

ЛАБОРАТОРНАЯ РАБОТА №2
ПО ПРЕДМЕТУ «ЧАСТОТНЫЕ МЕТОДЫ»
ПО ТЕМЕ «ПРЕОБРАЗОВАНИЕ ФУРЬЕ»

Лектор: Перегудин А. А.
Практик: Пашенко А. В.
Студент: Румянцев А. А.
Поток: ЧАСТ.МЕТ. 1.3

Факультет: СУиР
Группа: R3241

Содержание

1 Введение	2
2 Задание 1. Вещественное	3
2.1 Прямоугольная функция	3
2.2 Рассмотрим треугольную функцию следующего вида	4

1 Введение

В заданиях 1 и 2 используется унитарное преобразование Фурье к угловой частоте ω . Подсчет Фурье-образа производится по формуле ниже

$$\hat{f}(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(t) e^{-i\omega t} dt$$

В задании 3 используется преобразование Фурье к обыкновенной частоте ν . В общем виде формула имеет вид

$$\hat{f}(\nu) = \int_{-\infty}^{\infty} f(t) e^{-2\pi i \nu t} dt$$

Для проверки равенства Парсеваля используется формула ниже

$$\|f\|_2 = \|\hat{f}\|_2,$$

где $\|f\|_2$ – вторая норма заданной функции, $\|\hat{f}\|_2$ – вторая норма Фурье-образа функции f . Для нахождения нормы используются формулы, представленные ниже

$$\|f(t)\|_2 = \sqrt{\int_a^b f(t) \cdot f^*(t) dt}, \quad \|\hat{f}(\omega)\|_2 = \sqrt{\int_a^b \hat{f}(\omega) \cdot \hat{f}^*(\omega) d\omega}$$

Все графики строятся программой, написанной на языке программирования python. В 1 и 2 заданиях используется библиотека sympy, в задании 3 numpy и matplotlib. По ходу отчета приводится код для каждого задания. Для всех интегралов и графиков есть место с общими переменными и значениями – файл static.py. Основные используемые данные приведены ниже

```

1  from sympy import Symbol, Piecewise, Abs, sinc, E, oo
2
3  t = Symbol('t')
4  omega = Symbol('omega')
5
6  interval = [-oo, oo]
7
8  a_b_pars = [(1, 2), (2, 3), (3, 4)]
9  consts = [-1, 0.5, 1]
10 colors_strs = ['red', 'purple', 'blue', 'cyan']

```

Листинг 1: Основные данные из файла static.py

В этом файле программно заданы функции, графики которых приводятся по ходу отчета. Также они необходимы для нахождения их Фурье-образа

```

1  def rectangular_function(a, b):
2      return Piecewise((a, Abs(t) <= b), (0, Abs(t) > b))
3
4  def triangular_function(a, b):
5      return Piecewise((a - Abs(a * t / b), Abs(t) <= b), (0, Abs(t) > b))
6
7  def cardinal_sinus(a, b):
8      return a * sinc(b * t)
9
10 def gaussian_function(a, b):
11     return a * E ** (-b * t ** 2)
12
13 def double_attenuation(a, b):
14     return a * E ** (-b * Abs(t))

```

Листинг 2: Программно заданные функции для заданий 1 и 2

2 Задание 1. Вещественное

2.1 Прямоугольная функция

Рассмотрим прямоугольную функцию следующего вида

$$f(t) = \begin{cases} a, & |t| \leq b, \\ 0, & |t| > b. \end{cases}$$

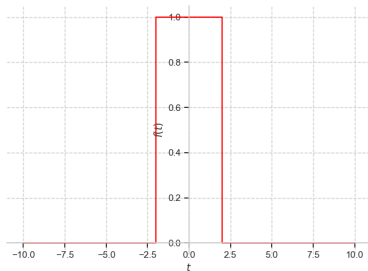
Рассмотрим программу. Сначала задается функция, принимающая параметры a и b , после чего методом `build_f_t` строится график $f(t)$. На строке 10 приведен пример использования кода

```

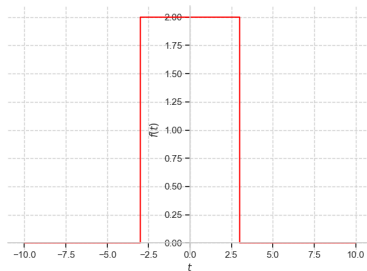
1 def build_f_t(f_t, clr, lbl):
2     if (lbl == None):
3         plot(f_t, line_color=clr, xlabel=r'$t$', ylabel=r'$f(t)$')
4     else:
5         plot(f_t, line_color=clr, xlabel=r'$t$', ylabel=r'$f(t)$', label=lbl, legend=True)
6
7 build_f_t(rectangular_function(1, 2), 'red', None)
```

Листинг 3: Программа для построения графика прямоугольной функции

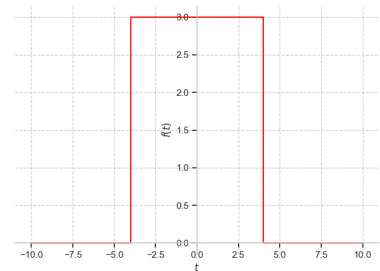
Построенные графики $f(t)$ для нескольких значений параметров $a, b > 0$ расположены ниже



(a) $a = 1, b = 2$



(b) $a = 2, b = 3$



(c) $a = 3, b = 4$

Рис. 1: Прямоугольные функции при различных значениях a и b

Рассмотрим программу. Методом `find_fimg` находится Фурье-образ заданной функции. Далее методом `build_fimg2` строится график Фурье-образа. На 16-17 строчках находится пример использования кода

```

1 def find_fimg(f_t, lim1, lim2):
2     integrand = f_t * E ** (-I * omega * t)
3
4     result = integrate(integrand, (t, lim1, lim2))
5     return coeff * result
6
7 def build_fimg2(fimg, clr, lbl):
8     if (lbl == None):
9         plot(fimg, line_color=clr,
10             xlabel=r'$\omega$', ylabel=r'$c(\omega)$')
11     else:
12         plot(fimg, line_color=clr,
13             xlabel=r'$\omega$', ylabel=r'$c(\omega)$',
14             label=lbl, legend=True)
15
16 rectfimg = find_fimg(rectangular_function, -oo, oo)
17 build_fimg2(rectfimg, 'purple', None)
```

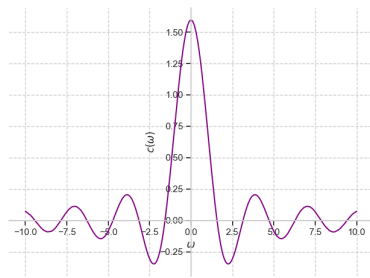
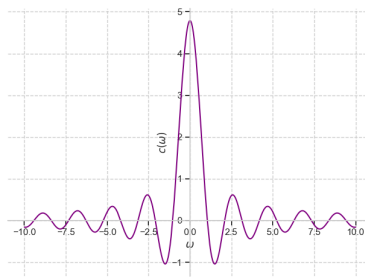
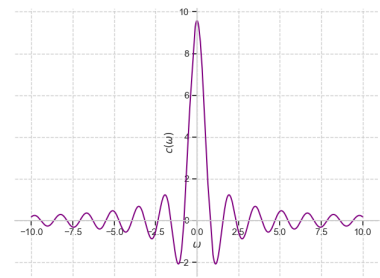
Листинг 4: Программа для построения графика Фурье-образа некоторой функции $f(t)$

Построенные графики $\hat{f}(\omega)$ для тех же значений a и b расположены ниже

Рассмотрим программу. Методом `find_norm2` находится вторая норма переданной функции. Метод `find_parseval` считает левую и правую части равенства Парсеваля. Пример использования кода расположен на 13-14 строчках листинга ниже

```

1 def find_norm2(f, lim1, lim2, var):
2     integrand = f * conjugate(f)
```

(a) $a = 1, b = 2$ (b) $a = 2, b = 3$ (c) $a = 3, b = 4$ Рис. 2: Фурье-образы прямоугольных функций при различных значениях a и b

```

3      result = integrate(integrand, (var, lim1, lim2)).evalf()
4      return sqrt(result).evalf()
5
6
7      def find_parseval(f, fimg, lim1, lim2):
8          pleft = find_norm2(f, lim1, lim2, t)
9          pright = find_norm2(fimg, lim1, lim2, omega)
10
11         return pleft, pright
12
13     pl, pr = find_parseval(rectangular_function, rectfimg, -oo, oo)
14     print(f'p_{{1}}: {{p1}} ?= {{pr}}')
```

Листинг 5: Программа для нахождения левой и правой сторон равенства Парсеваля

Программа вывела в консоль результаты, представленные ниже

```

1  p_1: 2.0000000000000000 ?= 2.0 + 0.e-114*I
2  p_2: 4.89897948556636 ?= 4.89897948556636 + 0.e-114*I
3  p_3: 8.48528137423857 ?= 8.48528137423857 + 0.e-114*I
```

Листинг 6: Результат выполнения программы для вычисления равенства Парсеваля

Мнимыми частями в правой части равенства Парсеваля пренебрежем вследствие их стремления к нулю. В таком случае равенство Парсеваля выполняется, что можно объяснить тем, что интеграл позволяет рассмотреть норму непрерывно на заданном промежутке, а ряд только дискретно, вследствие чего теряются какие-то члены ряда, которых не хватает для выполнения равенства Парсеваля

2.2 Рассмотрим треугольную функцию следующего вида

$$f(t) = \begin{cases} a - \left| \frac{at}{b} \right|, & |t| \leq b, \\ 0, & |t| > b. \end{cases}$$

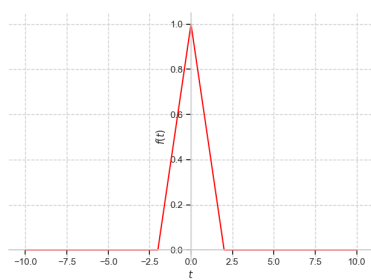
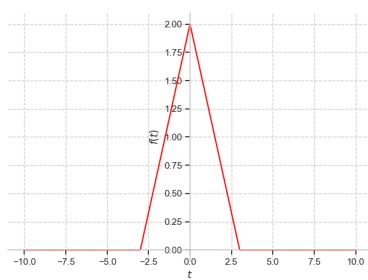
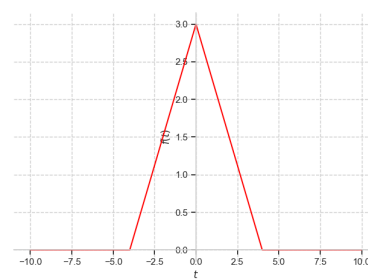
Для построения графиков треугольной функции сначала задается необходимая функция, после используется код, приведенный в пункте 2.1 для прямоугольной функции

```

1  null
```

Листинг 7: Программно заданная треугольная функция

Построенные графики $f(t)$ для нескольких значений параметров $a, b > 0$ расположены ниже

(a) $a = 1, b = 2$ (b) $a = 2, b = 3$ (c) $a = 3, b = 4$ Рис. 3: Треугольные функции при различных значениях a и b