Отчёт по лабораторной работе №7

Графики

Сырцов Александр Юрьевич

Содержание

List of Tables

List of Figures

1 Цель работы

Познакомиться с возможностями Octave в построении графиков неявных, полярных, комплексных и специальных функций и научиться их применению.

2 Задание

• Сделать отчёт по лабораторной работе в формате Markdown. • В качестве ответа предоставить отчёты в 3 форматах: pdf, docx и md (в архиве, поскольку он должен содержать скриншоты, Makefile и т.д.).

3 Выполнение лабораторной работы

3.1 Параметрические графики

1. Дана параметрически заданная функция циклоиды – переодическая кривая, которая задана окружностью радиусом `r` и углом `t` катящейся по прямой. Необходимо построить график функции с радиусом 2 и промежутком от 0 до 6π, чтобы отобразить 3 полных периода.

$$x = r(t - \sin(t))$$

,

$$y = r(1 - \cos(t))$$

Для построения графика задал вектор `t` как вектор значений, а далее определил радиус и по этим параметрам определил `x` и `y` по функциям выше. Построение графика обычной функции в декартовых координатах практически не отличается от параметрической функции, так как тут точно так же есть переменная-вектор, по которой определяется функция. Отличие в том, что в параметрической функции `x` и `y` являются двумя отдельными функциями, а вектор представляет из себя независимую переменную (рис. -fig. 3.1).

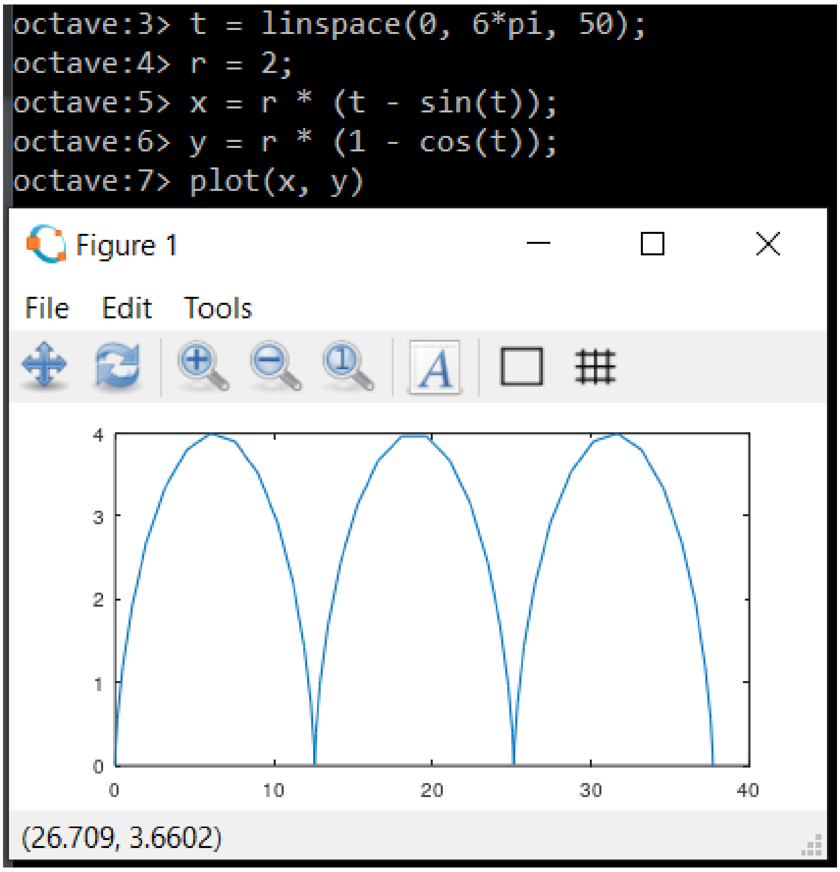

Figure 3.1: График циклоиды

2. Подгоняю оси функцией `axis()`. Порядок обеих операций ни на что не влияет (рис. -fig. 3.2).

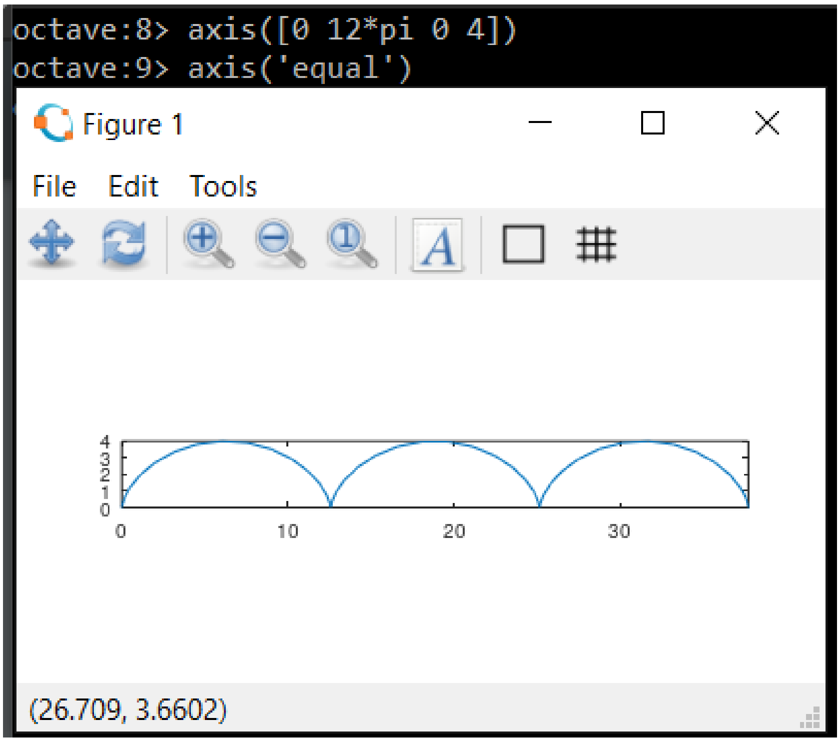

Figure 3.2: График циклоиды с подогнанными осями

3. Для более презентабельного вида графика можно поменять подписи к оси x, чтобы числа были в виде радиан. Здесь я просто демонстрирую ещё одну возможность Octave (рис. -fig. 3.3).

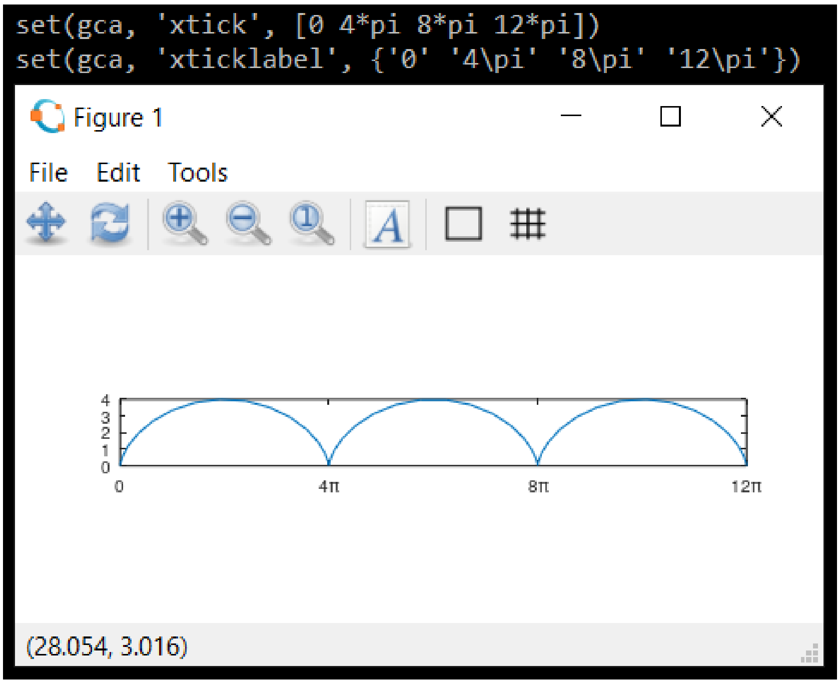

Figure 3.3: Более точная подпись осей

4. Сохраняю требуемый график в двух форматах. В формат pdf фигура сохранена через две разные функции, из-за чего файл `cycloid.pdf` был перезаписан (рис. -fig. 3.4).

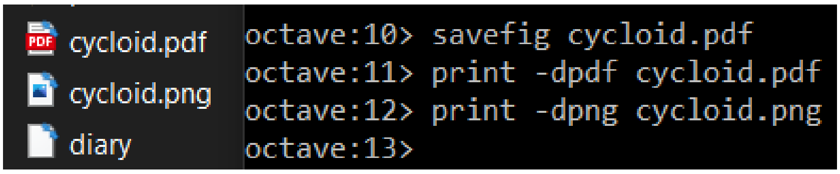

Figure 3.4: Сохранение циклоиды

3.2 Графики функций в полярных координатах

5. Дана кардиоида (улитка Паскаля). Построение не отличается от параметрической функции. Для перехода к полярным координатам используется следующая замена:

$$x = r\cos(\theta)$$

,

$$y = r\sin(\theta)$$

Нам уже задана функция в полярных координатах, поэтому замену я использую для отображения графика на декартовом пространстве. Сначала определяю вектор значений для угла тэта и определяю с его помощью радиус `r`, а для отображения графика подставляю их в формулы выше, отчищаю фигуру и вывожу график через полученные значения x, y(рис. -fig. 3.5).

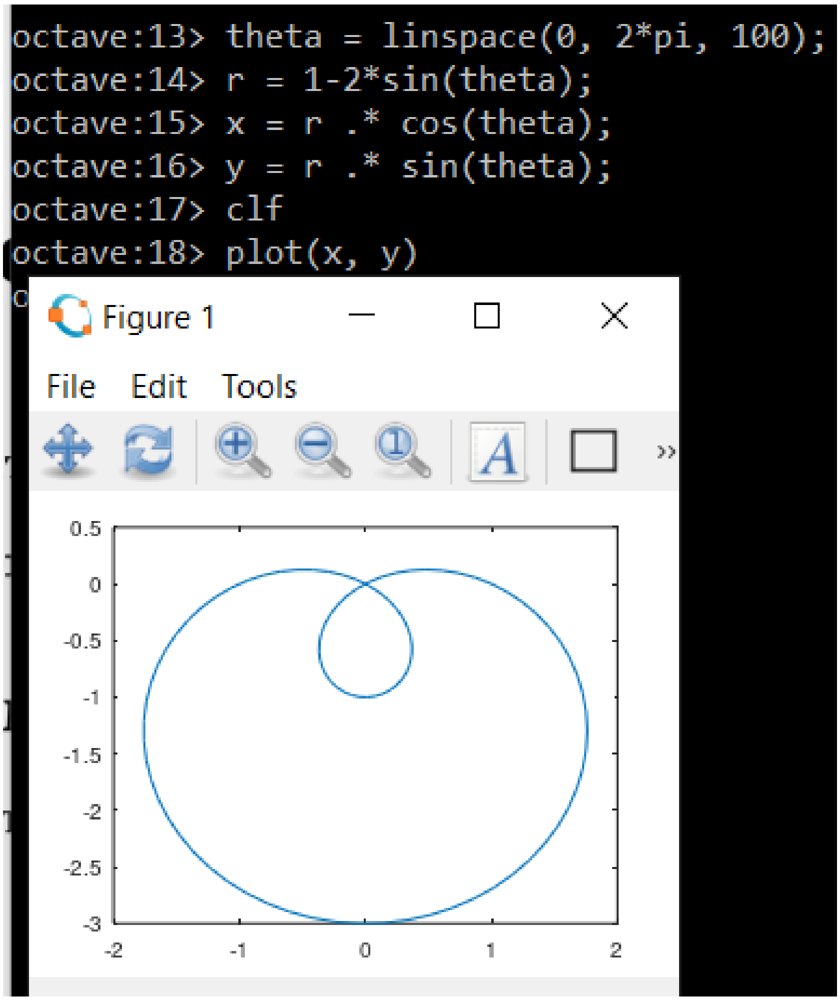

Figure 3.5: Кардиоида в декартовых координатах

6. Сохраняю фигуру (рис. -fig. 3.6).

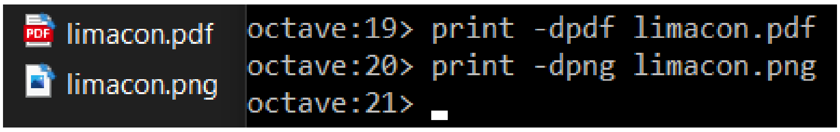

Figure 3.6: Сохранение кардиоиды

7. Для построения графика в полярных координатах проделываю предыдущие шаги, не определяя `x`, `y`, а используя специальную функцию `polar()`. Ключевое отличие в том, что полярная система координат не имеет осей, а лишь радиус, который меняется в зависимости от угла поворота (рис. -fig. 3.7).

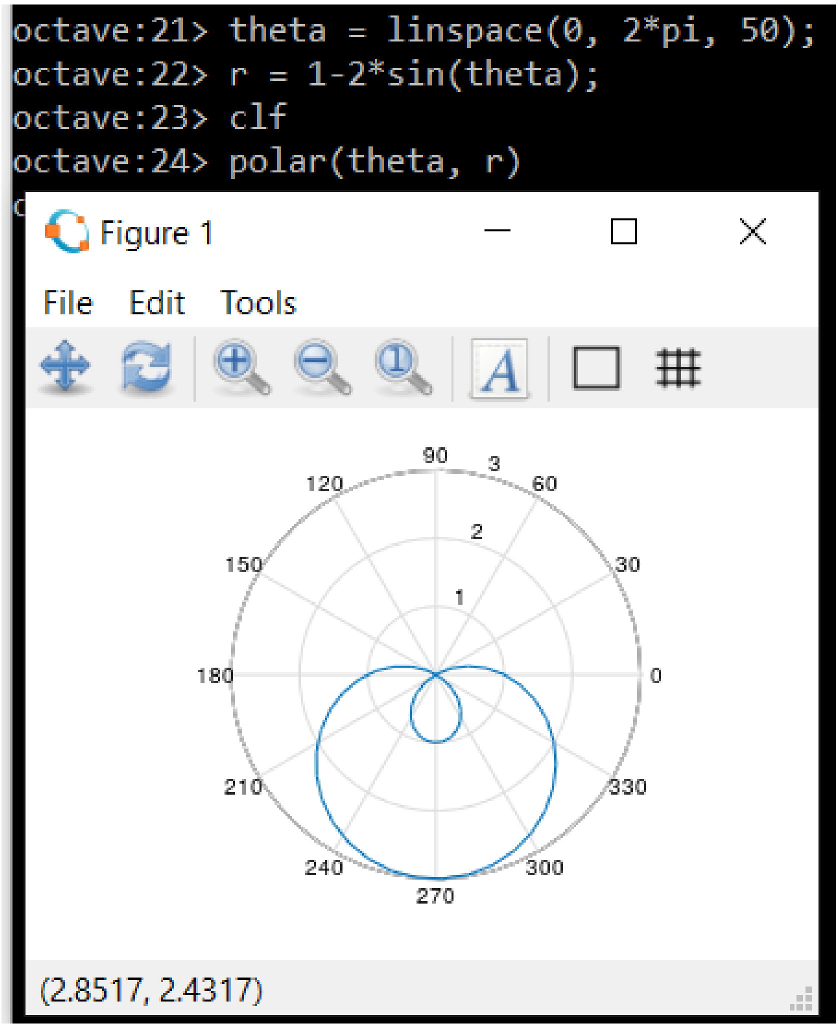

Figure 3.7: Кардиоида в полярных координатах

8. Сохраняю график (рис. -fig. 3.8).

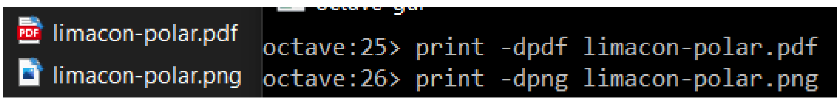

Figure 3.8: Сохранение кардиоиды в полярной системе координат

3.3 Графики неявных функций

9. В качестве функции. заданной неявно. дано уравнение кривой второго порядка. Для её построения определяю уравнение в виде функции, для

удобства это лямбда-функция `f`. Её достаточно передать в функцию первого рода `ezplot()`, которая автоматически строит график. По графику видно, что заданное уравнение является каноническим уравнением гиперболы под наклоном (рис. -fig. 3.9).

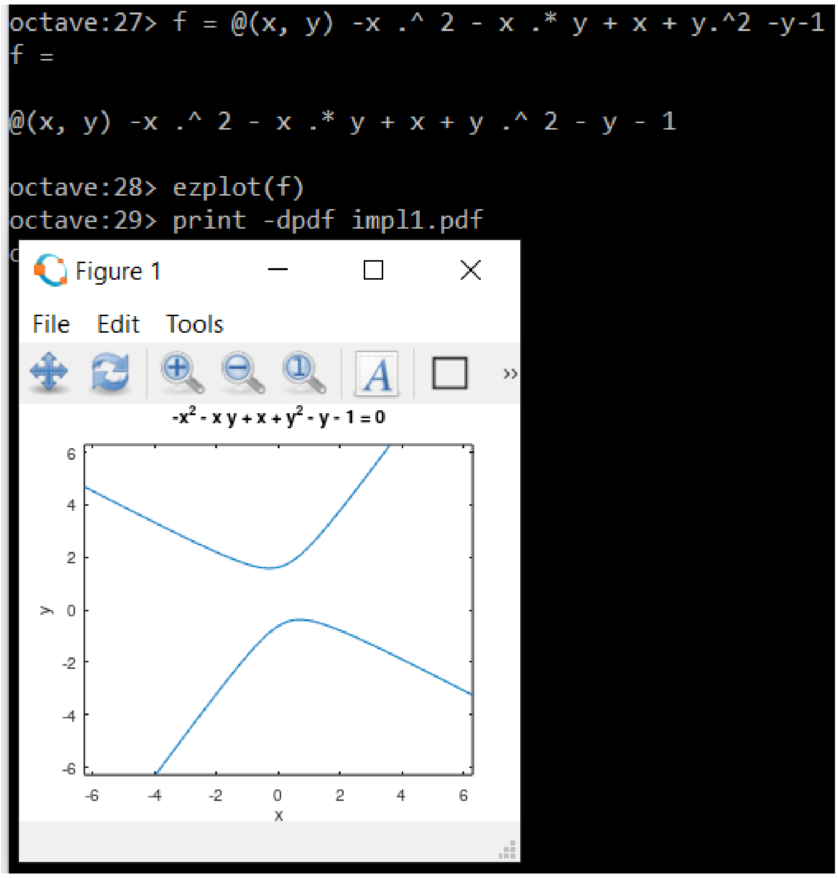

Figure 3.9: Неявная функция гиперболы

10. Теперь необходимо найти касательную в точке [-1, 4] к окружности. Уравнение окружности тоже неявно заданное, так что построение окружности происходит так же через лямбда-функцию и `ezplot()`. Дополнительно подгоняем оси вторым параметром (рис. -fig. 3.10).

$$(x-2)^2 + y^2 = 25$$

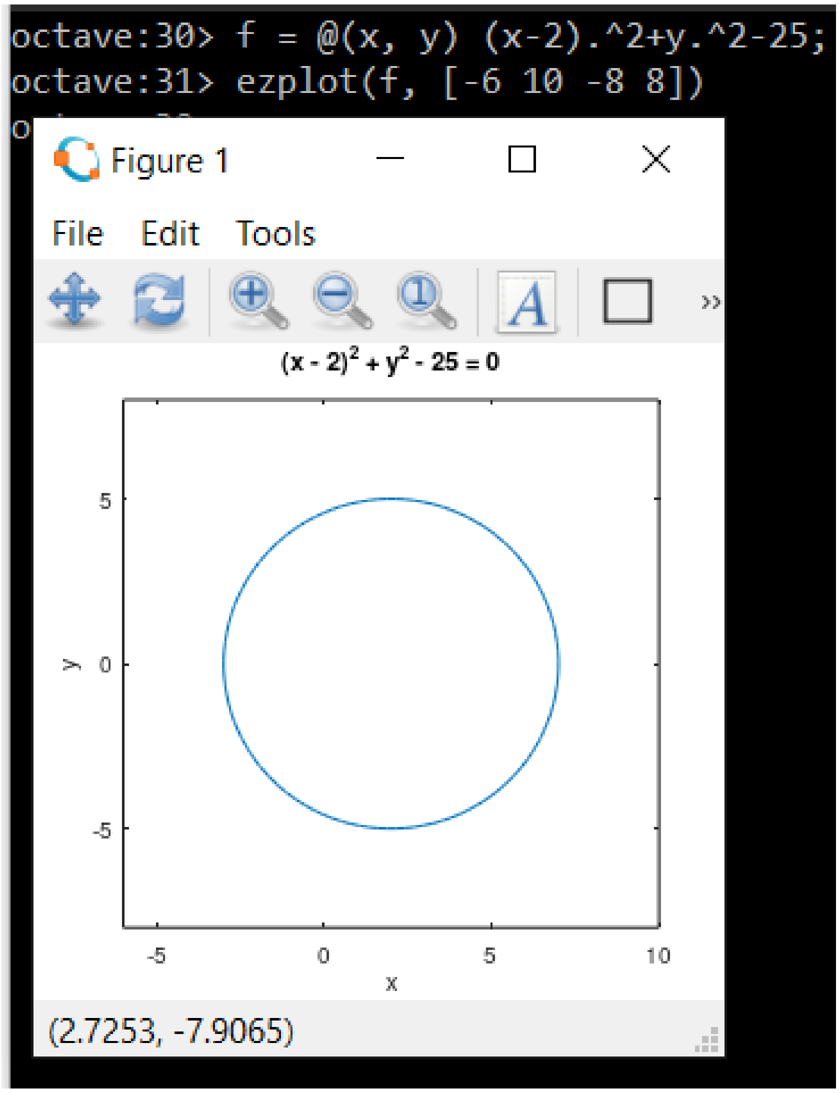

Figure 3.10: Окружность

11. Чтобы найти касательную к неявной функции, как, впрочем, и к любой другой функции, необходимо продифференцировать:

$$\frac{d}{dx}((x-2)^2) + \frac{d}{dx}(y^2) = 0$$

,

$$2(x-2) + 2y\frac{d}{dx}(y(x)) = 0$$

,

$$\frac{dy(x)}{dx} = \frac{2-x}{y}$$

Далее подставив координаты точки в производную наодим коэффицент наклона 3/4, а в конце получаем уравнение прямой

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

Строим график прямой на графике с окружностью, определив вектор x от -6 до 10 включительно и y по найденной формуле. Сразу сохраняю фигуру (рис. -fig. 3.11).

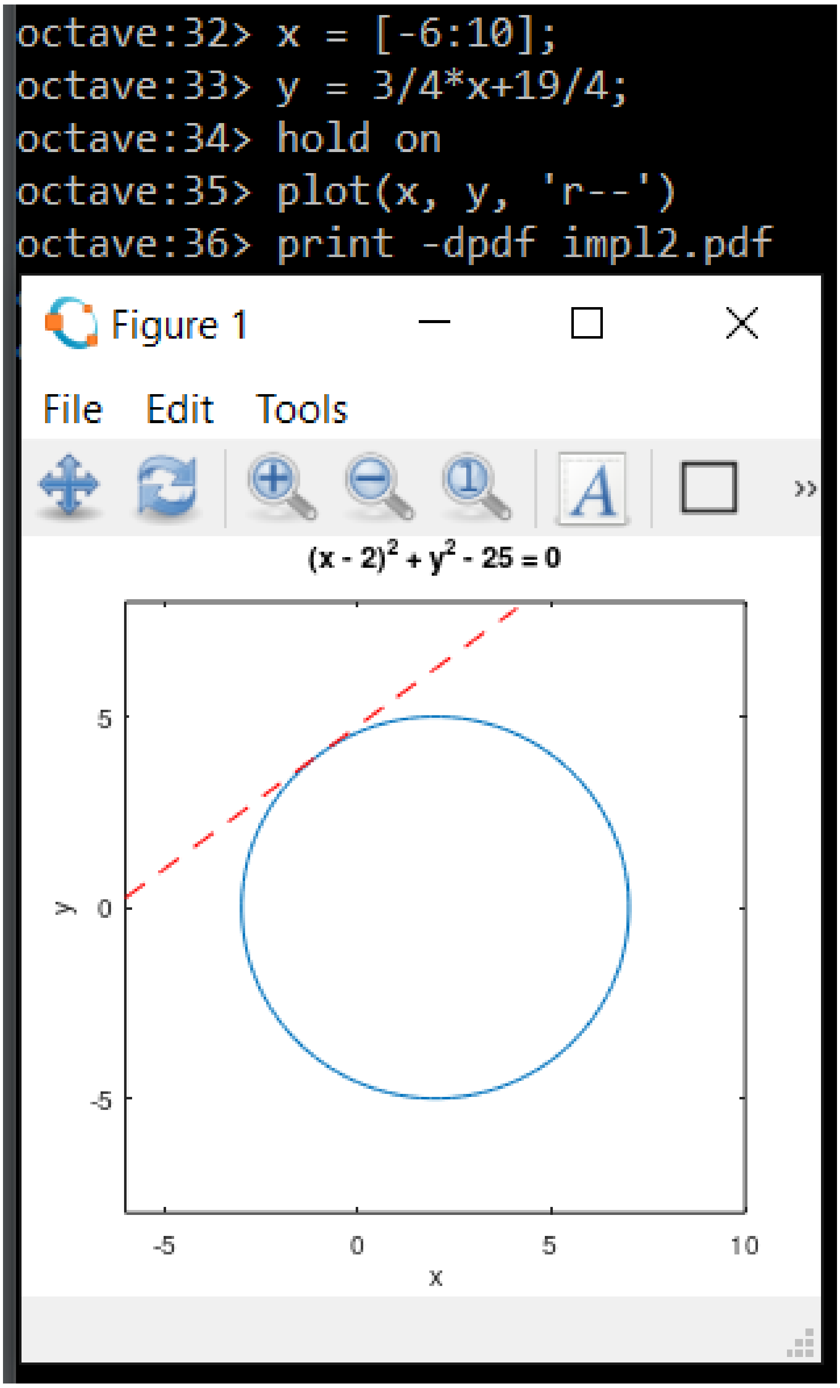

Figure 3.11: Окружность с касательной

3.4 Комплексные числа

12. Задаю комплексные числа `z1` и `z2` и провожу основные операции: сложение, умножение и деление (рис. -fig. 3.12).

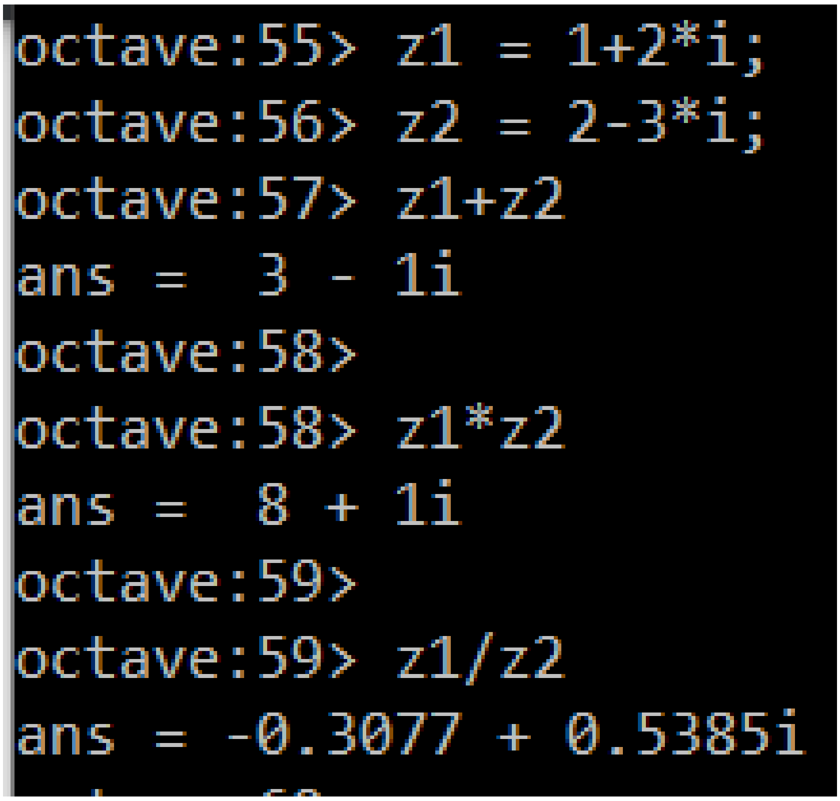

Figure 3.12: Комплексные операции

13. Отчистив фигуру, изображаю комплексные прямые и их сумму, используя функцию `compass()`, задаю легенду и сохраняю фигуру (рис. -fig. 3.13).

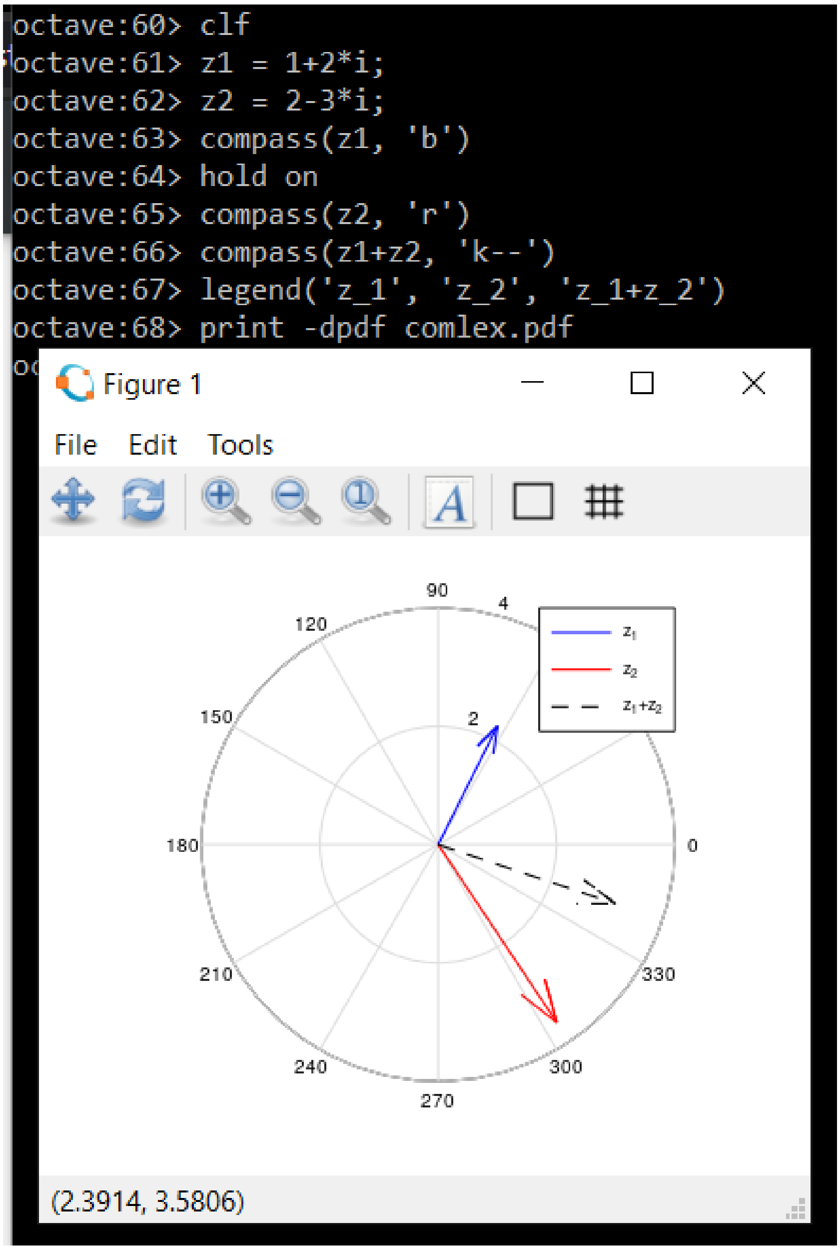

Figure 3.13: График комплексных прямых

14. Чтобы легенда не перекрывала поле или данные в принципе можно прописать более удачную позицию вне поля отображения графика (рис. - fig. 3.14).

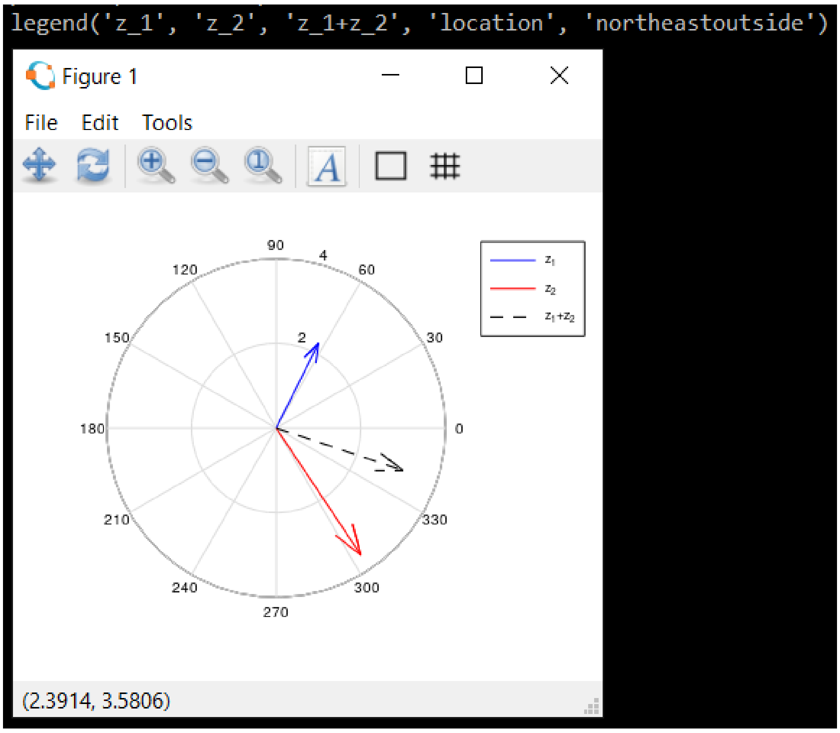

Figure 3.14: Исправленная легенда

15. При извлечении корня из отрицательного числа Octave автоматически возвращает комплексное число, поэтому, при извлечении корня третей степени из -8, вышло очень кривое число (рис. -fig. 3.15).

```
octave:70> (-8)^(1/3)
ans =  1.0000 + 1.7321i
```

Figure 3.15: Неоднозначный корень

Далее возвожу это число в третью степень, чтобы убедиться, что -8 с некоторым допущением можно получить (рис. -fig. 3.16).

```
octave:71> ans^3
ans = -8.0000e+00 + 2.2204e-15i
```

Figure 3.16: Неточный результат

Используя встроенную функцию получил действительный корень исходного выражения в обход стандартной механики языка, о которой сказано выше (рис. -fig. 3.17).

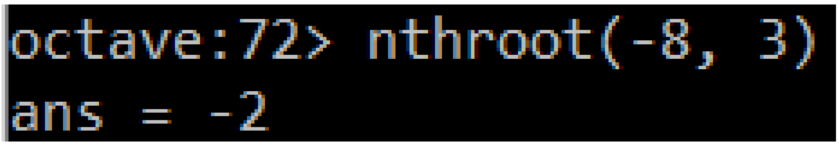

Figure 3.17: Точный результа с помощью *nthroot()*

3.5 Специальные функции

16. В Octave есть и специальные функции, вроде гамма-функции, которая соответствует следующему выражени:

$$\Gamma(n) = (n-1)!$$

,

$$\text{аналогично } n! = \Gamma(n+1)$$

Второе я и изображаю на графике. Задаю вектор значений `n` от 0 до 5 для факториала и вектор значений для функции гамма от выражения `x + 1`. Подогнал оси, включил сетку, задал легенду и сохранил график. По отрезку, на которому задан факториал видно, что выражения выше соответствуют истине (рис. -fig. 3.18).

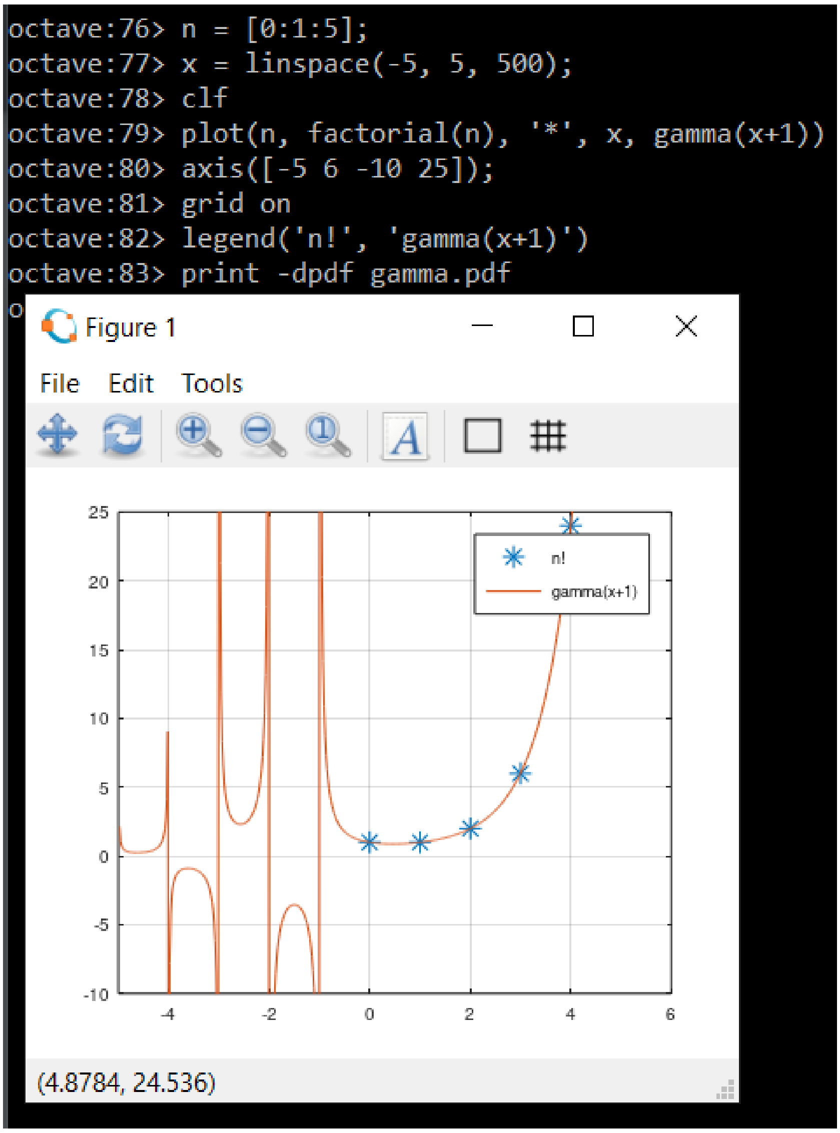

Figure 3.18: Гамма-функция и факториал

17. К сожалению, график имеет артефакты на горизонтальных асимтотах, поэтому задаю частичные промежутки, исключая промежутки с артефактами. Промежутков 5: от x1 до x5. Для отображения всех промежутков на фигуре отображаем их как дополнительные графики. После этого вновь задаю легенду с уже более презентабильным названием для гамма-функции и по стандарту сохраняю фигуру (рис. -fig. 3.19).

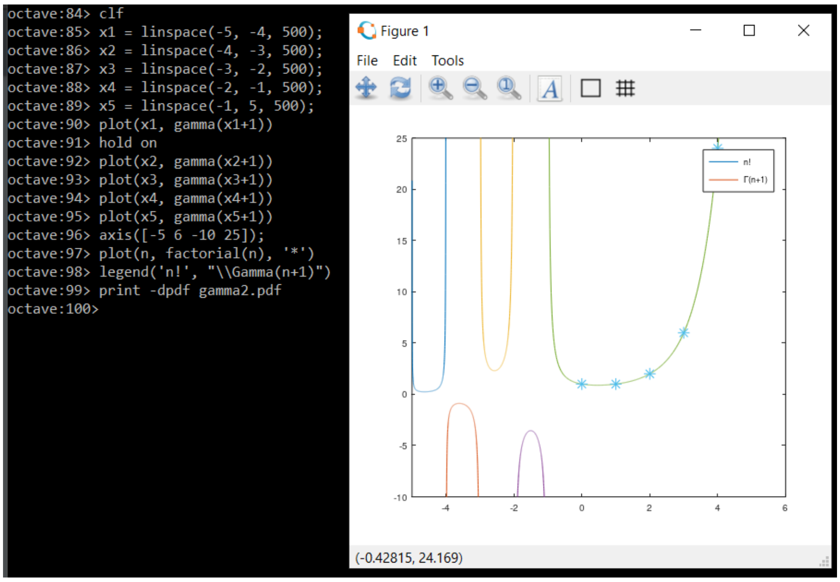

Figure 3.19: Гамма-функция и факториал без артефактов

Примечание: аналогичного эффекта, возможно, можно добится изменением мелкости разбиения. Во всяком случае в подобных языках такая ситуация возникает.

4 Выводы

Я познакомился с принципами построения параметрических и специальных функций в Octave, функциями `compass()`, `polar()` и `ezplot()` и смог применить их для создания наглядных фигур.