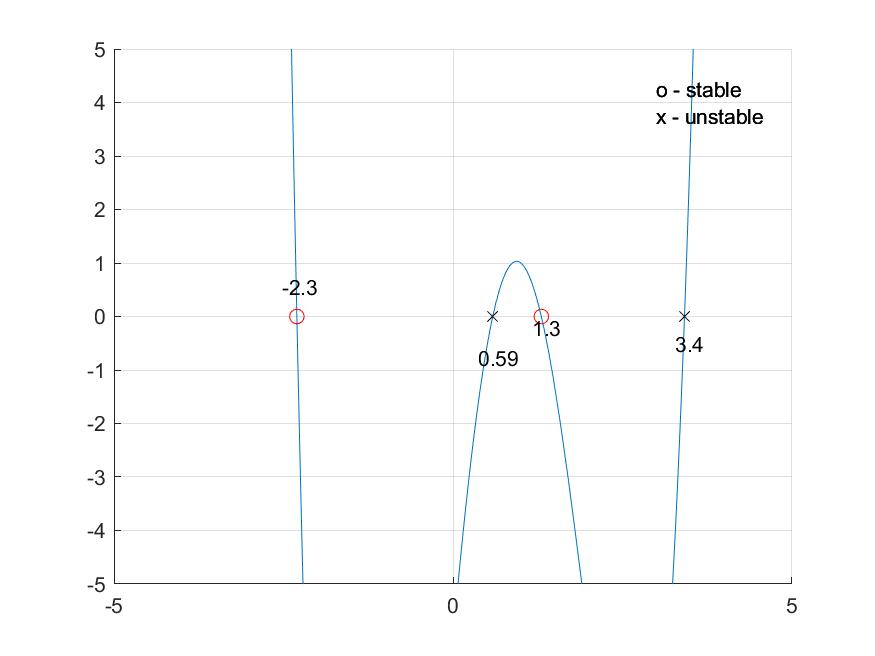


Рисунок - Фазовый портрет системы 1

**Вывод:**

Динамическая система первого порядка имеет четыре состояния равновесия:

x = 0.5858 – неустойчивое

x = -2.3028 – устойчивое

x = 1.3028 – устойчивое

x = 3.4142 – неустойчивое

Гарантированные интервалы устойчивости: x < 0.59 и 3.4 > x > 0.59

Гарантированные интервалы неустойчивости: x > 3.4

**Система 3**

Уравнение системы 3:

****

В результате использования программы получаем следующий фазовый портрет системы (рис. 3):

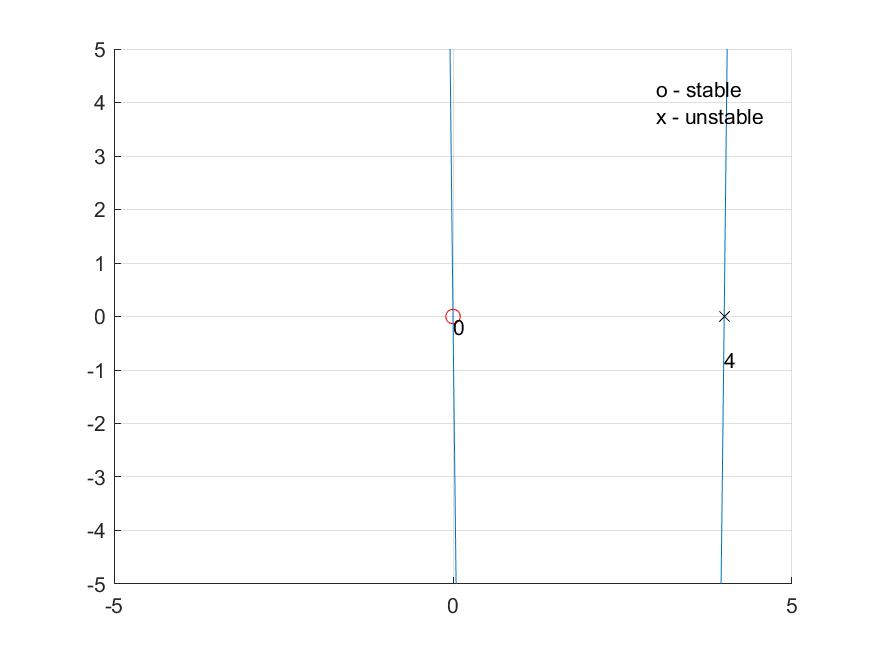


Рисунок 3 - Фазовый портрет системы 1

**Вывод:**

Динамическая система первого порядка имеет два состояния равновесия:

x = 0 – устойчивое

x = 4 – неустойчивое

Гарантированные интервалы устойчивости: x < 4

Гарантированные интервалы неустойчивости: x > 4

**Система 4**

Уравнение системы 4:



В результате использования программы получаем следующий фазовый портрет системы (рис. 4):

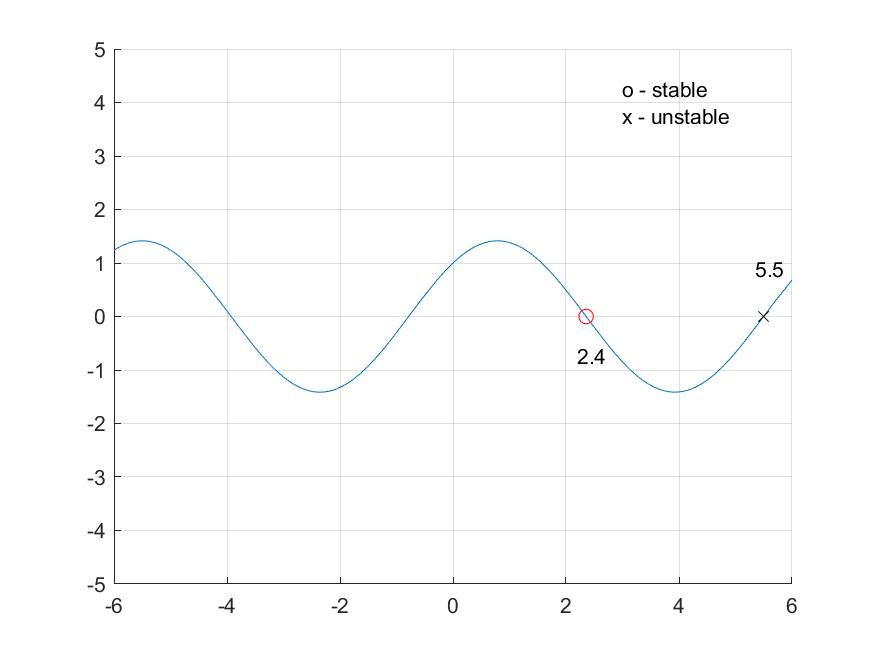


Рисунок 4 - Фазовый портрет системы 1

**Вывод:**

Динамическая система первого порядка является колебательной и имеет следующие равновесные состояния:

|  |  |
| --- | --- |
| Периодические | На одном периоде |
|  | x = 2.3562 – устойчивое  x = 5.4978 – неустойчивое |

Гарантированных интервалов устойчивости/неустойчивости нет.

**Система 5**

Уравнение системы 5:



В результате использования программы получаем следующий фазовый портрет системы (рис. 5):

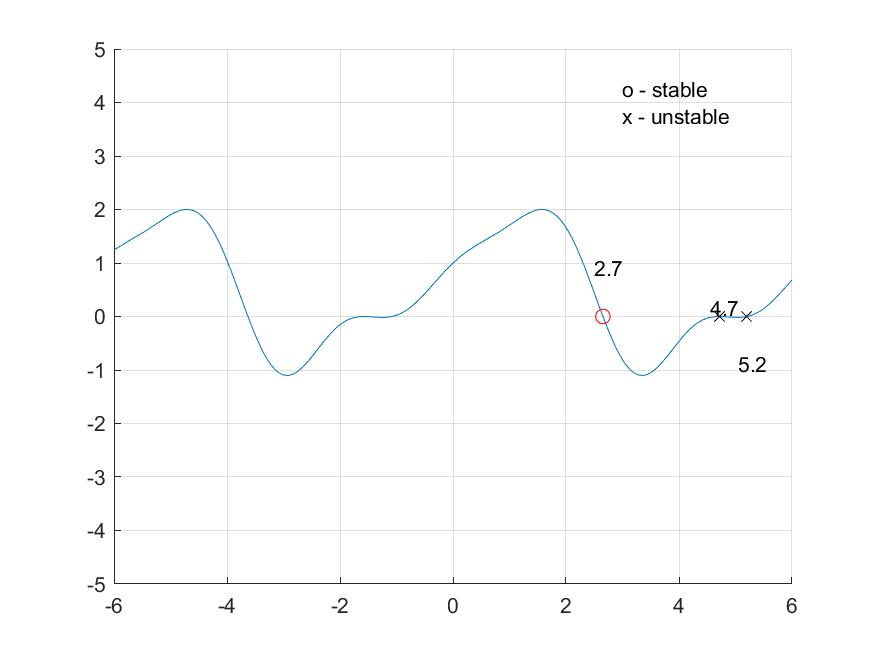


Рисунок 5 - Фазовый портрет системы 1

**Вывод:**

Динамическая система первого порядка является колебательной и имеет следующие равновесные состояния:

|  |  |
| --- | --- |
| Периодические | На одном периоде |
|  | x = 2.6534 – устойчивое  x = 4.7124 – неустойчивое  x = 5.2005 – неустойчивое |

Гарантированных интервалов устойчивости/неустойчивости нет.

**Приложение 1**

Код программы на Matlab

clc, close all

syms x

f1 = 2\*x^6 - x^4 - 5\*x^2 - 2;

f2 = (x^2 - 3\*x + 1)^2 + 3\*(x - 1)\*(x^2 - 3\*x + 1) - 4\*(x - 1)^2;

f3 = (x - 1)^4 + (x - 3)^4 - 82;

f4 = sin(x) + cos(x);

f5 = sin(x) + (sin(x))^2 + (cos(x))^3;

f = [f1 f2 f3 f4 f5];

legend = 'o - stable';

legend = [legend newline 'x - unstable'];

for i = 1:5

disp(f(i))

[points, isStable] = special\_points(f(i), x)

figure(i)

hold on

fplot(f(i))

ylim([-5 5])

for j = 1:size(points,1)

if isStable(j) == 'stable'

plot(points(j),0,'ro','MarkerSize', 7)

text(3,4,legend)

else

plot(points(j),0,'kx','MarkerSize', 7)

end

text(points(j),-1 + 2.\*rand,sprintf(' % -.2g', points(j)),HorizontalAlignment='center')

end

grid on

name = sprintf('portrait%d .jpg', i);

saveas(i, name);

hold off

end

**Приложение 2**

Код функции поиска особых точек на Matlab

function [points, stable] = special\_points(func, arg)

point\_struct = solve(func, arg, 'Real',true, 'ReturnConditions',true); % roots structure

points = point\_struct.x;

if size(point\_struct.parameters, 2) > 0 % check if func is periodic

period\_counter = 0; % periodic roots counter

j = 0; % integer iterator for periodic roots

points = [];

while period\_counter < size(point\_struct.x,1)

for i = 1:size(point\_struct.x,1)

if subs(point\_struct.x(i),point\_struct.parameters,j) < 2\*pi && subs(point\_struct.x(i),point\_struct.parameters,j) >= 0 % check if the root within the peroid

points = [points; subs(point\_struct.x(i),point\_struct.parameters,j)]; % append new roots

else

if subs(point\_struct.x(i),point\_struct.parameters,j) >= 0

period\_counter = period\_counter + 1;

end

end

end

j = j + 1;

end

end

points = eval(points);

df = diff(func);

stable = subs(df, arg, points);

for i = 1:size(stable,1)

if eval(stable(i)) < 0

stable(i) = 'stable';

else

stable(i) = 'unstable';

end

end

end