Лабораторная работа №7

Графики в Octave

Смирнов-Мальцев Егор Дмитриевич

Содержание

Список иллюстраций

Цель работы

Научиться строить графики в Octave.

Задание

- Построить параметрический графики
- Построить график в полярных координатах
- Построить график неявной функции
- Построить график в комплексной области
- Построить график встроенной функции

Теоретическое введение

Декартова или прямоугольная система координат, задается двумя перпендикулярными прямыми, называемыми осями координат.

Функция считается заданной, если для каждого значения аргумента существует соответствующее значение функции. Чаще всего используют следующие способы задания функций: • табличный – числовые значения функции уже заданы и занесены в таблицу, недостаток заключается в том, что таблица может не содержать все нужные значения функции; • графический – значения функции заданы при помощи линии (графика), у которой абсциссы изображают значения аргумента, а ординаты – соответствующие значения функции; • аналитический – функция задается одной или несколькими формулами (уравнениями), при этом, если зависимость между x и y выражена уравнением, разрешенным относительно y, то говорят о явно заданной функции, в противном случае функция считается неявной.

Совокупность всех значений, которые может принимать в условиях поставленной задачи аргумент x функции y=f(x), называется областью определения этой функции. Совокупность значений y, которые принимает функция f(x), называется множеством значений функции.

Выполнение лабораторной работы

Графики параметрически заданных функций

Построим три периода циклоиды радиуса 2. Поскольку период 2π, зададим параметр на отрезке $[0, 6\pi]$ для трёх полных циклов. Определим t как вектор в этом диапазоне, затем вычислим x и y(рис. [-@fig:001],[-@fig:002]).

```
octave:2> t = linspace(0,6*pi,50);
octave:3> r = 2;
octave:4> x = r*(t-sin(t));
octave:5> y = r*(1-cos(t));
octave:6> plot(x,y)
octave:7> axis('equal');
octave:8> axis([0 12*pi 0 4])
```

Рис. 1: Построение циклоиды

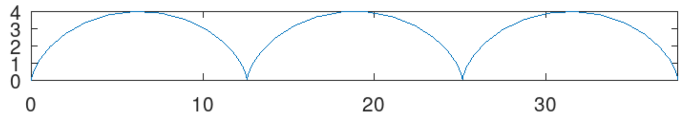
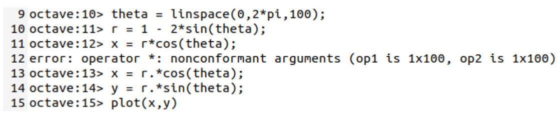


Рис. 2: Циклоида радиуса 2

Графики в полярных координатах

Построим улитку Паскаля. Для этого определим независимую переменную θ, отвечающую за угол в полярных координатах. Затем вычислим расстояние r до начала координат. Построим график, используя стандартное преобразование координат(рис. [-@fig:003],[-@fig:004]).



```
octave:10> theta = linspace(0,2*pi,100);
octave:11> r = 1 - 2*sin(theta);
octave:12> x = r*cos(theta);
error: operator *: nonconformant arguments (op1 is 1x100, op2 is 1x100)
octave:13> x = r.*cos(theta);
octave:14> y = r.*sin(theta);
octave:15> plot(x,y)
```

Рис. 3: Построение улитки Паскаля

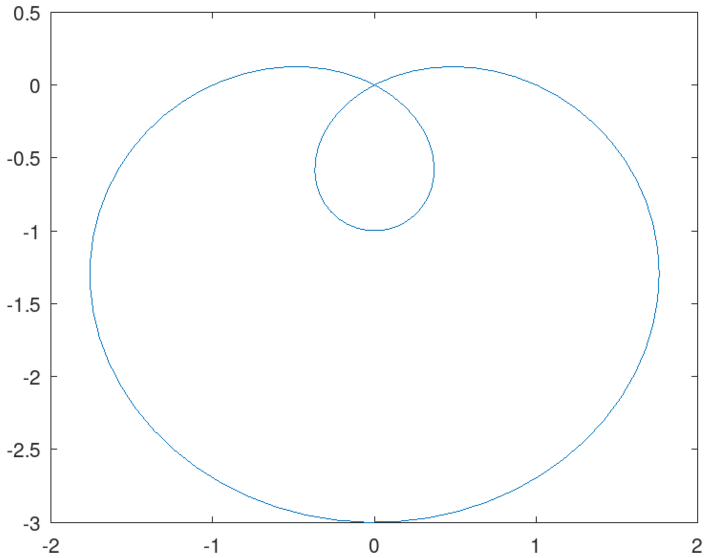


Рис. 4: График улитки Паскаля в декартовых координатах

Построим этот же график в полярных координатах, с помощью встроенной функции polar(рис. [-@fig:005],[-@fig:006]).

```
octave:17> theta = linspace(0,2*pi,50);
octave:18> r = 1 - 2*sin(theta);
octave:19> polar(theta, r)
```

Рис. 5: Построение улитки Паскаля с помощью функции polar

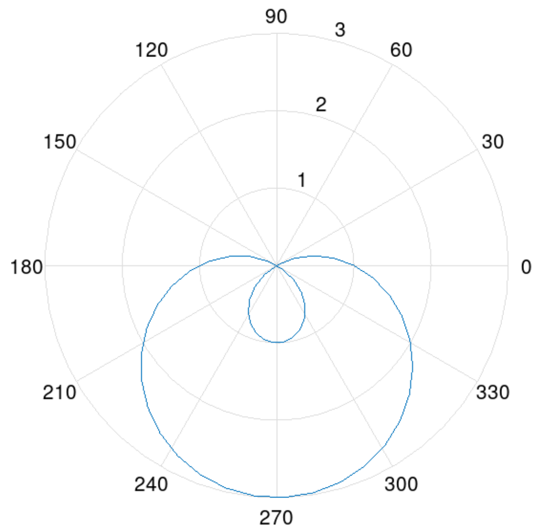


Рис. 6: График улитки Паская в полярных координатах

Графики неявных функций

Построим график кривой, заданной уравнением:

$$-x^2 - xy + x + y^2 - y = 1$$

Для этого используем функцию ezplot (рис. [-@fig:007],[-@fig:008]).

```
octave:21> f = @(x,y) -x.^2-x.*y+x+y.^2-y-1
f =

@(x, y) -x .^ 2 - x .* y + x + y .^ 2 - y - 1

octave:22> ezplot(f)
```

Рис. 7: Построение графика кривой. заданной в неявном виде

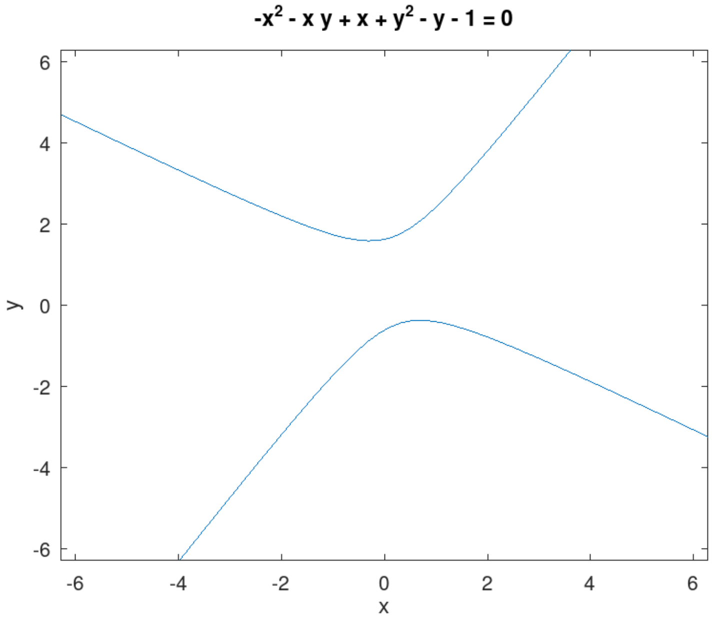


Рис. 8: График кривой. заданной в неявном виде

Построим окружность $(x-2)^2+y^2=25$. Найдем касательную к ней в точке $(-1,4)$. Для этого продиффиринцируем функцию в данной точке. Производная равна $\frac{3}{4}$. Поэтому уравнение касательной имеет вид:

$$y = \frac{3}{4}x + \frac{19}{4}$$

Построим ее график (рис. [-@fig:009],[-@fig:010]).

```
octave:24> f = @(x,y) (x-2).^2+y.^2-25
f =

@(x, y) (x - 2) .^ 2 + y .^ 2 - 25

octave:25> ezplot(f)
octave:26> ezplot(f, [-6 10 -8 8])
octave:27> print -dpng circle.png
octave:28> x = [-6:10];
octave:29> y = 3/4*x+19/4;
octave:30> hold on
octave:31> plot(x,y,'r--')
```

Рис. 9: Построение график касательной к окружности

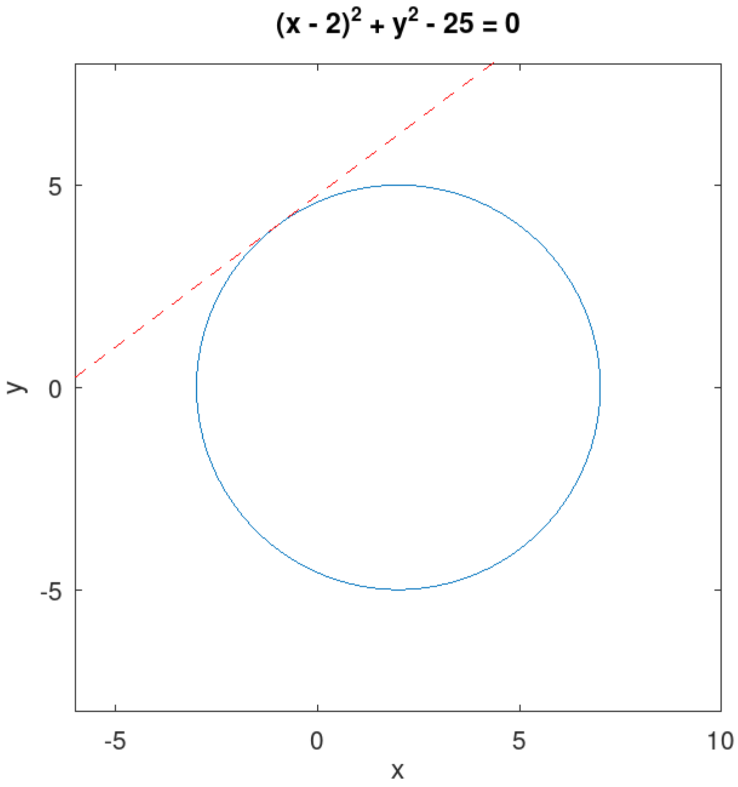


Рис. 10: График касательной к окружности

Комплексная плоскость

Зададим комплексные числа и выполним основные арифметические операции над ними (рис. [-@fig:011]).

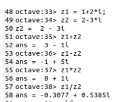


Рис. 11: Основные арифметические операции над комплексными числами

Построим график в комплексной плоскости, используя команду compass(рис.[-@fig:012],[-@fig:013]).

```
octave:40> compass(z1,'b')
octave:41> hold on
octave:42> compass(z2,'r')
octave:43> compass(z1+z2,'k--')
octave:44> legend('z_1','z_2','z1+z2')
```

Рис. 12: Построение графика в комплексной плоскости

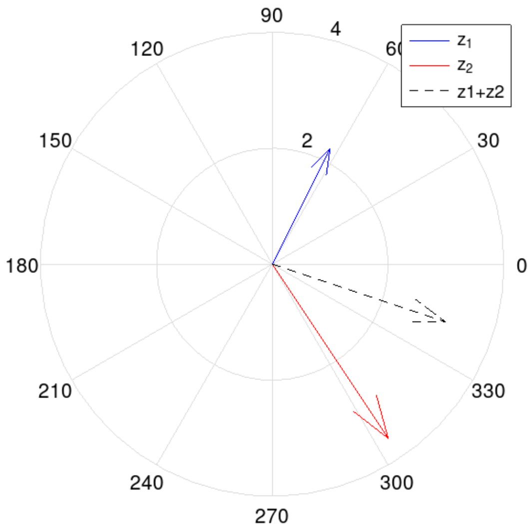


Рис. 13: График в комплексной плоскости

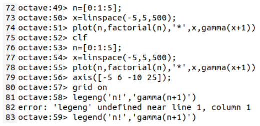
Вычислим $\sqrt[3]{-8}$. Делая это возведением в степень $1/3$, получим комплексный корень. Для получения вещественного корня необходимо использовать функцию nthroot(рис. [-@fig:014]).

```
octave:46> (-8)^(1/3)
ans =  1.0000 + 1.7321i
octave:47> ans^3
ans = -8.0000e+00 + 2.2204e-15i
octave:48> nthroot(-8,3)
ans = -2
```

Рис. 14: Нахождение корня из отрицательного числа

Построение графиков встроенных функций

Построим функции $\Gamma(x+1)$ и $n!$ на одном графике(рис. [-@fig:015],[-@fig:016]).



```
octave:49> n=[0:1:5];
octave:50> x=linspace(-5,5,500);
octave:51> plot(n,factorial(n),'*',x,gamma(x+1))
octave:52> clf
octave:53> n=[0:1:5];
octave:54> x=linspace(-5,5,500);
octave:55> plot(n,factorial(n),'*',x,gamma(x+1))
octave:56> axis([-5 6 -10 25]);
octave:57> grid on
octave:58> legeng('n!','gamma(n+1)')
error: 'legeng' undefined near line 1, column 1
octave:59> legend('n!','gamma(n+1)')
```

Рис. 15: Построение графиков $\Gamma(x+1)$ и $n!$

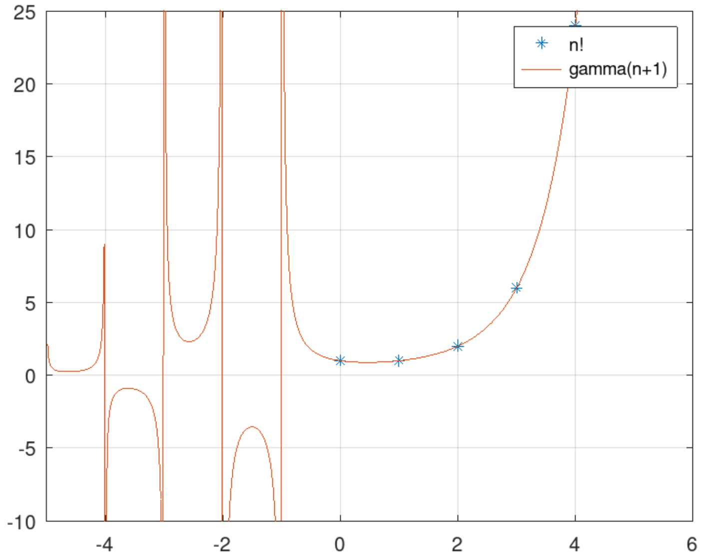


Рис. 16: Графики $\Gamma(x+1)$ и $n!$

Вертикальные асимптоты в районе отрицательных чисел на графике являются артефактами вычисления. Уберём их, разделив область значений(рис. [-@fig:017],[-@fig:018]).

```
octave:62> x1 = linspace(-5,-4,500);
octave:63> x2 = linspace(-4,-3,500);
octave:64> x3 = linspace(-3,-2,500);
octave:65> x4 = linspace(-2,-1,500);
octave:66> x5 = linspace(-1,-5,500);
octave:67> plot(x1,gamma(x1+1))
octave:68> hold on
octave:69> plot(x2,gamma(x2+1))
octave:70> plot(x3,gamma(x3+1))
octave:71> plot(x4,gamma(x4+1))
octave:72> plot(x5,gamma(x5+1))
octave:73> axis([-5 6 -10 25]);
octave:74> plot(n,factorial(n),'*')
octave:75> legend('n!',"\\Gamma(n+1)")
octave:76> x5 = linspace(-1,5,500);
octave:77> clf
octave:78> plot(x1,gamma(x1+1))
octave:79> hold on
octave:80> plot(x2,gamma(x2+1))
octave:81> plot(x3,gamma(x3+1))
octave:82> plot(x4,gamma(x4+1))
octave:83> plot(x5,gamma(x5+1))
octave:84> axis([-5 6 -10 25]);
octave:85> plot(n,factorial(n),'*')
octave:86> legend('n!',"\\Gamma(n+1)")
```

Рис. 17: Построение графика $\Gamma(x+1)$ без вертикальных асимптот

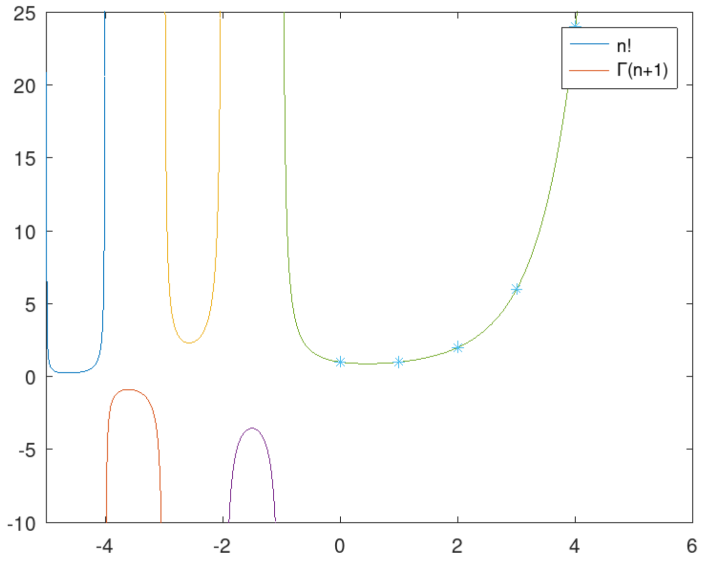


Рис. 18: График $\Gamma(x+1)$ без вертикальных асимптот

Выводы

В результате выполнения работы мы научились строить двумерные графики в декартовых и полярных координатах в Octave.

Список литературы