МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное
образовательное учреждение высшего образования
«Самарский национальный исследовательский университет
имени академика С.П. Королева»
(Самарский университет)

Институт информатики и кибернетики
Кафедра технической кибернетики

Отчёт по лабораторной работе №2
Дисциплина: «Оптическая информатика»

Численная реализация оптического преобразования Фурье на основе быстрого преобразования Фурье

Выполнил: Гуторов Владислав
Сергеевич
Группа: 6402-010302D
Вариант 3

Самара 2024

Задание

1. Реализовать одномерное финитное преобразование Фурье с помощью применения алгоритма БПФ.
2. Построить график гауссова пучка e^{-x^2}. Здесь и далее для каждого графика следует строить отдельно графики амплитуды и фазы.
3. Убедиться в правильности реализации преобразования, подав на вход гауссов пучок e^{-x^2} – собственную функцию преобразования Фурье. На выходе тоже должен получиться гауссов пучок (построить график на правильной области определения [-b,b]). Рекомендуемая входная область: [-a,a]=[-5, 5].
4. Реализовать финитное преобразование Фурье стандартным методом численного интегрирования (например, методом прямоугольников). Важно: необходимо вычислить интеграл для каждого дискретного значения u, чтобы получить результат в виде вектора. На вход преобразования вновь следует подавать гауссов пучок.
5. Построить результаты двух разных реализаций преобразования на одном изображении (одно для амплитуды, одно для фазы) и убедиться, что они совпадают.
6. Используя первую реализацию преобразования, подать на вход световое поле, отличное от гауссова пучка, в соответствии со своим вариантом (таблица 1). Построить графики самого пучка и результата преобразования.
7. Рассчитать аналитически результат преобразования своего варианта поля и построить график на одной системе координат с результатом, полученным в предыдущем пункте.
8. Исследовать параметры N и M алгоритма БПФ. Для этого сначала необходимо варьировать N, задавая при этом $M = N$. Что происходит при увеличении и уменьшении N? После этого следует зафиксировать N и изменять $M > N$. Что происходит при увеличении M? Выводы следует поместить в таблицу 2. Подкрепить выводы графиками. Важно: число M должно оставаться степенью двойки, чтобы график фазы содержал минимальное число ошибок.
9. Выполнить пункты 1-3 и 6 для двумерного случая. Графики изменятся на двумерные изображения, одномерные функции следует заменить на двумерные, равные произведению

соответствующих одномерных функций. Например, гауссов пучок поменяется на $e^{-x^2-y^2}$. Изображение следует строить не в виде 3D-графиков, а в виде цветовой схемы (см. примеры двумерных пучков в лекциях). Рассчитать аналитически результат двумерного преобразования и нарисовать изображения его амплитуды и фазы.

	Входное поле	Примечание
3	$\exp(2\pi i x) + \exp(-5\pi i x)$	Для аналитики применить финитное преобразование.

Результаты работы

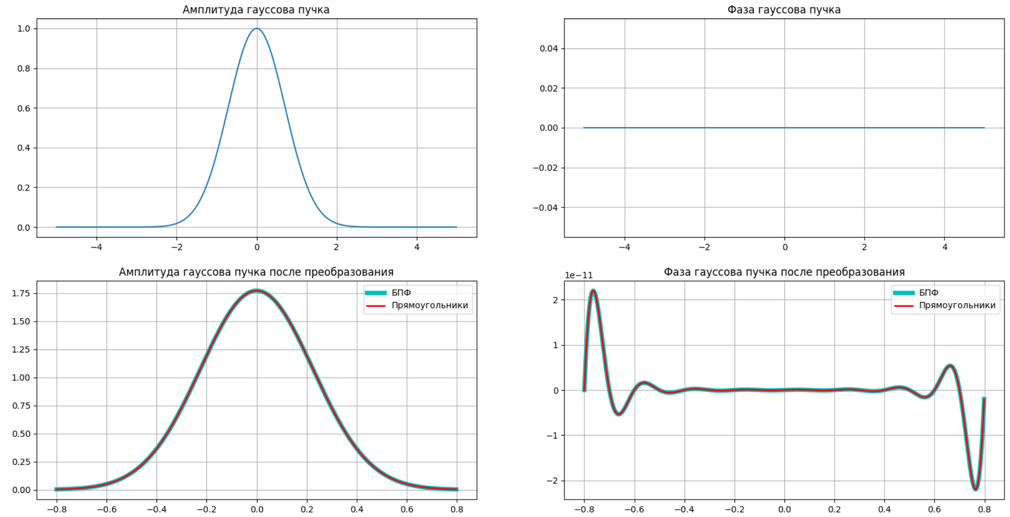

Рисунок 1 – Амплитуда и фаза гауссова пучка изначально и после БПФ библиотечным и численным методами

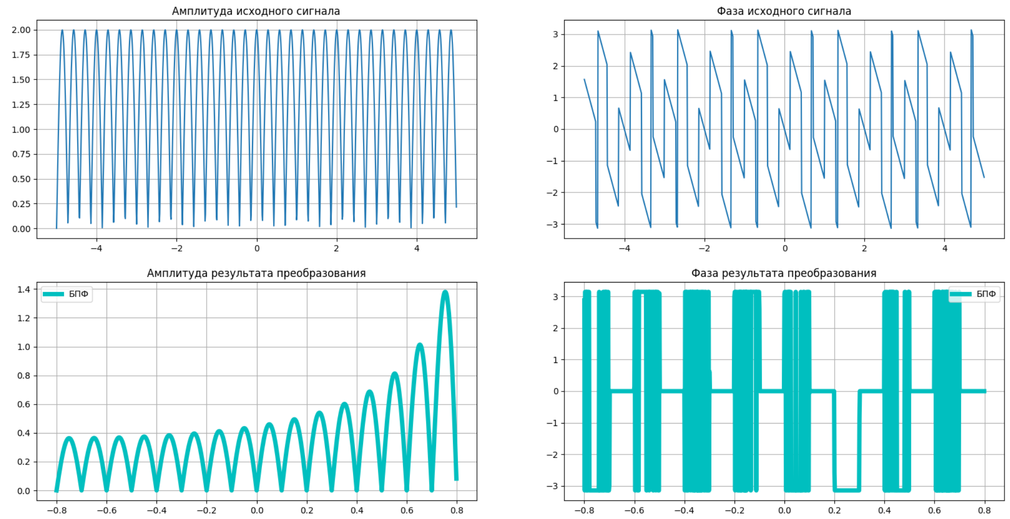

Рисунок 2 – Амплитуда и фаза исходного поля согласно варианту ия после БПФ

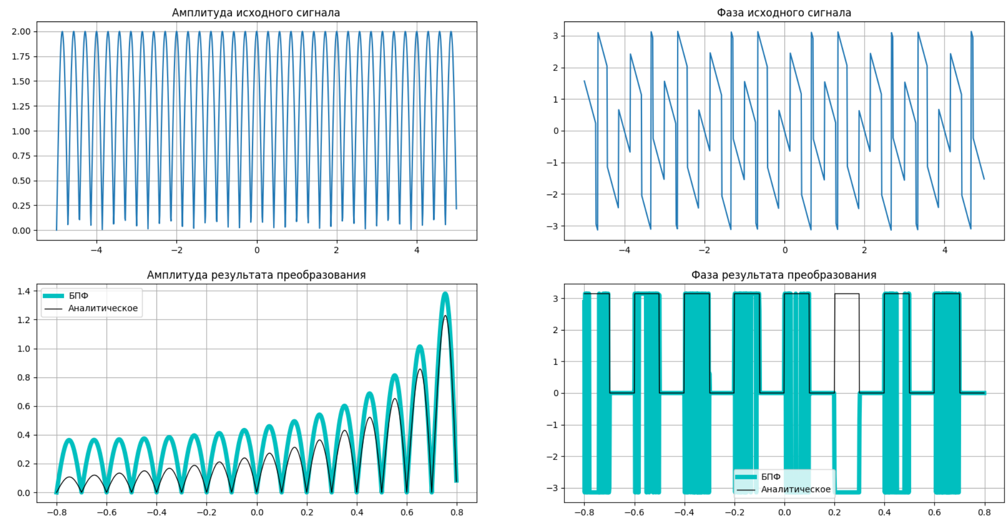

Рисунок 3 – Амплитуда и фаза светового поля в соответствии с вариантом, сравнение амплитуды и фазы с помощью БПФ и аналитического преобразования Фурье

Расчет аналитически:

Заданная функция:

$$f(x) = e^{2\pi i x} + e^{-5\pi i x}$$

Определение конечного преобразования Фурье:

$$F(u) = \int_{-a}^{a} f(x) e^{-i2\pi x u}\, dx.$$

Подставляя $f(x)$:

$$F(u) = \int_{-a}^{a} \left(e^{2\pi i x} + e^{-5\pi i x}\right) e^{-i2\pi x u}\, dx.$$

Распишем интеграл на два слагаемых:

$$F(u) = \int_{-a}^{a} e^{i2\pi x(1-u)}\, dx + \int_{-a}^{a} e^{i2\pi x(-5-u)}\, dx.$$

Обозначим:

$$k_1 = 2\pi(1-u), \quad k_2 = 2\pi(-5-u).$$

Известно, что:

$$\int_{-a}^{a} e^{ikx}\, dx = \frac{2\sin(ka)}{k}.$$

Применим это к каждому интегралу.

1. Для первого интеграла:

$$\int_{-a}^{a} e^{i2\pi(1-u)x}\,dx = \frac{2\sin\big(2\pi(1-u)a\big)}{2\pi(1-u)} = \frac{\sin\big(2\pi(1-u)a\big)}{\pi(1-u)}.$$

2. Для второго интеграла:

$$\int_{-a}^{a} e^{i2\pi(-5-u)x}\,dx = \frac{2\sin\big(2\pi(-5-u)a\big)}{2\pi(-5-u)} = \frac{\sin\big(2\pi(-5-u)a\big)}{\pi(-5-u)}.$$

Таким образом:

$$F(u) = \frac{\sin\big(2\pi(1-u)a\big)}{\pi(1-u)} + \frac{\sin\big(2\pi(-5-u)a\big)}{\pi(-5-u)}.$$

Особые точки:

- При $u = 1$: знаменатель $\pi(1-u) \to 0$. В этом случае по правилу Лопиталя $\lim_{u\to 1} \frac{\sin(2\pi(1-u)a)}{\pi(1-u)} = 2a$.
- При $u = -5$: аналогично, $\lim_{u\to -5} \frac{\sin(2\pi(-5-u)a)}{\pi(-5-u)} = 2a$.

Таким образом, при $u = 1$ и $u = -5$ значение соответствующего слагаемого нужно брать как $2a$.

Итоговая формула:

$$F(u) = \frac{\sin\big(2\pi(1-u)a\big)}{\pi(1-u)} + \frac{\sin\big(2\pi(-5-u)a\big)}{\pi(-5-u)}.$$

С учетом особых точек:

$$F(u) = \begin{cases} 2a + \frac{\sin\left(2\pi(-6)a\right)}{\pi(-6)}, & u = 1 \\ \frac{\sin\left(2\pi(1-u)a\right)}{\pi(1-u)} + 2a, & u = -5 \\ \frac{\sin\left(2\pi(1-u)a\right)}{\pi(1-u)} + \frac{\sin\left(2\pi(-5-u)a\right)}{\pi(-5-u)}, & \text{иначе} \end{cases}$$

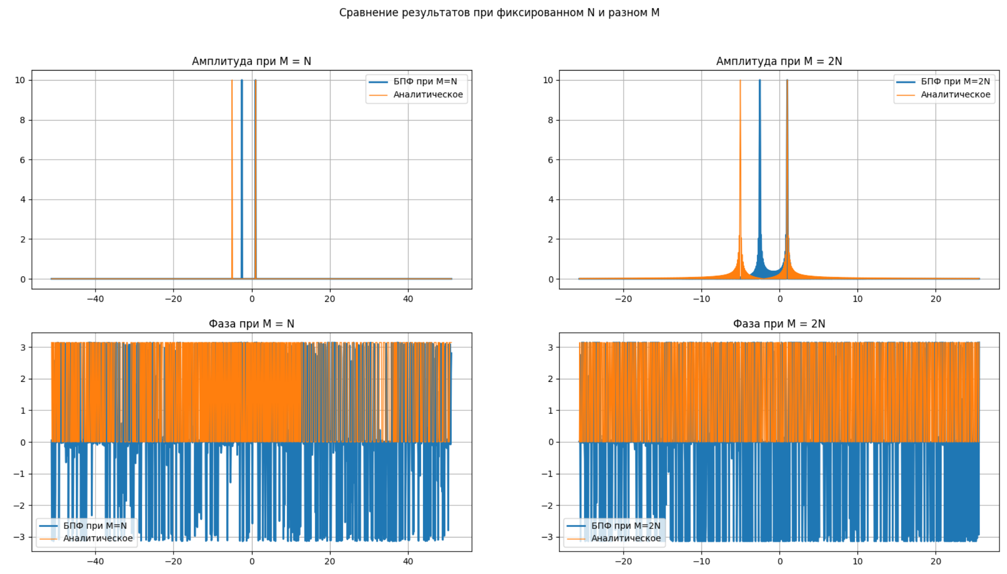

Рисунок 4 – Сравнение результатов при фиксированном N и разных M

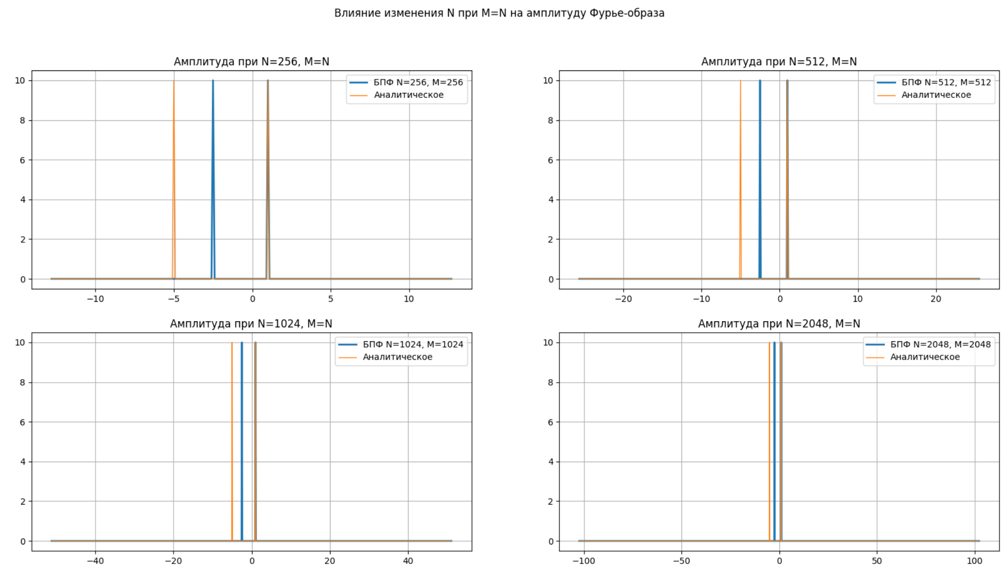

Рисунок 5 – Сравнение результатов при фиксированном N = M

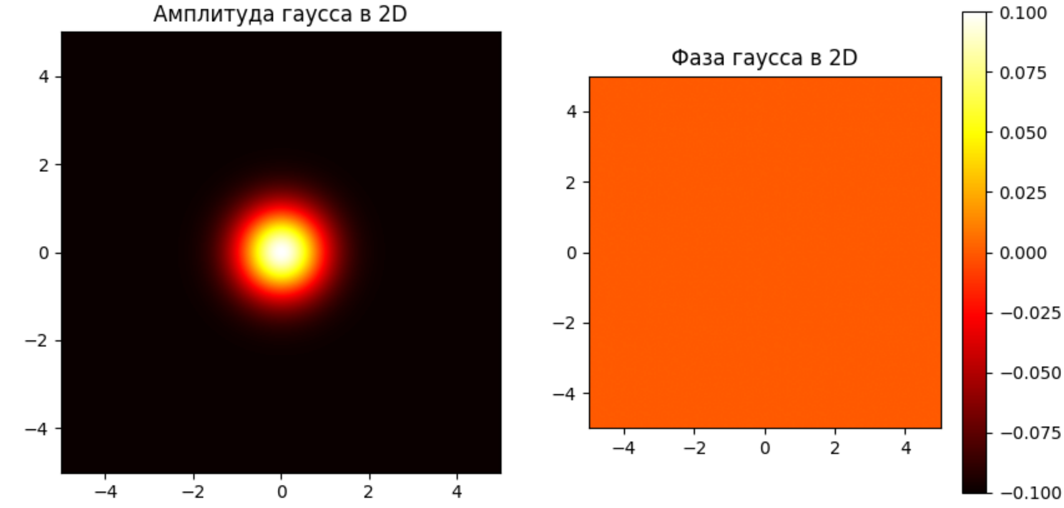

Рисунок 6 – Амплитуда и фаза двухмерного гауссова пучка

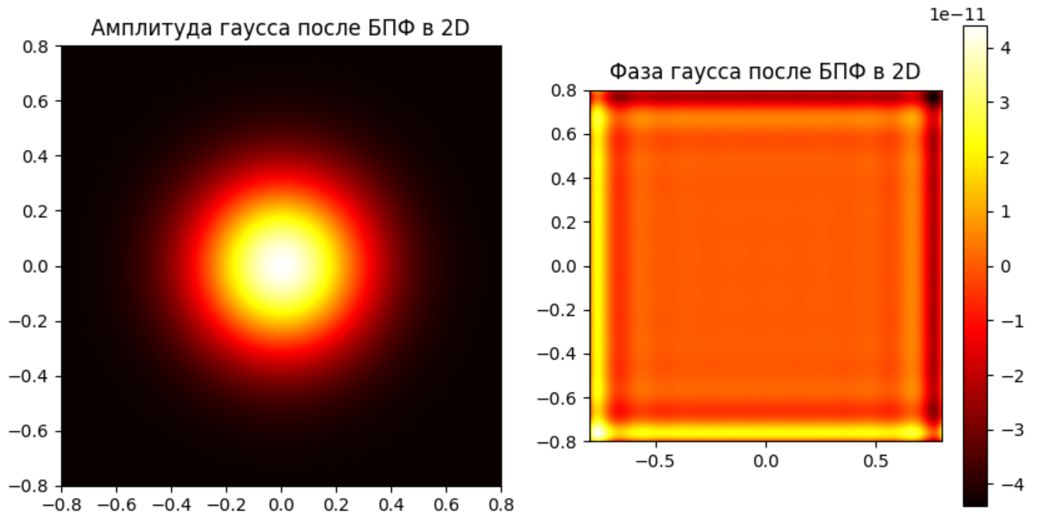

Рисунок 7 – Амплитуда и фаза гауссова пучка после БПФ

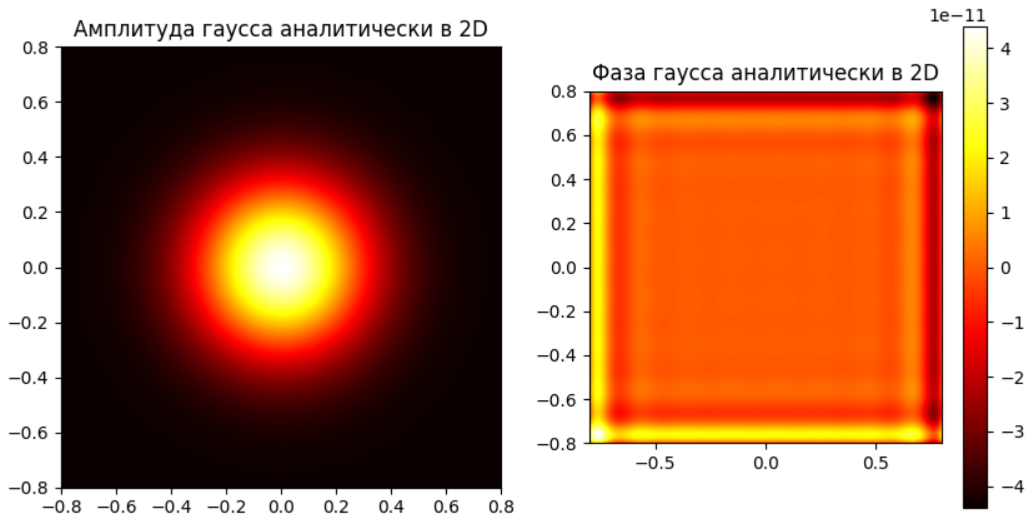

Рисунок 8 – Амплитуда и фаза гауссова пучка после аналитического вычисления

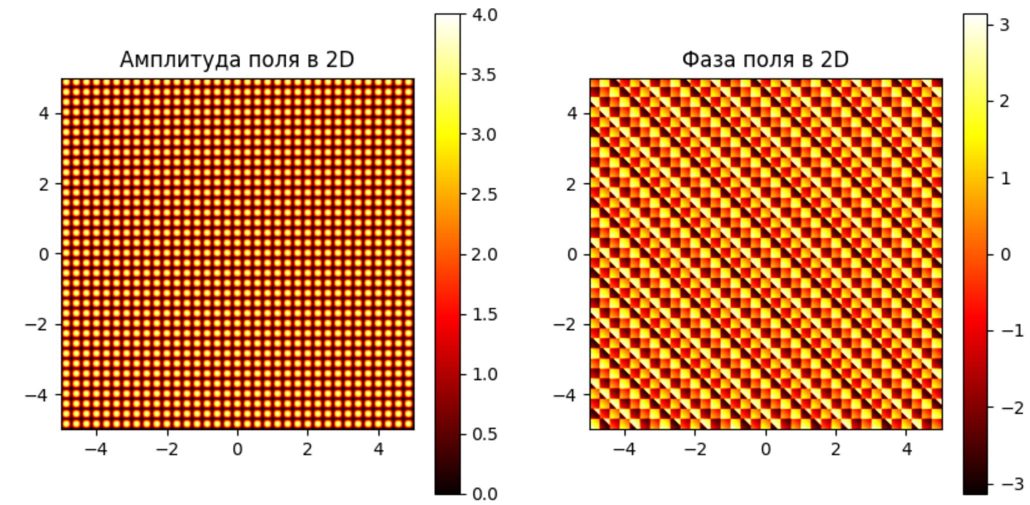

Рисунок 9 – Амплитуда и фаза светового поля, согласно варианту

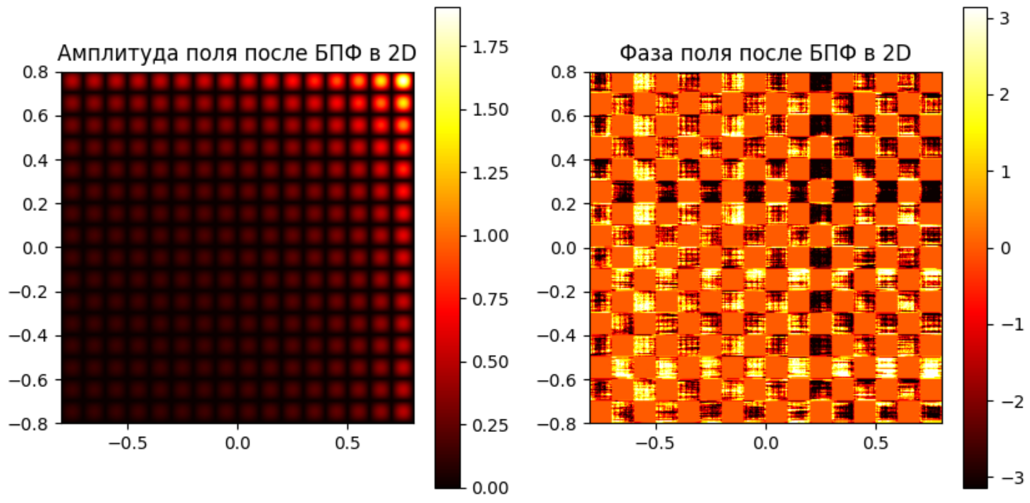

Рисунок 10 – Амплитуда и фаза светового поля, согласно варианту, после БПФ

Выводы

N	При увеличении N (при M=N) мы получаем более точное и гладкое приближение аналитического результата, так как увеличивается разрешение по пространству и, следовательно, уменьшается ошибка дискретизации.
M	При увеличении M (при фиксированном N) в частотной области становится более плотной дискретизация, что приводит к более плавному графику частотного спектра.

Код

```
import numpy as np
import matplotlib.pyplot as plt

# ------------------------- Исходные параметры -------------------------
a1 = -5.0
a2 = 5.0
a = a2  # Половина интервала
N = 1 << 10  # Количество точек по x
M = 1 << 16  # Количество точек для нулевого дополнения при БПФ
x = np.linspace(a1, a2, N, endpoint=False)
hx = (a2 - a1) / N

# Частотная область
# Определение частотного шага и диапазона
b2 = (N * N) / (4 * a2 * M)
b1 = -b2
```

```python
u = np.linspace(b1, b2, N, endpoint=False)  # Массив частот


# --------------------------- Функция ---------------------------
# f(x) = e^{2 π i x} + e^{-5 π i x}
def f_variant(x):
    return np.exp(2j * np.pi * x) + np.exp(-5j * np.pi * x)

f = f_variant(x)

def f_analytic(u, a=5):
    # Вычисляем F(u) аналитически
    # Аккуратно обрабатываем особые точки
    def val(u):
        part1_num = np.sin(2 * np.pi * (1 - u) * a)
        part1_den = np.pi * (1 - u)
        part2_num = np.sin(2 * np.pi * (-5 - u) * a)
        part2_den = np.pi * (-5 - u)

        # Проверка особых точек
        # u близко к 1?
        if np.isclose(u, 1.0, atol=1e-14):
            part1 = 2 * a  # Предел при u->1
        else:
            part1 = part1_num / part1_den

        # u близко к -5?
        if np.isclose(u, -5.0, atol=1e-14):
            part2 = 2 * a  # Предел при u->-5
        else:
            part2 = part2_num / part2_den

        return part1 + part2

    # Применяем поэлементно
    return np.array([val(ui) for ui in u], dtype=complex)


# --------------------------- Численное преобразование Фурье методом
прямоугольников ---------------------------
# Используем прямое определение интеграла:
# F(u) ≈ Σ f(x_j) * e^{-i2πx_j u} * hx, j=0..N-1
# Здесь для каждого u из массива u вычисляем интеграл.

def numeric_fourier_transform_rect(f_vals, x_vals, u_vals):
    F_vals = np.zeros_like(u_vals, dtype=complex)
    for i_u, uu in enumerate(u_vals):
        # e^{-i2πx_j u}
        integrand = f_vals * np.exp(-2j * np.pi * x_vals * uu)
        F_vals[i_u] = np.sum(integrand) * hx
    return F_vals


# --------------------------- Быстрое преобразование Фурье (БПФ) -------------
-------------
# Реализация с нулевым дополнением и учетом hx.
def bpf(f_vals, N, M, hx):
    k = (M - N) // 2
    F = np.insert(f_vals, 0, np.zeros(k))
    F = np.append(F, np.zeros(k))
    F = np.fft.fftshift(F)
    F = np.fft.fft(F) * hx
    F = np.fft.fftshift(F)
    F = F[len(F) // 2 - N // 2: len(F) // 2 + N // 2]
```

```python
    return F

# --------------------------- Гауссов пучок для проверки реализации ----------
-----------------
gauss = lambda x: np.exp(-x ** 2)
f_gauss = gauss(x)

F_gauss_fft = bpf(f_gauss, N, M, hx)  # БПФ для гаусса

# Аналитическое преобразование гаусса известное:
# Преобразование Фурье от e^{-x^2} с нормировкой 2π даёт также гауссову
форму.
# Для конечного отрезка приближение будет близко к гауссу. Для сравнения
просто используем численный интеграл:
F_gauss_rect = numeric_fourier_transform_rect(f_gauss, x, u)

plt.figure(figsize=(20, 10))
plt.subplot(2, 2, 1)
plt.plot(x, np.abs(f_gauss))
plt.title('Амплитуда гауссова пучка')
plt.grid()

plt.subplot(2, 2, 2)
plt.plot(x, np.angle(f_gauss))
plt.title('Фаза гауссова пучка')
plt.grid()

plt.subplot(2, 2, 3)
plt.plot(u, np.abs(F_gauss_fft), label='БПФ', c='c', lw=5)
plt.plot(u, np.abs(F_gauss_rect), label='Прямоугольники', c='r', lw=2)
plt.title('Амплитуда гауссова пучка после преобразования')
plt.legend()
plt.grid()

plt.subplot(2, 2, 4)
plt.plot(u, np.angle(F_gauss_fft), label='БПФ', c='c', lw=5)
plt.plot(u, np.angle(F_gauss_rect), label='Прямоугольники', c='r', lw=2)
plt.title('Фаза гауссова пучка после преобразования')
plt.legend()
plt.grid()

plt.show()

plt.figure(figsize=(20, 10))
plt.subplot(2, 2, 1)
plt.plot(x, np.abs(f_gauss))
plt.title('Амплитуда гауссова пучка')
plt.grid()

plt.subplot(2, 2, 2)
plt.plot(x, np.angle(f_gauss))
plt.title('Фаза гауссова пучка')
plt.grid()

plt.subplot(2, 2, 3)
plt.plot(u, np.abs(F_gauss_fft), label='БПФ', c='c', lw=5)
plt.plot(u, np.abs(F_gauss_rect), label='Прямоугольники', c='r', lw=2)
plt.title('Амплитуда гауссова пучка после преобразования')
plt.legend()
plt.grid()

plt.subplot(2, 2, 4)
plt.plot(u, np.angle(F_gauss_fft), label='БПФ', c='c', lw=5)
plt.plot(u, np.angle(F_gauss_rect), label='Прямоугольники', c='r', lw=2)
```

```python
plt.title('Фаза гауссова пучка после преобразования')
plt.legend()
plt.grid()

plt.show()

plt.figure(figsize=(20, 10))
plt.subplot(2, 2, 1)
plt.plot(x, np.abs(f_gauss))
plt.title('Амплитуда гауссова пучка')
plt.grid()

plt.subplot(2, 2, 2)
plt.plot(x, np.angle(f_gauss))
plt.title('Фаза гауссова пучка')
plt.grid()

plt.subplot(2, 2, 3)
plt.plot(u, np.abs(F_gauss_fft), label='БПФ', c='c', lw=5)
plt.plot(u, np.abs(F_gauss_rect), label='Прямоугольники', c='r', lw=2)
plt.title('Амплитуда гауссова пучка после преобразования')
plt.legend()
plt.grid()

plt.subplot(2, 2, 4)
plt.plot(u, np.angle(F_gauss_fft), label='БПФ', c='c', lw=5)
plt.plot(u, np.angle(F_gauss_rect), label='Прямоугольники', c='r', lw=2)
plt.title('Фаза гауссова пучка после преобразования')
plt.legend()
plt.grid()

plt.show()
# -------------------------- Вычисления--------------------------

F_fft = bpf(f, N, M, hx)  # Численный метод (БПФ)
F_an = f_analytic(u)  # Аналитический результат

# -------------------------- Построение графиков--------------------------
plt.figure(figsize=(20, 10))
plt.subplot(2, 2, 1)
plt.plot(x, np.abs(f))
plt.title('Амплитуда исходного сигнала')
plt.grid()

plt.subplot(2, 2, 2)
plt.plot(x, np.angle(f))
plt.title('Фаза исходного сигнала')
plt.grid()

plt.subplot(2, 2, 3)
plt.plot(u, np.abs(F_fft), label='БПФ', c='c', lw=5)
plt.plot(u, np.abs(F_an), label='Аналитическое', c='k', lw=1)
plt.title('Амплитуда результата преобразования')
plt.legend()
plt.grid()

plt.subplot(2, 2, 4)
plt.plot(u, np.angle(F_fft), label='БПФ', c='c', lw=5)
plt.plot(u, np.angle(F_an), label='Аналитическое', c='k', lw=1)
plt.title('Фаза результата преобразования')
plt.legend()
plt.grid()

plt.show()
```

```
# -------------------------- Двумерный случай --------------------------
# Определим двумерную функцию для варианта:
# f_2d(x,y) = f_variant(x)*f_variant(y), но в варианте требуется именно отличное поле.
# Можно использовать заданное f_variant(x) для каждой координаты:
X, Y = np.meshgrid(x, x)
f_2d = f_variant(X) * f_variant(Y)  # двумерный аналог

fig, arr = plt.subplots(1, 2, figsize=(10, 5))
amp = arr[0].imshow(np.abs(f_2d), cmap='hot', extent=[a1, a2, a1, a2], origin='lower')
arr[0].set_title('Амплитуда поля в 2D')
fig.colorbar(amp, ax=arr[0])
phase = arr[1].imshow(np.angle(f_2d), cmap='hot', extent=[a1, a2, a1, a2], origin='lower')
arr[1].set_title('Фаза поля в 2D')
fig.colorbar(phase, ax=arr[1])
plt.show()


# Двумерное БПФ: сначала по строкам, потом по столбцам
def bpf_2d(Z, a, b, N, M):
    # Применяем бпф по столбцам
    for i in range(N):
        h = (b - a) / N
        Z[:, i] = bpf(Z[:, i], N, M, h)
    # Применяем бпф по строкам
    for i in range(N):
        h = (b - a) / N
        Z[i, :] = bpf(Z[i, :], N, M, h)
    return Z


F_2d = f_2d.astype(np.complex128)
F_2d = bpf_2d(F_2d, a1, a2, N, M)

fig, arr = plt.subplots(1, 2, figsize=(10, 5))
amp = arr[0].imshow(np.abs(F_2d), cmap='hot', extent=[b1, b2, b1, b2], origin='lower')
arr[0].set_title('Амплитуда поля после БПФ в 2D')
fig.colorbar(amp, ax=arr[0])
phase = arr[1].imshow(np.angle(F_2d), cmap='hot', extent=[b1, b2, b1, b2], origin='lower')
arr[1].set_title('Фаза поля после БПФ в 2D')
fig.colorbar(phase, ax=arr[1])
plt.show()

# Аналитическое двухмерное решение можно получить как произведение двух одномерных решений:
F_an_2d = np.outer(F_an, F_an)

fig, arr = plt.subplots(1, 2, figsize=(10, 5))
amp = arr[0].imshow(np.abs(F_an_2d), cmap='hot', extent=[b1, b2, b1, b2], origin='lower')
arr[0].set_title('Амплитуда аналитически в 2D')
fig.colorbar(amp, ax=arr[0])
phase = arr[1].imshow(np.angle(F_an_2d), cmap='hot', extent=[b1, b2, b1, b2], origin='lower')
arr[1].set_title('Фаза аналитически в 2D')
fig.colorbar(phase, ax=arr[1])
plt.show()

# Аналогично можно выполнить для двумерного гаусса:
```

```
gauss_2d = lambda x, y: np.exp(-x ** 2 - y ** 2)
f_gauss_2d = gauss_2d(X, Y)

fig, arr = plt.subplots(1, 2, figsize=(10, 5))
amp = arr[0].imshow(np.abs(f_gauss_2d), cmap='hot', extent=[a1, a2, a1, a2],
origin='lower')
arr[0].set_title('Амплитуда гаусса в 2D')
phase = arr[1].imshow(np.angle(f_gauss_2d), cmap='hot', extent=[a1, a2, a1,
a2], origin='lower')
arr[1].set_title('Фаза гаусса в 2D')
fig.colorbar(phase, ax=arr[1])
plt.show()

F_gauss_2d = f_gauss_2d.astype(np.complex128)
F_gauss_2d = bpf_2d(F_gauss_2d, a1, a2, N, M)

fig, arr = plt.subplots(1, 2, figsize=(10, 5))
amp = arr[0].imshow(np.abs(F_gauss_2d), cmap='hot', extent=[b1, b2, b1, b2],
origin='lower')
arr[0].set_title('Амплитуда гаусса после БПФ в 2D')
phase = arr[1].imshow(np.angle(F_gauss_2d), cmap='hot', extent=[b1, b2, b1,
b2], origin='lower')
arr[1].set_title('Фаза гаусса после БПФ в 2D')
fig.colorbar(phase, ax=arr[1])
plt.show()

# Аналитический 2D для гаусса - произведение одномерных результатов:
# Для гаусса аналитически: Фурье также гаусс. Для упрощения возьмём результат
численно рассчитанный одномерно:
F_gauss_1d = numeric_fourier_transform_rect(f_gauss, x, u)
F_gauss_2d_an = np.outer(F_gauss_1d, F_gauss_1d)

fig, arr = plt.subplots(1, 2, figsize=(10, 5))
amp = arr[0].imshow(np.abs(F_gauss_2d_an), cmap='hot', extent=[b1, b2, b1,
b2], origin='lower')
arr[0].set_title('Амплитуда гаусса аналитически в 2D')
phase = arr[1].imshow(np.angle(F_gauss_2d_an), cmap='hot', extent=[b1, b2,
b1, b2], origin='lower')
arr[1].set_title('Фаза гаусса аналитически в 2D')
fig.colorbar(phase, ax=arr[1])
plt.show()

# ----------------------------------------------------
# Исследуем влияние N и M.

# Набор значений N для исследования (все степени двойки)
N_values = [256, 512, 1024, 2048]
fixed_N = 1024

# Функция, которая возвращает координатную сетку и частотную сетку в
зависимости от N, M
def get_grids(N, M, a1=-5, a2=5):
    x = np.linspace(a1, a2, N, endpoint=False)
    hx = (a2 - a1) / N
    b2 = (N * N) / (4 * a2 * M)
    b1 = -b2
    u = np.linspace(b1, b2, N, endpoint=False)
    return x, u, hx

# Исследуем влияние увеличения/уменьшения N при M = N
plt.figure(figsize=(20, 10))

for i, NN in enumerate(N_values, 1):
    MM = NN  # M = N
```

```
    x, u, hx = get_grids(NN, MM, a1, a2)
    f_vals = f_variant(x)
    F_bpf = bpf(f_vals, NN, MM, hx)
    F_an = f_analytic(u)

    plt.subplot(2, 2, i)
    plt.plot(u, np.abs(F_bpf), label=f'БПФ N={NN}, M={MM}', lw=2)
    plt.plot(u, np.abs(F_an), label='Аналитическое', lw=1)
    plt.title(f'Амплитуда при N={NN}, M=N')
    plt.legend()
    plt.grid()

plt.suptitle('Влияние изменения N при M=N на амплитуду Фурье-образа')
plt.show()

N_fixed = fixed_N
M_new = 2 * N_fixed
x_fixed, u_fixed, hx_fixed = get_grids(N_fixed, N_fixed, a1, a2)
x_newM, u_newM, hx_newM = get_grids(N_fixed, M_new, a1, a2)

f_fixed = f_variant(x_fixed)
f_newM = f_variant(x_newM) # хотя сетка x такая же, просто u_newM будет
другой

F_fixed = bpf(f_fixed, N_fixed, N_fixed, hx_fixed)
F_newM = bpf(f_newM, N_fixed, M_new, hx_newM)
F_an_fixed = f_analytic(u_fixed)
F_an_newM = f_analytic(u_newM)

plt.figure(figsize=(20, 10))

plt.subplot(2, 2, 1)
plt.plot(u_fixed, np.abs(F_fixed), label='БПФ при M=N', lw=2)
plt.plot(u_fixed, np.abs(F_an_fixed), label='Аналитическое', lw=1)
plt.title('Амплитуда при M = N')
plt.legend()
plt.grid()

plt.subplot(2, 2, 2)
plt.plot(u_newM, np.abs(F_newM), label='БПФ при M=2N', lw=2)
plt.plot(u_newM, np.abs(F_an_newM), label='Аналитическое', lw=1)
plt.title('Амплитуда при M = 2N')
plt.legend()
plt.grid()

plt.subplot(2, 2, 3)
plt.plot(u_fixed, np.angle(F_fixed), label='БПФ при M=N', lw=2)
plt.plot(u_fixed, np.angle(F_an_fixed), label='Аналитическое', lw=1)
plt.title('Фаза при M = N')
plt.legend()
plt.grid()

plt.subplot(2, 2, 4)
plt.plot(u_newM, np.angle(F_newM), label='БПФ при M=2N', lw=2)
plt.plot(u_newM, np.angle(F_an_newM), label='Аналитическое', lw=1)
plt.title('Фаза при M = 2N')
plt.legend()
plt.grid()

plt.suptitle('Сравнение результатов при фиксированном N и разном M')
plt.show()
```