Содержание

[Задание 2](#_Toc150895265)

[Результаты работы 4](#_Toc150895266)

[Код программы 18](#_Toc150895267)

# Задание

1. Реализовать одномерное финитное преобразование Фурье с помощью применения алгоритма БПФ.
2. Построить график гауссова пучка . Здесь и далее для каждого графика следует строить отдельно графики амплитуды и фазы.
3. Убедиться в правильности реализации преобразования, подав на вход гауссов пучок – собственную функцию преобразования Фурье. На выходе тоже должен получиться гауссов пучок (построить график на правильной области определения . Рекомендуемая входная область: .
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. 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. Аналитический расчёт должен быть помещён в отчёт. Прикрепление в отчёт фотографии или скана вывода формулы не является допустимым. В некоторых вариантах после преобразования могут появиться особые точки. Значения в них нужно вычислять отдельно, в том числе в коде программы.
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 1. 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 . 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. Двумерное преобразование следует реализовывать по предложенному выше алгоритму, а именно: применить одномерный алгоритм к строкам и столбцам матрицы. Если реализовывать алгоритм через двумерное БПФ, то добавление огромного числа нулей к матрице не будет оптимальным, сведёт на нет все преимущества БПФ, а потому будет поводом для снижения оценки за работу.

Вариант 15:



**Результаты работы**

**Задание 1-3:**

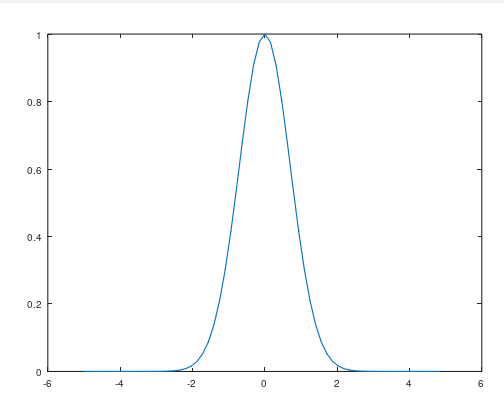


Рисунок 1 – График амплитуды гауссова пучка

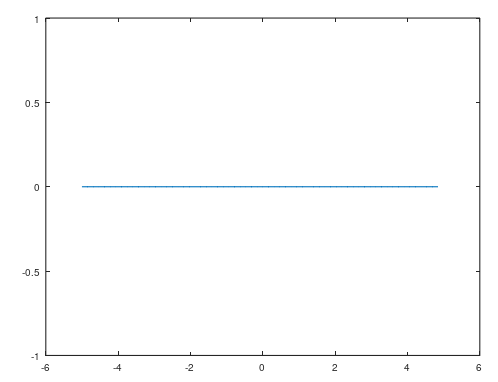


Рисунок 2 – График фазы гауссова пучка

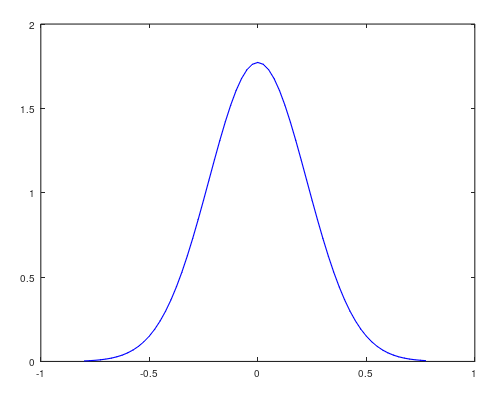
**

Рисунок 3 – График амплитуды БПФ гауссова пучка

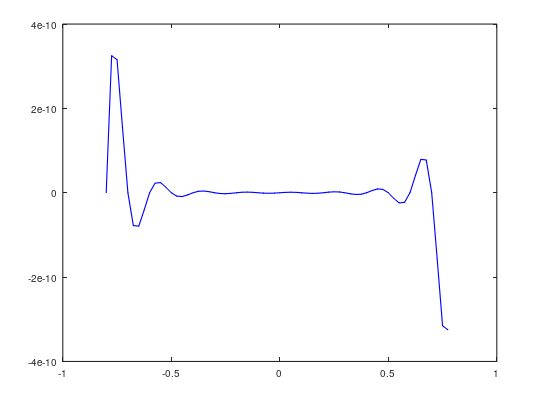


Рисунок 4 – График фазы БПФ гауссова пучка

**Задание 4-5:**

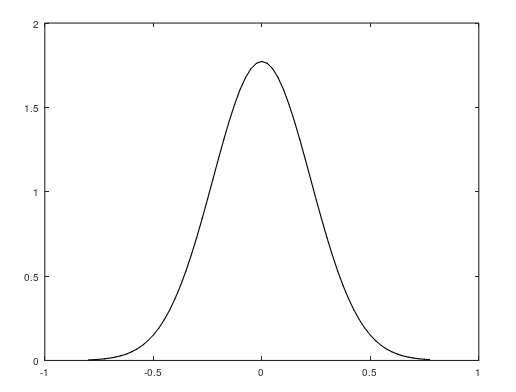


Рисунок 5 – График амплитуды преобразования стандартным методом численного интегрирования

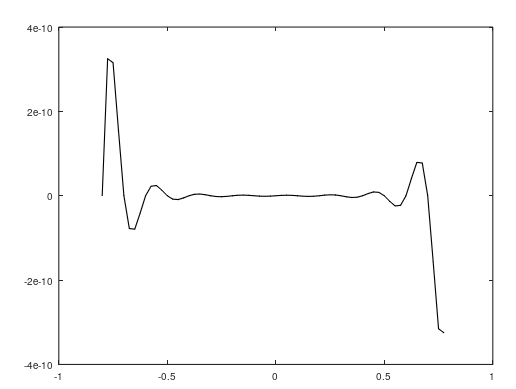


Рисунок 6 – График фазы преобразования стандартным методом численного интегрирования

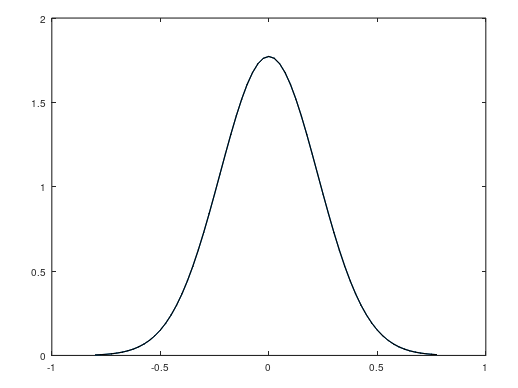


Рисунок 7 – Графики амплитуд результатов БПФ и преобразования стандартным методом численного интегрирования

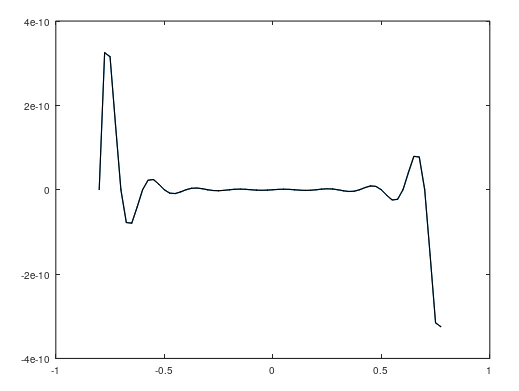
**

Рисунок 8 – Графики фаз результатов БПФ и преобразования стандартным методом численного интегрирования

Исходя из рисунков 7 и 8, где синим цветом изображены графики, полученные с помощью БПФ, а черным – с использованием стандартного метода численного интегрирования, можно сделать вывод, что результаты двух преобразований совпадают.

**Задание 6:**

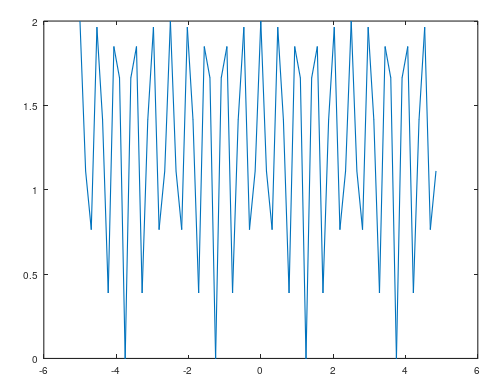


Рисунок 9 – График амплитуды светового поля

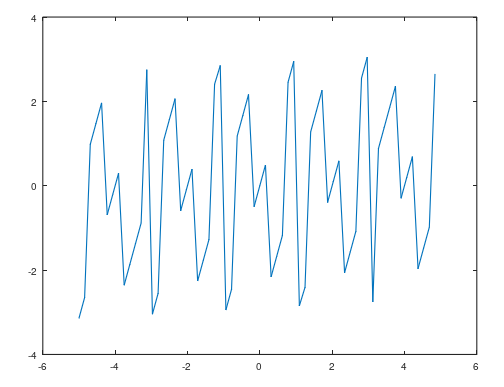


Рисунок 10 – График фазы светового поля

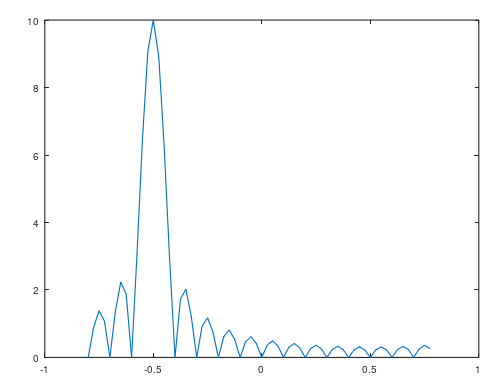


Рисунок 11 – График амплитуды БПФ светового поля

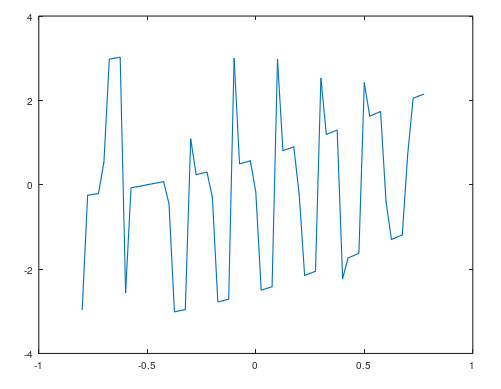


Рисунок 12 – График фазы БПФ светового поля

**Задание 7:**

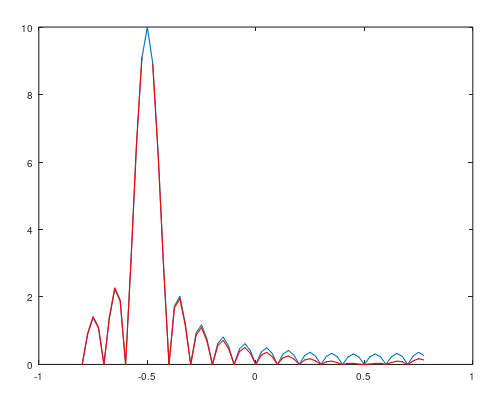


Рисунок 13 – Графики амплитуды результатов преобразований светового поля

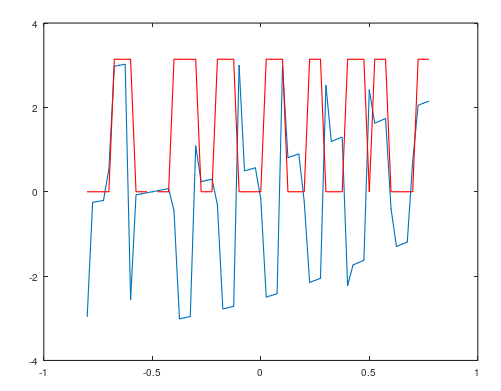


Рисунок 14 – Графики фазы результатов преобразований светового поля

На рисунках 13 и 14 представлены графики амплитуд и фаз соответственно, где синим цветом изображены графики, полученные с помощью БПФ, а красным с помощью аналитического решения.

**Задание 8:**

Сначала будем увеличивать и брать

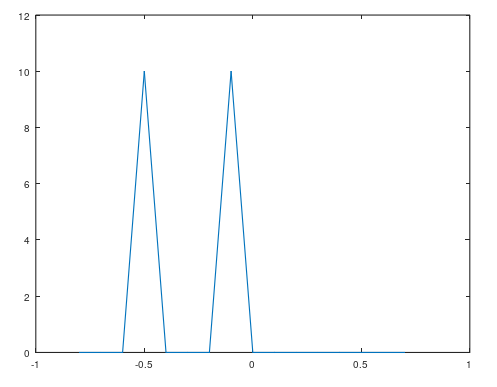


Рисунок 15 – График амплитуды БПФ светового поля при

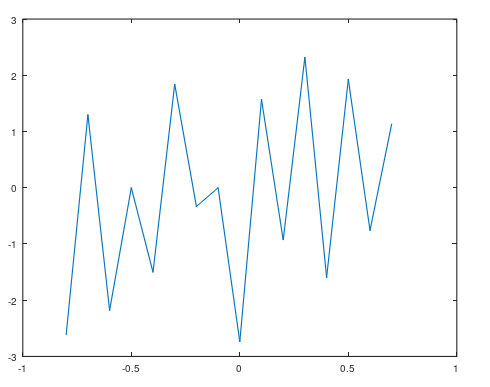


Рисунок 16 – График фазы БПФ светового поля при

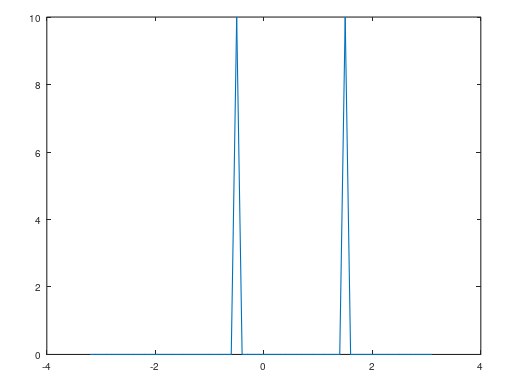


Рисунок 17 – График амплитуды БПФ светового поля при

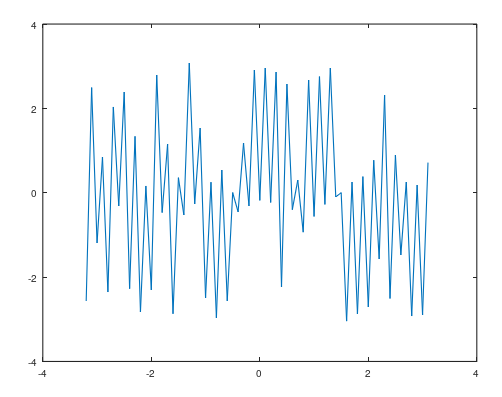


Рисунок 18 – График фазы БПФ светового поля при

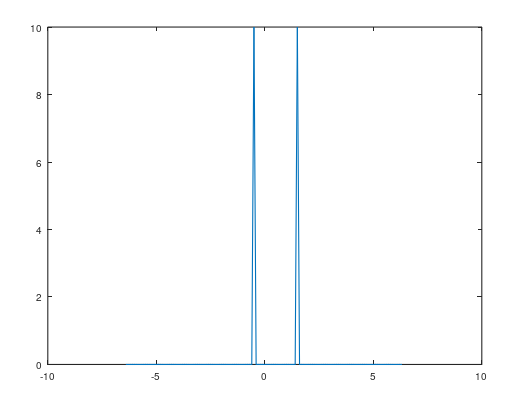


Рисунок 19 – График амплитуды БПФ светового поля при

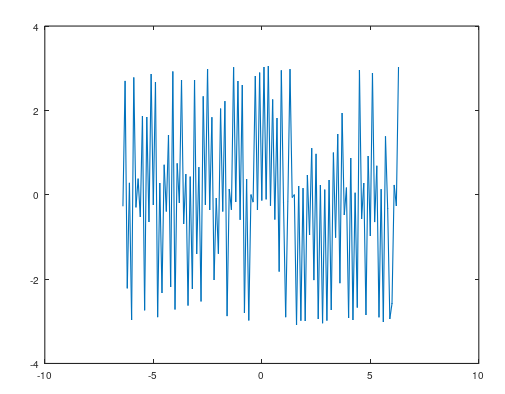


Рисунок 20 – График фазы БПФ светового поля при

Выводы по влиянию параметра представлены в таблице 1.

Теперь зафиксируем и будем увеличивать параметр .

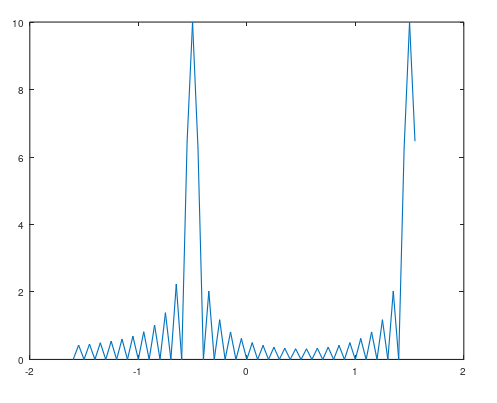


Рисунок 21 – График амплитуды БПФ светового поля при

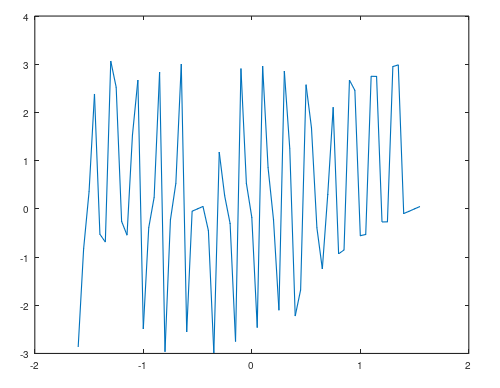


Рисунок 22 – График фазы БПФ светового поля при

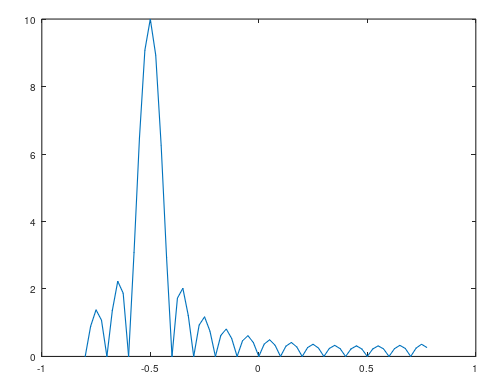


Рисунок 23 – График амплитуды БПФ светового поля при

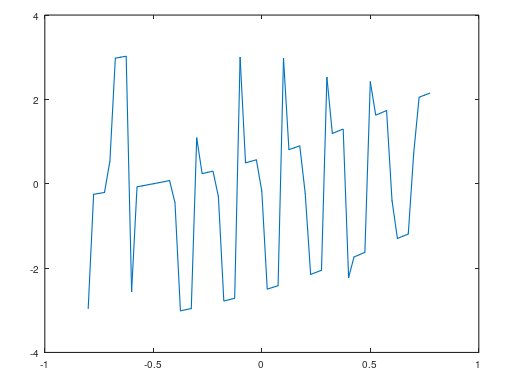


Рисунок 24 – График фазы БПФ светового поля при

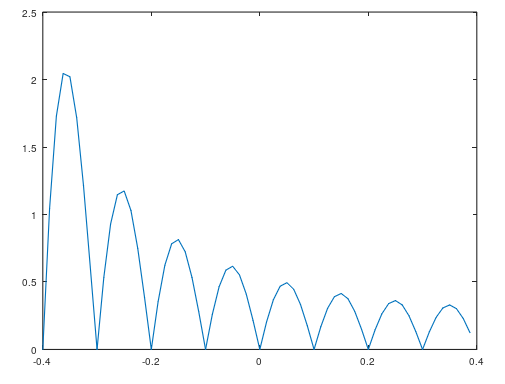


Рисунок 25 – График амплитуды БПФ светового поля при

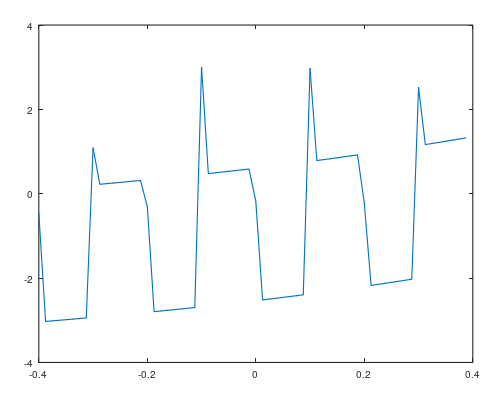


Рисунок 26 – График фазы БПФ светового поля при

Выводы по влиянию параметра представлены в таблице 1.

Таблица 1. Влияние параметров на график результата преобразования.

|  |  |
| --- | --- |
|  | При изменении (увеличении или уменьшении), выходная область преобразования меняется соответственно, а также каждый раз меняются экстремумы. |
|  | При увеличении , выходная область преобразования уменьшается, и масштаб графика увеличивается. |

**Задание 9:**

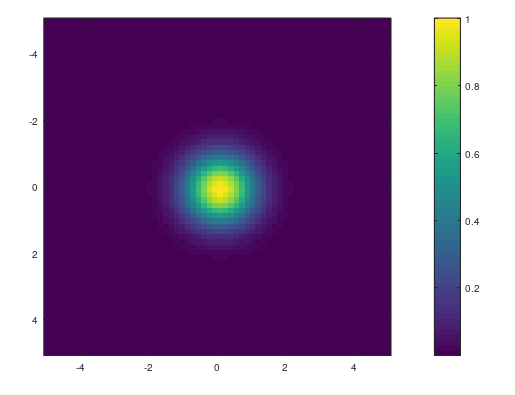


Рисунок 27 – Изображение амплитуды двумерного гауссова пучка

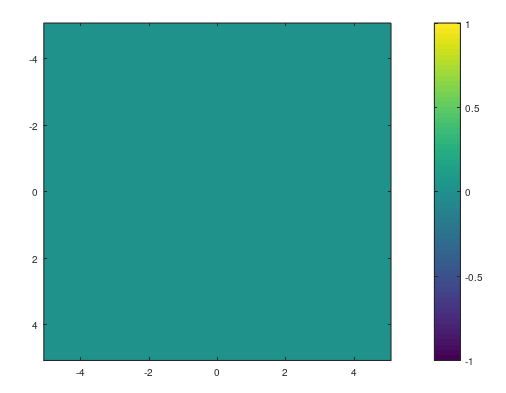


Рисунок 28 – Изображение фазы двумерного гауссова пучка

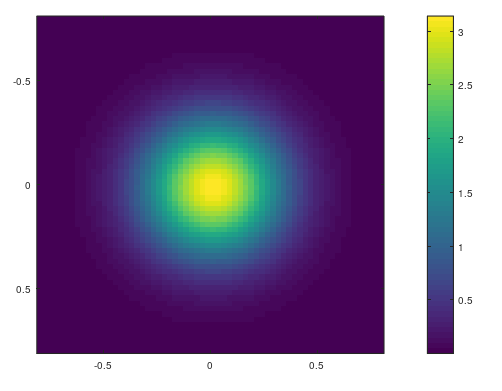


Рисунок 29 – Изображение амплитуды двумерного преобразования гауссова пучка

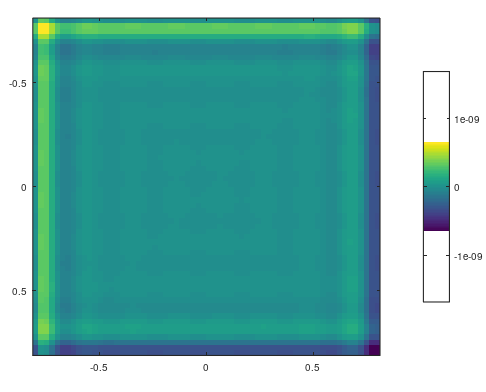


Рисунок 30 – Изображение фазы двумерного преобразования гауссова пучка

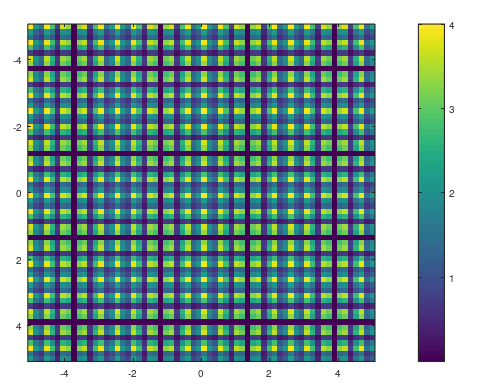


Рисунок 31 – Изображение амплитуды двумерного светового поля

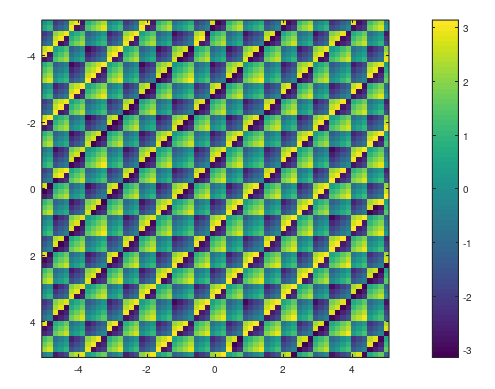


Рисунок 32 – Изображение фазы двумерного светового поля

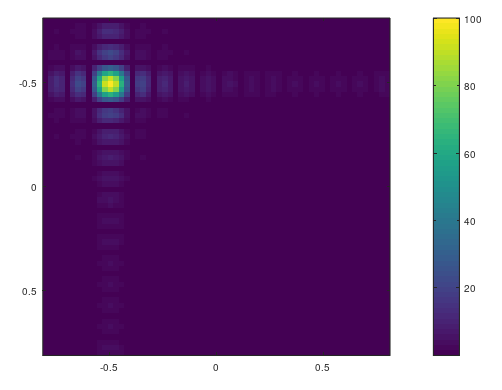


Рисунок 33 – Изображение амплитуды двумерного преобразования светового поля

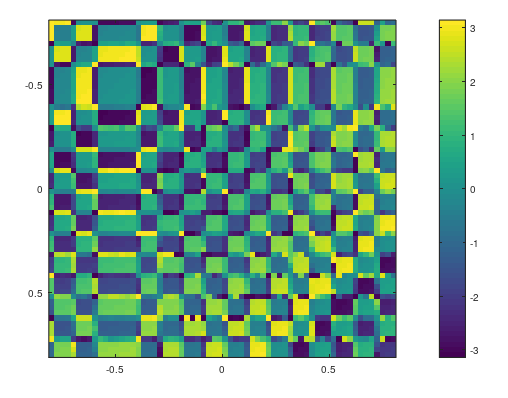


Рисунок 34 – Изображение фазы двумерного преобразования светового поля

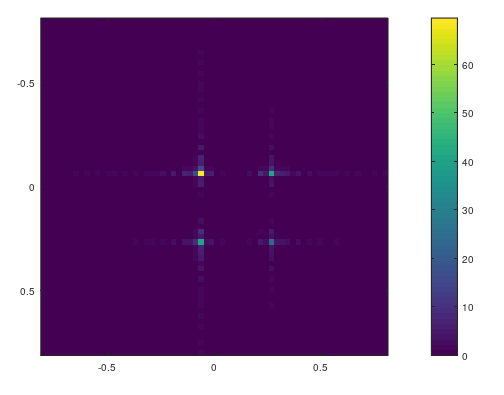


Рисунок 35 – Изображение амплитуды аналитического результата двумерного преобразования светового поля

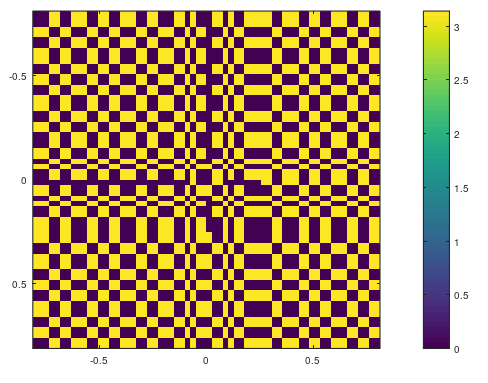


Рисунок 36 – Изображение фазы аналитического результата двумерного преобразования светового поля

# Код программы

clear;

# БПФ

function F = DFT(func, M, hx)

s = (M - length(func)) / 2;

func = [zeros(s, 1).', func, zeros(s, 1).'];

func = [func(1 + M/2:end), func(1:M / 2)];

F = fft(func) \* hx;

F = [F(1 + M/2:end), F(1:M / 2)];

F = F(s + 1:M - s);

end

# Аналитическое решение

function solve = AnaliticSolve(u, a)

#e = exp(2i\*a\*pi.\*u);

#solve = (1i/pi).\*(4\*u - 2)./(4\*u.^2 - 4\*u - 3).\*(e.^2-1)./e;

solve = (1i/pi).\*(4\*u - 2)./(4\*u.^2 - 4\*u - 3).\*(exp(2i\*a\*pi.\*u)-exp(-2i\*a\*pi.\*u));

end

# Аналитическое решение для двумерного случая

function solve = AnaliticSolve2D(u, v, a)

#e = exp(2i\*a\*pi.\*u);

#solve = (1i/pi).\*(4\*u - 2)./(4\*u.^2 - 4\*u - 3).\*(e.^2-1)./e;

solve = (-1/(pi^2)).\*(4\*u - 2)./(4\*u.^2 - 4\*u - 3).\*(exp(2i\*a\*pi.\*u)-exp(-2i\*a\*pi.\*u)).\*(4\*v - 2)./(4\*v.^2 - 4\*v - 3).\*(exp(2i\*a\*pi.\*v)-exp(-2i\*a\*pi.\*v));

end

N = 64;

a = 5;

hx = 2 \* a / N;

x = -a:hx:(a - hx / 2);

# Гауссов пучок

f = exp(-(x.^2));

#figure(1);

#plot(x, abs(f)); # Амплитуда

#figure(2);

#plot(x, arg(f)); # Фаза

# БПФ гауссова пучка

M = 256;

F = DFT(f, M, hx);

b = N ^ 2 / (4 \* a \* M);

hxi = 2 \* b / N;

xi = -b:hxi:(b - hxi / 2);

#figure(1);

#plot(xi, abs(F), "b"); # Амплитуда

#figure(2);

#plot(xi, arg(F), "b"); # Фаза

# Стандартный метод численного интегрирования

[X, XI] = meshgrid(x, xi);

Kernel = exp(-2 \* pi \* 1i \* X.\*XI);

F2 = Kernel \* f.' \* hx;

#figure(1);

#plot(xi, abs(F2), "k"); # Амплитуда

#figure(2);

#plot(xi, arg(F2), "k"); # Фаза

# Сравнение БПФ и стандартного метода численного интегрирования

#figure(3);

#plot(xi, abs(F)); # Амплитуда

#hold on;

#plot(xi, abs(F2), "k"); # Амплитуда

#figure(4);

#plot(xi, arg(F)); # Фаза

#hold on;

#plot(xi, arg(F2), "k"); # Фаза

# Cветовое поле

LF = exp(-pi \* 1i \* x) + exp(3 \* pi \* 1i \* x)

#figure(1);

#plot(x, abs(LF)); # Амплитуда

#figure(2);

#plot(x, arg(LF)); # Фаза

# Преобразование светового поля

FLF = DFT(LF, M, hx);

#figure(1);

#plot(xi, abs(FLF)); # Амплитуда

#figure(2);

#plot(xi, arg(FLF)); # Фаза

# Аналитическое решение

ASolve = AnaliticSolve(xi, a);

#figure(1);

#plot(xi, abs(FLF)); # Амплитуда

#hold on;

#plot(xi, abs(ASolve), "r"); # Фаза

#figure(2);

#plot(xi, arg(FLF)); # Амплитуда

#hold on;

#plot(xi, arg(ASolve), "r"); # Фаза

# Двумерный гауссов пучок

y = x.';

f = exp(-(x.^2)-(y.^2));

intervalA = [-a, a];

#figure(1);

#imagesc(intervalA, intervalA, abs(f)); # Амплитуда

#colorbar;

#figure(2);

#imagesc(intervalA, intervalA, arg(f)); # Фаза

#colorbar;

# Двумерное преобразование гауссова пучка

[X, Y] = meshgrid(x, y);

F2D = exp(-(X.^2)-(Y.^2));

intervalB = [-b, b];

for i = 1:rows(F2D)

F2D(i, :) = DFT(F2D(i, :), M, hx);

endfor

for j = 1:columns(F2D)

F2D(:, j) = DFT(F2D(:, j).', M, hx).';

endfor

#figure(1);

#imagesc(intervalB, intervalB, abs(F2D)); # Амплитуда

#colorbar;

#figure(2);

#imagesc(intervalB, intervalB, arg(F2D)); # Фаза

#colorbar;

# Двумерное световое поле

y = x.';

LF2D = (exp(-pi \* 1i \* x) + exp(3 \* pi \* 1i \* x)).\*(exp(-pi \* 1i \* y) + exp(3 \* pi \* 1i \* y));

#figure(1);

#imagesc(intervalA, intervalA, abs(LF2D)); # Амплитуда

#colorbar;

#figure(2);

#imagesc(intervalA, intervalA, arg(LF2D)); # Фаза

#colorbar;

# Двумерное преобразование светового поля

[X, Y] = meshgrid(x, y);

FLF2D = (exp(-pi \* 1i \* x) + exp(3 \* pi \* 1i \* x)).\*(exp(-pi \* 1i \* y) + exp(3 \* pi \* 1i \* y));

intervalB = [-b, b];

for i = 1:rows(FLF2D)

FLF2D(i, :) = DFT(FLF2D(i, :), M, hx);

endfor

for j = 1:columns(FLF2D)

FLF2D(:, j) = DFT(FLF2D(:, j).', M, hx).';

endfor

#figure(1);

#imagesc(intervalB, intervalB, abs(FLF2D)); # Амплитуда

#colorbar;

#figure(2);

#imagesc(intervalB, intervalB, arg(FLF2D)); # Фаза

#colorbar;

# Аналитическое решение двумерного случая

[X, Y] = meshgrid(x, y);

ASolve2D = AnaliticSolve2D(X, Y, a);

figure(1);

imagesc(intervalB, intervalB, abs(ASolve2D)); # Амплитуда

colorbar;

figure(2);

imagesc(intervalB, intervalB, arg(ASolve2D)); # Фаза

colorbar;