Отчет по лабораторной работе 1

Генералов Даниил, НПИбд-01-21, 1032202280

Содержание

Список иллюстраций

Список таблиц

1 Цель работы

В рамках этой лабораторной работы требуется познакомится с языком программирования GNU Octave, и на основе его узнать про модуляцию сигналов.

2 Задание

1.3.1. Построение графиков в Octave 1.3.2. Разложение импульсного сигнала в частичный ряд Фурье 1.3.3. Определение спектра и параметров сигнала 1.3.4. Амплитудная модуляция 1.3.5. Кодирование сигнала. Исследование свойства самосинхронизации сигнала # Выполнение лабораторной работы

Сначала в рабочей директории я создал файл с кодом, чтобы посчитать и нарисовать на графике две функции и сохранить их в файл.

Рис. 2.1: octave

Для того, чтобы вывод программы оставался на экране, я добавил в конце

файла команду `_ = input("...")`. После запуска этого файла в Octave я получаю окно с графиком введеной функции.

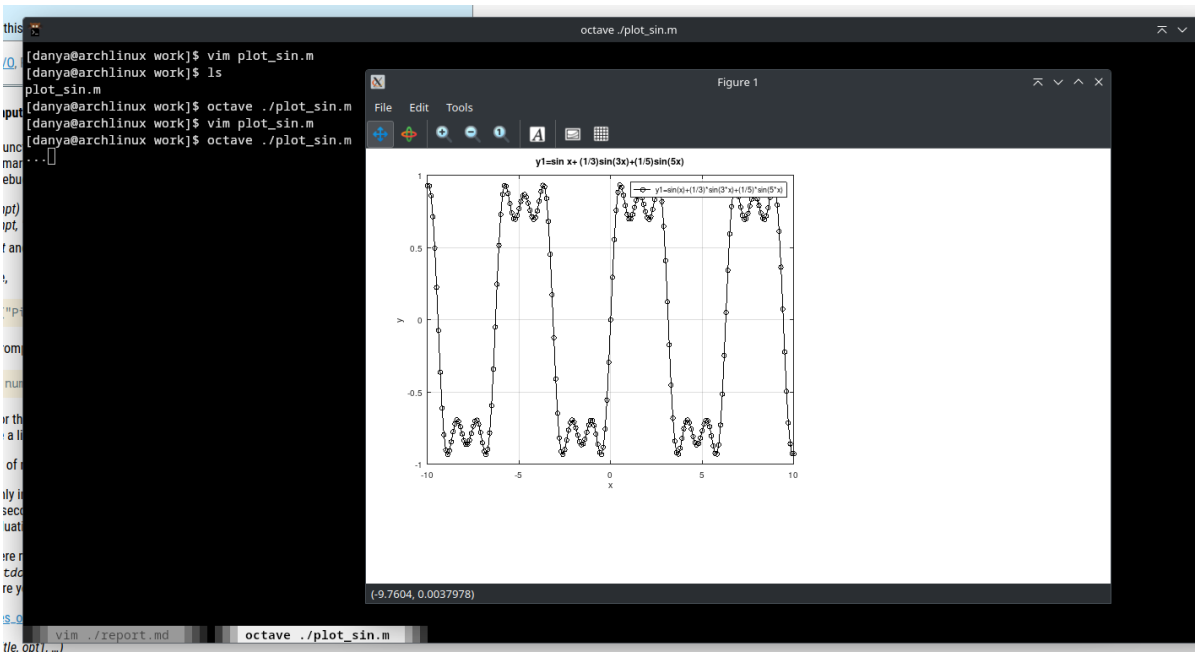

Рис. 2.2: plot

После этого я скопировал этот файл и сделал так, чтобы отображались два графика на одном рисунке.

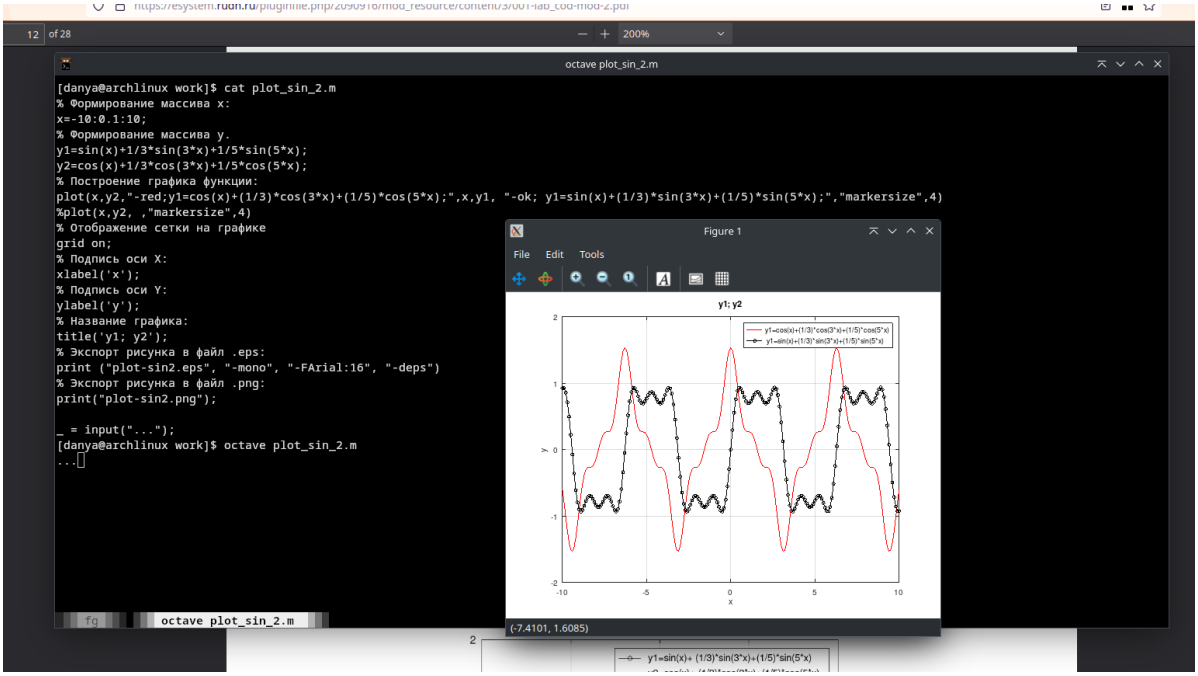

Рис. 2.3: plot

После этого я скопировал программу, которая выводит несколько разных приближений квадратной волны, добавив сохранение рисунка в файл.

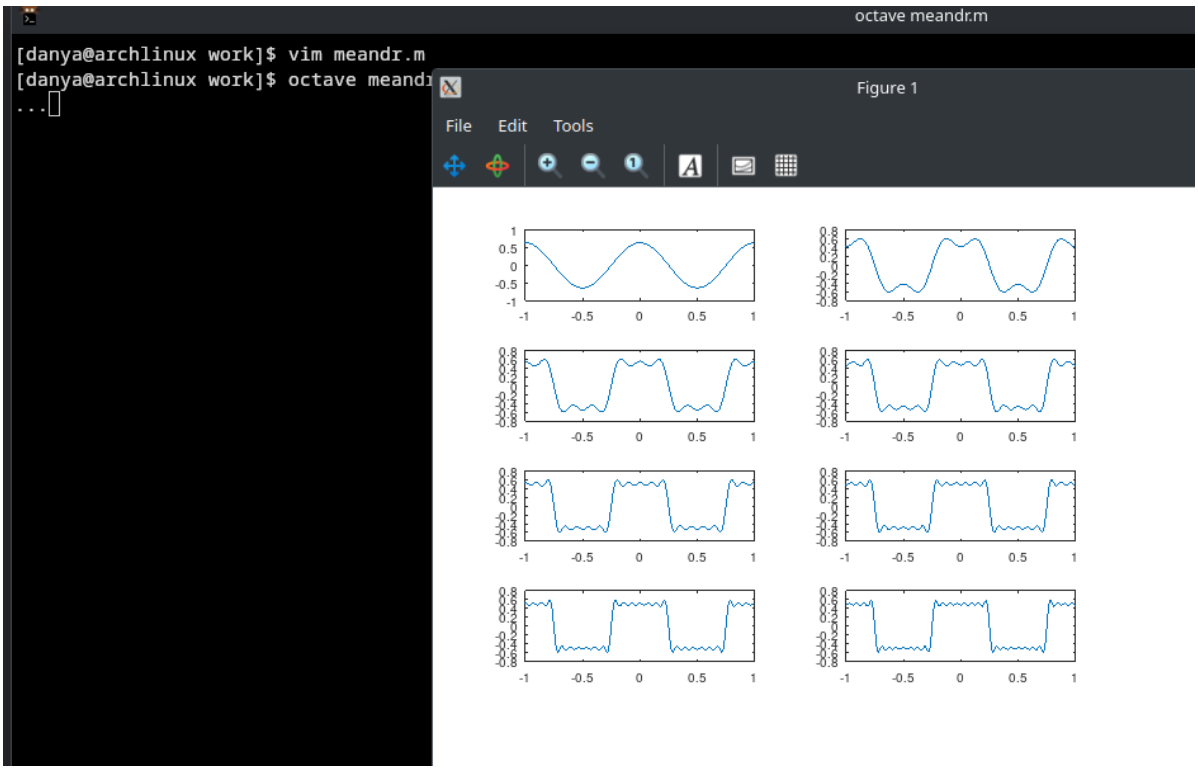

Рис. 2.4: plot

Затем я изменил формулу на альтернативную, которая задается через синусы, а не косинусы. Для этого нужно было изменить генерацию массива гармоник, а также убрать строчку, меняющую знак каждого второго элемента массива коэффициентов.

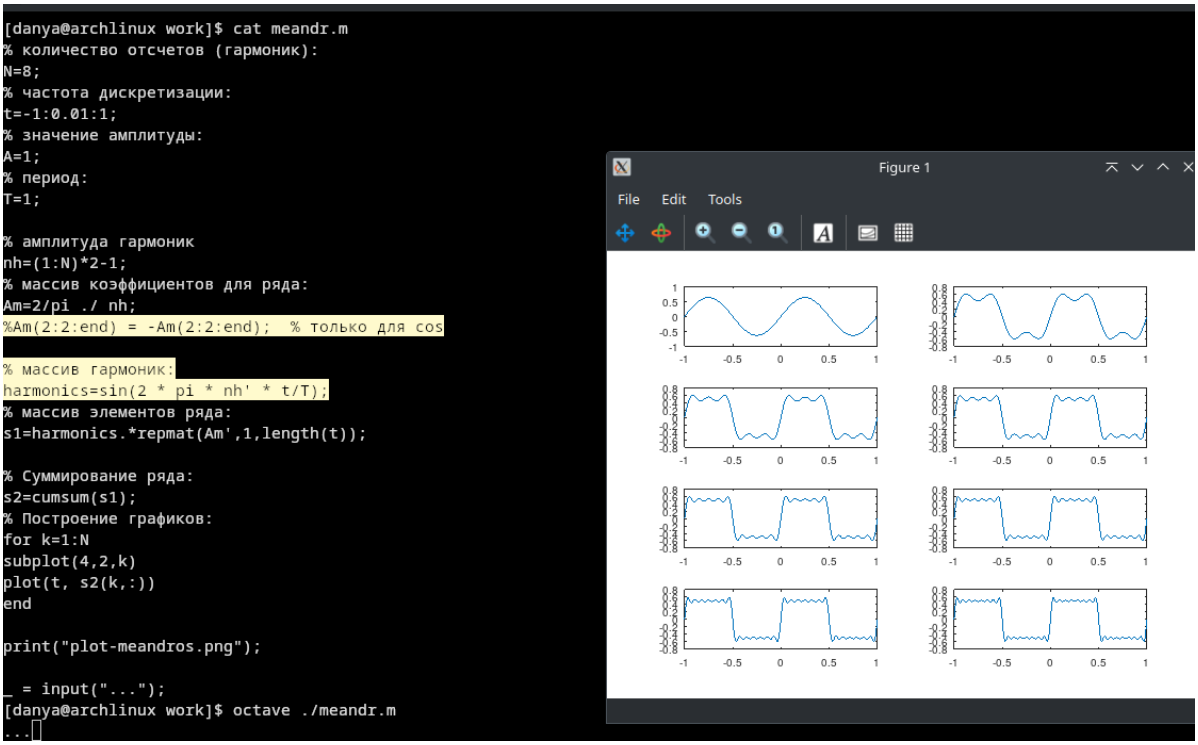

Рис. 2.5: plot

После этого я скопировал программу, которая получает сигнал и вычисляет его преобразование Фурье. Она генерирует два графика, которые показываются в одном окне один за другим. К счастью, они также сохраняются на диск, поэтому их можно посмотреть в программе просмотра изображений.

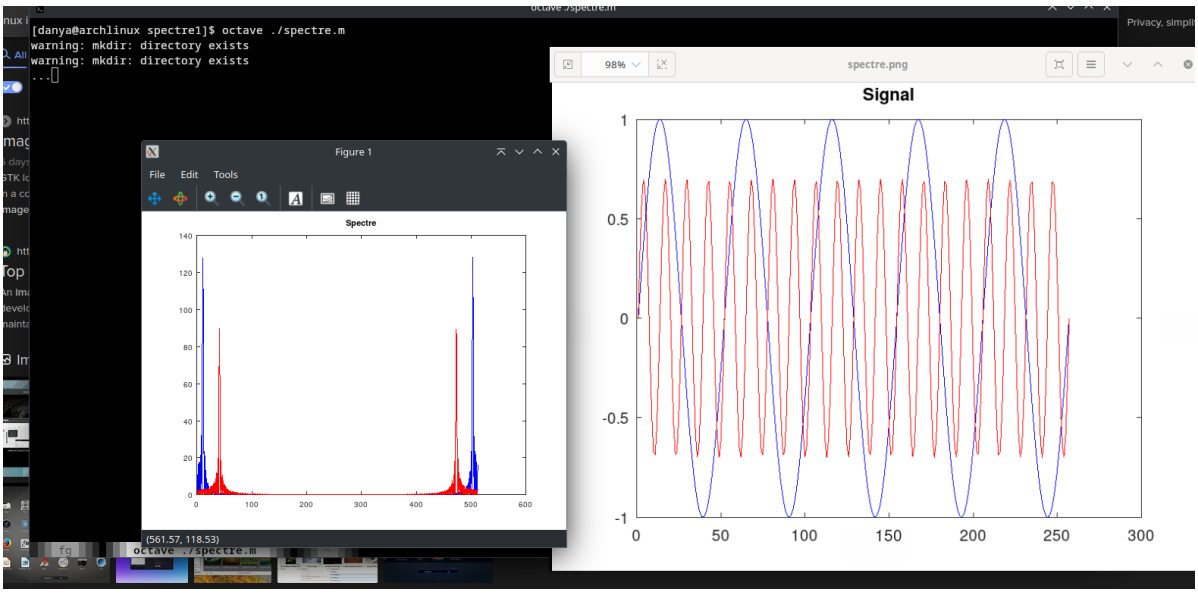

Рис. 2.6: plot

Для исправления артифактов, связанных с тем, что преобразование Фурье находит высокочастотные гармоники сигнала, мы ограничиваем отображение диапазоном от 0 до 100, что можно увидеть, сравнив графики до и после этого изменения.

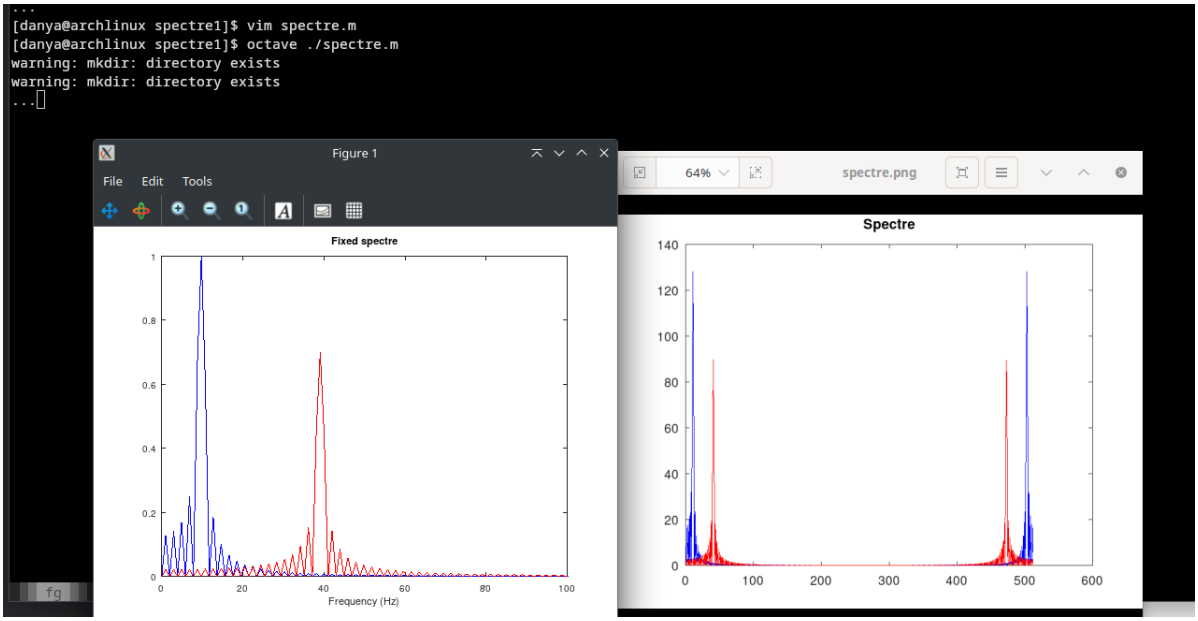

Рис. 2.7: plot

Затем я запустил вторую программу, которая рисует тот же самый график преобразования Фурье для суммы сигналов. Этот график для суммы сигналов похож на сумму графиков для отдельных сигналов, что демонстрирует свойство преобразования Фурье.

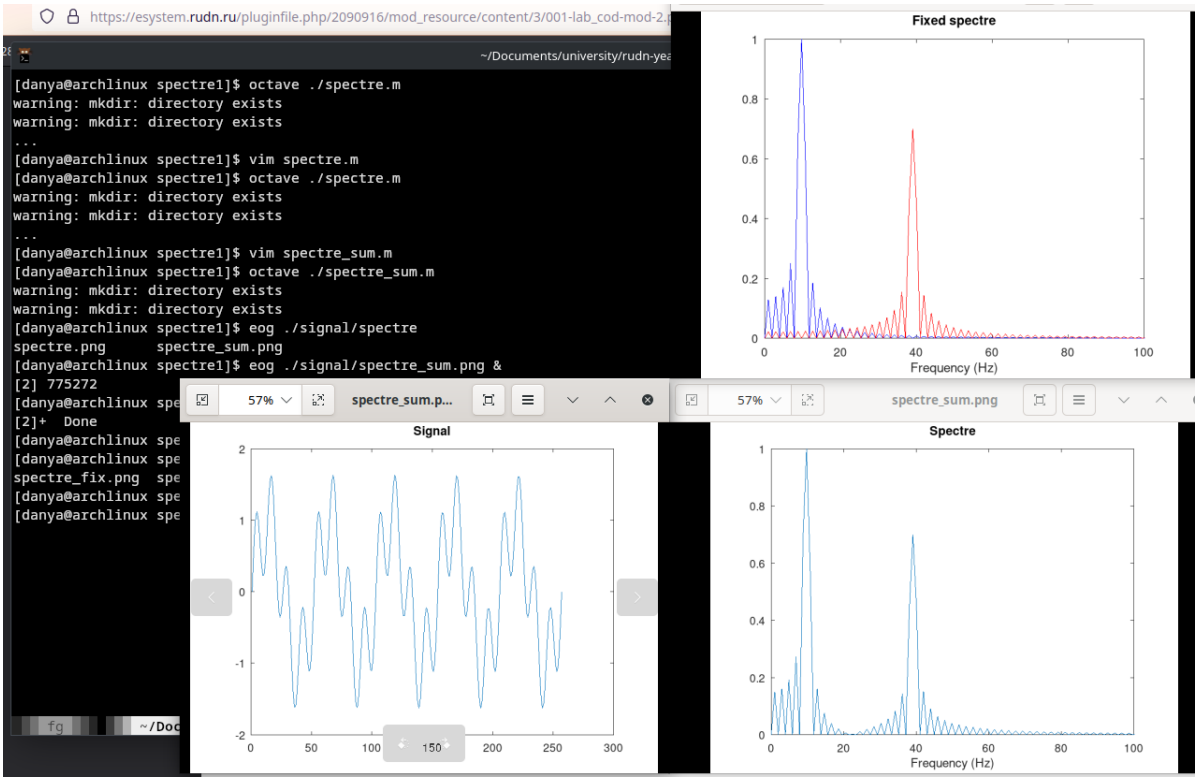

Рис. 2.8: plot

После этого я взял программу, которая выполняет амплитудную модуляцию сигнала, поточечно умножая его с несущей волной. По полученному графику преобразования Фурье видно, что получившийся сигнал слабо коррелирует с частотой несущей волны 50Гц, но сильно коррелирует с (50+5)Гц и (50-5)Гц; 5Гц – это именно частота модулируемого сигнала.

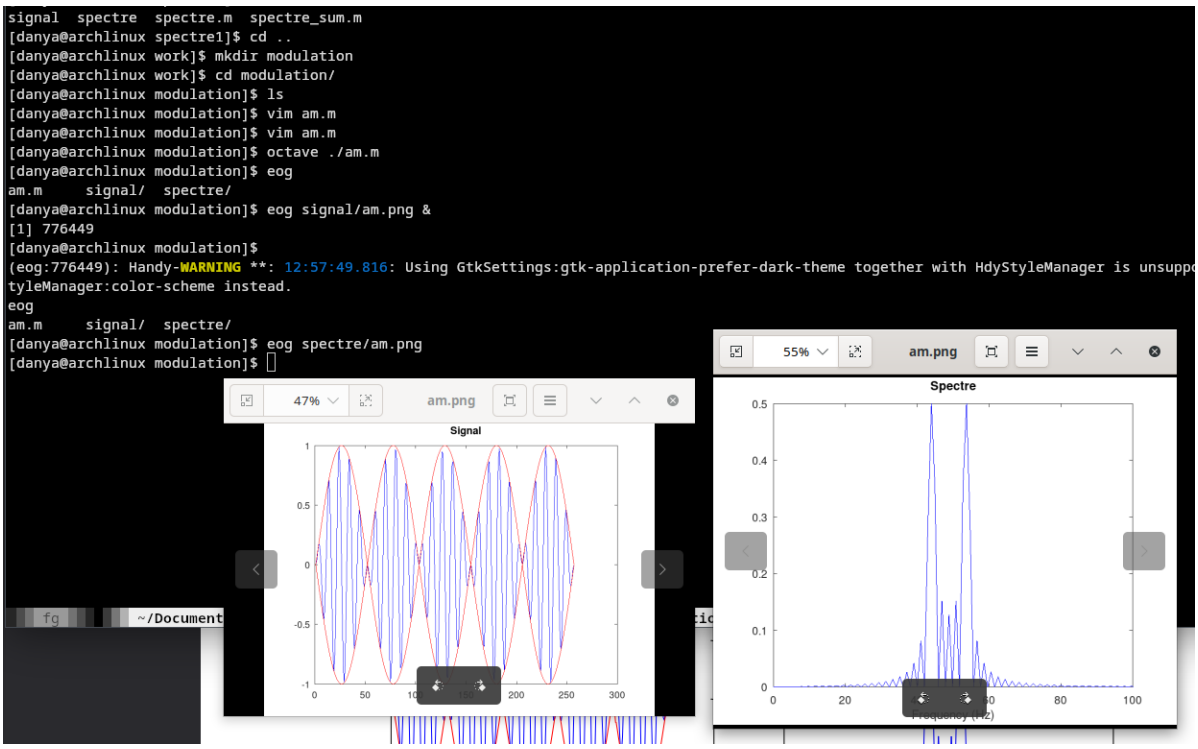

Рис. 2.9: plot

После этого я установил библиотеки `control` и `signal`, а затем скопировал файлы с кодом, чтобы использовать их. После запуска я получил папки с несколькими графиками, которые показывают результаты кодирования информации с помощью каждого из кодов.

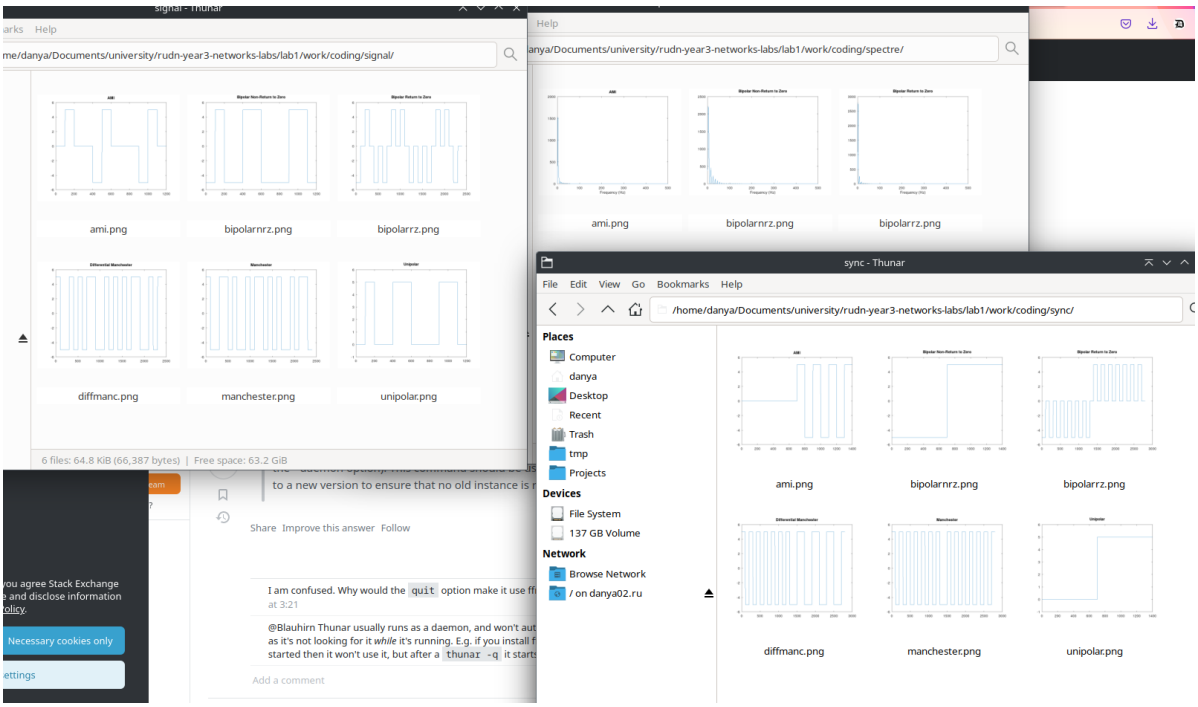
ami.png
bipolarnrz.png
bipolarrz.png
diffmanc.png
manchester.png
unipolar.png
6 files: 64.8 KiB (66,387 bytes) | Free space: 63.2 GiB
sync - Thunar
File Edit View Go Bookmarks Help
/home/danya/Documents/university/rudn-year3-networks-labs/lab1/work/coding/sync/
Places
Computer
danya
Desktop
Recent
Trash
tmp
Projects
Devices
File System
137 GB Volume
Network
Browse Network
/ on danya02.ru

Рис. 2.10: plot

3 Выводы

Я получил опыт работы с Octave для расчета и визуализации данных на примере сигналов и их модуляции.

Самая полезная информация из рисунков разных методов кодирования – это то, как они ведут себя при наличии повторяющихся битов. В папке `sync` показаны примеры того, как кодируется последовательность из множества нулей, а затем множества единиц – вопрос в том, как из сигнала определить конец одного символа и начало другого. Это можно сделать для нуля и единицы в Манчестерском, дифференциированом Манчестерском, и биполярном RZ-кодах; только для единицы – в AMI-коде, и нельзя вообще для униполярного и биполярного NRZ-кода.