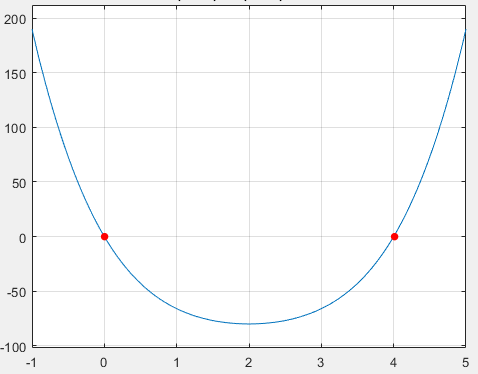
**Лабораторная работа №1**

Определение особых точек системы первого порядка

**Цель работы:** нахождение особых точек системы первого порядка и определение их типа с помощью Matlab.

**Основные сведения:** под особыми точками системы подразумеваются состояния равновесия. Состояние равновесия, это такое состояние системы, где все производные переменных состояния равны нулю. Таким образом, если система задана уравнением , то для того, чтобы найти все особые точки системы необходимо найти решения **Важное замечание:** так как переменные состояния не могут принимать комплексные значения, то под состояниями равновесия принимаются исключительно вещественные решения данного уравнения.

Для определения устойчивости системы в точке равновесия необходимо оценить поведение системы в окрестностях этой точки. Очевидно, что при положительном значении производной, значение переменной состояния будет возрастать. И наоборот, при отрицательном значении производной, значение переменной состояния будет убывать. Тогда, если рассмотреть пример на рисунке 1, можно увидеть, что у системы две точки равновесия. Причем левая является устойчивой, так как из всех окрестностей этой точки система стремится в положение равновесия, а правая – неустойчивой, так как система всегда будет расходиться от этой точки.



**+**

**+**

**-**

*Рисунок 1. – состояния равновесия системы*

Для того чтобы реализовать алгоритм автоматического определения типа особых точек системы, необходимо формализовать рассмотренное условие. Исходя из рисунка, можно увидеть, что состояние равновесия является устойчивым, если функция *f(t)* совершает переход через ось абсцисс сверху вниз в рассматриваемой точки и не устойчивым, если переход происходит снизу вверх. Таким образом, для однозначного определения типа особой точки необходимо определить знак угла наклона функции. Для этого можно взять производную от *f(x)*. То есть необходимо определить знак , где *xi* – все состояния равновесия системы. В случае если производная принимает отрицательное значение, состояние равновесия является устойчивым, если производная равна нулю или положительна, состояние равновесия является неустойчивым.

**Реализация алгоритма в Matlab:** так как необходимо выполнять аналитические операции над функцией, проще всего воспользоваться записью функции через символьные переменные. Для этого необходимо объявить символьную переменную с помощью типа syms. Тогда все выражения, включающие такую переменную, также будут символьные. Например:

>> syms x

>> fun = x^2 + 5

fun =

x^2 + 5

Для самопроверки можно построить график с помощью функции *fplot(fun, [X\_min X\_max]),* где вторым аргументом являются пределы отображения функции на графике.

Для решения функции можно воспользоваться функцией *solve(fun).* Данная функция возвращает все решения в символьном виде. Если необходимо вывести значение в консоль, рекомендуется применить к выводу операцию *eval*, которая приведет выражение из вида, в котором оно хранится в памяти Matlab к более простому и удобному виду. Например:

|  |  |
| --- | --- |
| >> solve(fun)  ans =  -5^(1/2)\*1i  5^(1/2)\*1i | >> eval(solve(fun))  ans =  0.0000 - 2.2361i  0.0000 + 2.2361i |

Для нахождения производной от функции можно воспользоваться функцией *diff(fun, arg),*  где второй аргумент – переменная по которой производится дифференцирование (в случае функции одной переменной можно опустить).

Для того чтобы подставить в символьное выражение какое либо другое значение можно воспользоваться функцией *subs(fun, x, y)*, где *x* – переменная которую надо заменить, *y* – переменная которую надо подставить. Например:

>> subs(fun, x, y)

ans =

y^2 + 5

Подставлять можно не только переменные, но и числа, в этом случае результат сразу вычислится настолько, насколько это возможно. Также существует достаточно любопытное свойство данной функции: если единственным аргументом указать только функцию, то выполнится подстановка всех переменных на самих себя. Однако, если с момента объявления выражения переменная была переинициализирована, подставится новое значение. Например:

>> syms x

>> fun = x^2 + 5

fun =

x^2 + 5

>> x = 2

x =

2

>> subs(fun)

ans =

9

У большинства функций есть необязательные аргументы, которые могут упростить решение различных задач. В данном руководстве намеренно не описывается весь функционал, чтобы поощрять чтение документации, так что не стесняйтесь использовать функции *help* и *doc.*

**Задание на лабораторную работу:** необходимо реализовать алгоритм поиска особых точек системы и определения их типа. Программу необходимо оформить как функцию, которая на вход принимает функцию в символьном виде, а возвращает все найденные особые точки с указанием их типа. Формат вывода необходимо придумать самостоятельно.