

ЛАБ. РАБОТА №3. ПОСТРОЕНИЕ ГРАФИКОВ

PLOT - построение графиков, заданных матрицами.

PLOT. Синтаксис

PLOT (X, Y) отображает вектор Y как функцию от аргумента вектора X.
Если X или Y - матрица, то вектор строится по строкам или столбцам матрицы, в зависимости от того, какая линия.
Если X - скаляр, а Y - вектор, длина (Y) отложенные точки нанесены на график.

PLOT (Y) отображает столбцы Y по по аргументу их индекса.
Если Y комплексно, PLOT (Y) эквивалентно PLOT (real (Y), imag (Y)). Во всех других применениях PLOT мнимая часть игнорируется.

Различные типы линий, графические символы и цвета могут быть получены с помощью

PLOT (X, Y, S), где S - символьная строка («LineSpec»), сформированная из символов:

ColorOrder - Первый символ строки S

Символ	Смысл	Расшифровка
b	blue	синий
g	green	зеленый
r	red	красный
c	cyan	Голубой (электрик)
m	magenta	пурпурный
y	yellow	желтый
k	black	черный
w	white	белый

Второй символ строки S

Символ	Смысл	Расшифровка
.	point	Точка
o	circle	Круг
x	x-mark	Значок X
+	plus	Значок +
*	star	Звездочка
s	square	Квадрат
d	diamond	Ромбик
v	triangle (down)	Треугольник (вниз)
^	triangle (up)	Треугольник (вверх)
<	triangle (left)	Треугольник (слева)
>	triangle (right)	Треугольник (справа)
p	pentagram	Пентаграмма
h	hexagram	Гексаграмма

LineStyleOrder - Остальные символ/символы строки S

Символы	Смысл	Расшифровка
-	solid	Непрерывная линия
:	dotted	точками
-.	dashdot	Пунктир-точка
--	dashed	Пунктир

Например:

PLOT (X, Y, 'c +:') отображает голубую пунктирную линию с плюсом в виде маркера точек.

в каждой точке данных; PLOT (X, Y, 'bd') отображает синий алмаз по каждому данным
но не рисует ни одной линии.

PLOT (X1, Y1, S1, X2, Y2, S2, X3, Y3, S3, ...) объединяет графики, определенные (X, Y, S) троек, где X и Y являются векторами или матрицами и S – это строки.

Например, PLOT (X, Y, 'y -', X, Y, 'go') разбивает данные дважды, сплошная желтая линия, интерполирующая зеленые круги в точках данных.

Команда PLOT, если цвет не указан, автоматически использует цвета, указанные свойством ColorOrder. По умолчанию ColorOrder указан в таблице выше для цветных систем, где по умолчанию синий для одной строки, а для нескольких строк – для цикла через первые шесть цветов в таблице. Для монохромных систем, PLOT циклически перемещает по осям свойство LineStyleOrder.

PLOT возвращает вектор столбца дескрипторов для объектов LINE, один для каждой строки. У пар X, Y или X, Y, S трижды может следовать пары параметров / значений для указания дополнительных свойств линий.

Смотрите также функции:

SEMILOGX, SEMILOGY, LOGLOG, PLOTYY, GRID, CLF, CLC, TITLE, XLABEL, YLABEL, AXIS, AXES, HOLD, COLORDEF, LEGEND, SUBPLOT, STEM.

PLOT. ПРИМЕРЫ ДЛЯ ПРОВЕРКИ

```
% №1
% Очистка Command Window и прорисовка
% строки приглашения для ввода команд
help clc
clc
```

```
% №2
% Очистка (WorkSpace)
help clear
```

```
% №3
% Рисование графика для табличной функции
help plot
```

```
% Построим первый график
x=-32:0.01:32;
y=sin(x)./x;
plot(x,y);
```

```
% №4
% Специальные форматы plot отображения графиков, когда в качестве X
% используется матрица, либо матрицы используются в качестве X и Y
X1 = 1:0.2:2
X2 = 3:0.2:4
X3 = 6:0.2:7
Z4 =[X1',X2',X3']
plot (1,X1,'d')      % Аргумент – скаляр, Функция – вектор
plot (X1)            % Прорисовка вектора
plot (Z4)            % Прорисовка матрицы
plot (Z4, Z4)        % Аргумент – матрица, Функция – матрица
```

```
% №5
% Заморозка окна для рисования в нем многих графиков (hold on)
help hold
```

```
% Включение grid on / выключение сетки grid off
help grid
```

```
% Команда закрыть окно графика
help close
```

```
% №6.1
% Построим графики в замороженном (общем) окне.
% Порядок построения с возрастающим диапазоном по X
clear all;
close all;
hold on;           % Заморозить окно
x=-2*pi:0.2:2*pi;  % Диапазон 4*pi
y=sin(x)./x;
plot(x,y, '-*r');
input('Для продолжения нажмите Enter ...')
x=-3*pi:0.4:3*pi;  % Диапазон 6*pi
y=2*sin(x);
plot(x,y, '-db');
grid on
hold off
% ВЫВОД: Общее окно автоматически масштабируется функцией
% с наибольшим диапазоном по X и Y
```

```
% №6.2
% Построим графики в замороженном (общем) окне.
% Порядок построения с убывающим диапазоном по X
clear all;
close all;
hold on;           % Заморозить окно
x=-3*pi:0.4:3*pi;  % Диапазон 6*pi
y=sin(x)./x;
plot(x,y, '-*r');
input('Для продолжения нажмите Enter ...')
x=-2*pi:0.2:2*pi;  % Диапазон 4*pi
y=2*sin(x);
plot(x,y, '-db');
grid on
hold off
% ВЫВОД: Общее окно автоматически масштабируется функцией
% с наибольшим диапазоном по X и Y
```

FPLOT - построение графиков, заданных символьной строкой.

FPLOT. Синтаксис

FPLOT (FUN, LIMS) отображает функцию FUN, заданную литералом в символьном виде, в границах оси X, указанных LIMS = [XMIN XMAX]. Использование LIMS = [XMIN XMAX YMIN YMAX] позволяет также контролировать границы на оси Y.

При работе FPLOT функция FUN (X) возвращает вектор строки для каждого элемента вектора X.

Например, если FUN возвращает несколько значений для одного аргумента [f1 (x), f2 (x), f3 (x)], то при вводе вектора [x1; x2] функция должна возвращать матрицу:

```
f1 (x1) f2 (x1) f3 (x1)
f1 (x2) f2 (x2) f3 (x2)
```

FPLOT (FUN, LIMS, TOL) с TOL <1 указывает относительную ошибку (толерантность). По умолчанию TOL равен 2e-3, то есть 0,2 процента точности.

FPLOT (FUN, LIMS, N) с N >= 1 отображает функцию с минимумом N + 1 точек. Значение по умолчанию N равно 1. Максимальный размер шага ограничен (1 / N) * (XMAX-XMIN).

FPLOT (FUN, LIMS, «LineSpec») с заданной спецификацией линии (см. . PLOT)

FPLOT (FUN, LIMS, ...) принимает комбинации необязательных аргументов TOL, N и 'LineSpec', в любом порядке.

[X, Y] = FPLOT (FUN, LIMS, ...) возвращает X и Y такие, что Y = FUN (X), при этом на экран график **не** выводится.

FPLOT (FUN, LIMS, TOL, N, 'LineSpec', P1, P2, ...) использовать параметры P1, P2 и т. д. Параметры непосредственно передаются функции FUN: Y = FUN (X, P1, P2, ...). Чтобы использовать значения по умолчанию для TOL, N или «LineSpec», вы можете заменить их пустыми матрицами ([]).

FPLOT. ПРИМЕРЫ ДЛЯ ПРОВЕРКИ:

% №7

% Рисование графика для аналитической функции
help fplot

% №8.1

% Построим графики в замороженном (общем) окне.

% Порядок построения с **возрастающим** диапазоном по X

clear all;

close all;

hold on; % Заморозить окно

fplot('sin(x)./x',[-2*pi 2*pi], '-r'); % Диапазон 4*pi

input('Для продолжения нажмите Enter ...')

fplot('2*sin(x)',[-3*pi 3*pi]); % Диапазон 6*pi

grid on

hold off

% ВЫВОД: Общее окно автоматически масштабируется функцией

% с наибольшим диапазоном по X и Y

% №8.2

% Построим графики в замороженном (общем) окне.

% Порядок построения с **убывающим** диапазоном по X

clear all;

close all;

hold on; % Заморозить окно

fplot('2*sin(x)',[-3*pi 3*pi]); % Диапазон 6*pi

input('Для продолжения нажмите Enter ...')

fplot('sin(x)./x',[-2*pi 2*pi], '-r'); % Диапазон 4*pi

grid on

hold off

% ВЫВОД: Общее окно автоматически масштабируется функцией

% с наибольшим диапазоном по Y и **последним диапазоном по X**

```
% №8.3
% Проверим предшествующий вывод на примере функций
% с чередующимися диапазонами по X
close all;
hold on;
grid on;
fplot('[log(x)]',[1/4 8*pi], '-g');
input('Для продолжения нажмите Enter ...')
fplot('[exp(x)]',[-2*pi pi/4], '-c');
input('Для продолжения нажмите Enter ...')
fplot('[0.5*sin(x)^2]',[-2*pi 2*pi]);
input('Для продолжения нажмите Enter ...')
fplot('[2*sin(x)/x]',[-3*pi 3*pi], '-r');
hold off
% ВЫВОД: Общее окно автоматически масштабируется функцией
% с наибольшим диапазоном по Y и последним диапазоном по X
```

```
% №8.4
% Использование функции fplot для создания векторов аргумента и
% функции
clear all;
close all;
[X,Y]=fplot('[2*sin(x)]',[-3*pi 3*pi]); % Диапазон 6*pi
input('Для продолжения нажмите Enter ...')
plot(X,Y); % Диапазон 6*pi
grid on
% ВЫВОД: При использовании функции fplot для создания векторов
% аргумента и функции вывод графика отсутствует.
```

```
% №8.5
% Использование функции с различными значениями TOL и N
clear all;
close all;
hold on;
% TOL = 1e-1
fplot('[2*sin(x)]',[-3*pi 3*pi], 1e-1, 'rd-'); % Диапазон 6*pi
% TOL = 1e-1, N = 40
fplot('[2*sin(x)]',[-3*pi 3*pi], 1e-1, 40, 'go-'); % Диапазон 6*pi
hold off;
grid on;
```

```
% №9.1
% Использование литералов при рисовании графиков
% аналитических функций
clear all;
close all;
hold on;
s1='[log(0.5*(1+sin(x)^2))]';
fplot(s1,[-2*pi 2*pi], '-r');
s1='[2*abs(sin(x)/x)]';
fplot(s1,[-4*pi 4*pi]);
grid on;
hold off
```

```
% №9.2
% Использование параметров при рисовании графиков
% аналитических функций
```

```

clear all;
close all;
% TOL = [], N = [], S= [] , Параметр P1
P1=10;
fplot('[(sin(x*P1)+1)]',[ -3*pi 3*pi], [],[],[], P1);
grid on;

```

ДОПОЛНИТЕЛЬНЫЕ ФУНКЦИИ (*axis, ezplot, ezpolar, polar*)

```

% №10
% Создать пустое окно с заданного математического размера
help axis
close all;
axis([-4 4 -2 2]);

```

```

% №11.1
% Управление математическим размером
% окна просмотра в окне графика
close all;
x = 0:.025:pi/2;
plot(x,tan(x),'-ro')
input('Для продолжения нажмите Enter ...')
axis([0 pi/2 0 5]);
input('Для продолжения нажмите Enter ...')
axis([0 pi/4 0 2]);
input('Для продолжения нажмите Enter ...')
axis([0 pi/2 0 20]);
input('Для продолжения нажмите Enter ...')
close all;

```

```

% №11.2
% Ручное управление окном графиков и его масштабами
close all;
axis manual;
axis([-4 4 -2 2]); % Установить размер окна
hold on; % Заморозить окно и его вертикальный масштаб
fplot('[-3*sin(x)/x]', [-2*pi 2*pi], '-r');
fplot('[-abs(sin(x))]', [-2*pi 2*pi]);
input('Для продолжения нажмите Enter ...')
axis([-4 4 -2 6]);
input('Для продолжения нажмите Enter ...')
hold off;
input('Для продолжения нажмите Enter ...')
axis auto;

```

ezplot и ezpolar

ezplot. Простой в использовании плоттер.

ezplot. Синтаксис:

```

EZplot (f)
EZplot (f, [min,max])
EZplot (f, [Xmin, Xmax, Ymin, Ymax])
Где f(x) – функция заданная литералом (в виде строки)

EZplot (x, y)

```

`EZplot (x, y, [[tmin,tmax]])`
Где x , y – функции заданные литералом (в виде строки)

ezplot. Описание

`ezplot (f)` отображает выражение $f = f(x)$ по умолчанию: $-2\pi < x < 2\pi$.
`ezplot (f, [min, max])` графики $f = f(x)$ над областью аргумента: $\min < x < \max$.

Для неявно определенных функций двух переменных $f = f(x, y)$:

`ezplot(f)` отображает графики $f(x, y)=0$ над областью аргумента по умолчанию $-2\pi < x < 2\pi$, $-2\pi < y < 2\pi$.

`ezplot (f, [xmin, xmax, ymin, ymax])` графики $f(x, y) = 0$ над областью аргумента $x_{\min} < x < x_{\max}$ и $y_{\min} < y < y_{\max}$.

`ezplot (f, [min, max])` графики $f(x, y) = 0$ по $\min < x < \max$ и $\min < y < \max$.

Если f является функцией переменных u и v (а не x и y), то конечные точки для области аргумента u_{\min} , u_{\max} , v_{\min} и v_{\max} сортируются по алфавиту. Таким образом, `ezplot ('u ^ 2 - v ^ 2 - 1', [- 3,2, -2,3])` отображает $u^2 - v^2 - 1 = 0$ над $-3 < u < 2$, $-2 < v < 3$,

`ezplot (x, y)` изображает параметрически определенную планарную кривую $x = x(t)$ и $y = y(t)$ для области аргумента $0 < t < 2$.

`ezplot (x, y, [tmin, tmax])` графики $x = x(t)$ и $y = y(t)$ по $t_{\min} < t < t_{\max}$.

ezplot. Примеры:

```
EZplot ('sin (x)')
```

```
ezplot ('sin (x)', [- 2 * pi 2 * pi])
```

В этом примере отображается неявно определенная функция, $x:2 - y:4 = 0$ для области аргумента $[-2, 2]$:

```
ezplot ('x ^ 2-y ^ 4', [-2, 2])
```

```
ezplot ('x ^ 2 - y ^ 2')
```

```
ezplot ('sin (x)', 'sin (y)', [- 2 * pi 2 * pi])
```

```
ezplot ('sin (x)', 'cos (y)', [- 2 * pi 2 * pi])
```

ezpolar. Простой в использовании полярный координатный плоттер

ezpolar. Синтаксис

```
ezpolar (f)
```

```
ezpolar (f, [a, b])
```

Где f – функция заданная литералом (в виде строки)

ezpolar.Описание

`ezpolar (f)` отображает полярную кривую $\rho = f(\theta)$ для области аргумента (по умолчанию $0 < \theta < 2\pi$).

`ezpolar (f, [a, b])` графики f для $a < \theta < b$.

ezpolar.Примеры

В этом примере создается полярный график функции $1 + \cos(t)$ по области $[0, 2 * \pi]$:

```
ezpolar ( '1 + cos (t)')
```

ezplot, ezpolar, polar ПРИМЕРЫ ДЛЯ ПРОВЕРКИ

```
% №12.1
help ezplot
```

```
% №1
ezplot('sin(x)')
% №12.2
ezplot('sin(x)',[-2*pi 2*pi])
% №12.3
ezplot('sin(x)/x')
```

```
% №13.1
% Документированный способ управления окном отображения
close all;
ezplot('sin(x)/x',[-16*pi 16*pi])
axis([-16*pi 16*pi -0.4 1])
% №13.2
% НЕ Документированный способ управления окном отображения
close all;
ezplot('sin(x)/x',[-16*pi 16*pi -0.4 1])
```

```
% №14
% Функции двух независимых переменных X и Y
% с одинаковым диапазоном (Например: фигуры Лиссажу).
% №14.1
ezplot('sin(x)','sin(y)',[-2*pi 2*pi])
% №14.2
ezplot('sin(x)','cos(y)',[-2*pi 2*pi])
% №14.3
ezplot('sin(x)','cos(2*y)',[-2*pi 2*pi])
% №14.4
ezplot('sin(x)','cos(3*y)',[-2*pi 2*pi])
% №14.5
ezplot('sin(x+pi/4)','cos(3*y)',[-2*pi 2*pi])
% №14.6
ezplot('sin(x)','cos(4*y)',[-2*pi 2*pi])
```

```
% №15
% Функции двух литеральных переменных A и B, которые
% вычисляются на основе независимой переменной (t)
% с заданным диапазоном. . По умолчанию
% диапазон определяется как [-2*pi, 2*pi]
% №15.1
ezplot('t*cos(t)','t*sin(t)',[0,4*pi])
% №15.2
ezplot('sin(3*t)*cos(t)','sin(3*t)*sin(t)',[0,pi])
```

```
% №16
% Функции переменных: X (независимая) и переменной Y,
% которая связана с значениями X уравнением Y(X)=0
```



```
% №16.1
ezplot('x^2 - y')
% №16.2
ezplot('x^2 - y',[-1 20])
% №16.3
ezplot('x^2 - y^2 - 1')
% №16.4
ezplot('x^3 + y^3 - 5*x*y + 1/5',[-3,3])
% №16.5
ezplot('1/y-log(y)+log(-1+y)+x - 1')
```

```
% №17
% Прорисовка Функций в полярных координатах
help ezpolar
```

```
% Функции, которая отображается кончиком вектора длиной вычисляемой
% литералом и углом поворота (t) в заданном диапазоне. По умолчанию
% диапазон определяется как [0, 2*pi]
```

```
% №17.1
ezpolar('sin(t)',[0, 2*pi])
% №17.2
ezpolar('1+cos(t)')
% №17.3
ezpolar('1+cos(t)',[0, pi])
% №17.4
ezpolar('sin(t)+cos(t)',[0, pi])
% №17.5
ezpolar('sin(t)/t', [-6*pi,6*pi])
% №17.6
ezpolar('sin(2*t)*cos(3*t)',[0,pi])
% №17.7
ezpolar('sin(tan(t))')
% №17.8
ezpolar('cos(5*t)')
% №17.9
ezpolar('cos(8*t)')
```

```
% №18.1
% Прорисовка Функций в полярных координатах
help polar
```

```
clear all;
close all;
x = 0:.01:2*pi;
polar(x,sin(2*x),'--r')
```

```
% №18.2
% Прорисовка Функций в полярных координатах
clear all;
close all;
% An M-file script to produce          % Comment lines
% "flower petal" plots
theta = -pi:0.01:pi;                  % Computations
rho(1,:) = 2*sin(5*theta).^2;
rho(2,:) = cos(10*theta).^3;
rho(3,:) = sin(theta).^2;
rho(4,:) = 5*cos(3.5*theta).^3;
for k = 1:4
```

```
polar(theta,rho(k,:))      % Graphics output
pause
end
```

ПРАКТИЧЕСКАЯ РАБОТА.

1. Выполните все примеры из теоретической части лабораторной работы.
-

Финальная редакция материала 23.08.2018г.
Воронов С.И.