**ЛАБОРАТОРНАЯ РАБОТА №4**

**«АНАЛИЗ СЛУЧАЙНЫХ ПРОЦЕССОВ»**

* 1. **Цель работы**

1. Изучить основы статистического описания случайных процессов.

2. Изучить методы нахождения числовых характеристик случайных

величин.

3. Научится применять методы корреляционного и спектрального анализа

к решению практических задач.

4. Освоить способы программного моделирования случайных процессов.

**4.2 Постановка задачи (Вариант – 9)**

Ознакомится с функциями пакета MATLAB, **uigetfile**, **hist**, **xcov**, **plot**, **imread**, **Imshow**, **double**, **stem**, **xcorr**, **mean**, **std**.

Используя ниже, приведённый фрагмент программы, ввести указанное изображение в систему MATLAB.

clear all; % очистка рабочего пространства

close all; % закрываем все созданные фигуры

Ts=0.01; % шаг во времени (с) (частота квантования)

T= 100; % длительность процесса (с)

% ПОЛУЧЕНИЕ КАРТИНКИ ИЗОБРАЖЕННИЯ

[F\_Name,PathName]=uigetfile('\*.tif','Выберите имя файла с изображением');

% используем пользовательский интерфейс для выбора файла с картинкой

I = imread(F\_Name); % ввод имени файла и чтение изображения в переменную I

figure(1);

imshow(I); % отображение картинки в figure 1

Чтобы обеспечить возможность дальнейшей обработки изображения, необходимо преобразовать беззнаковое целое **uint8** изображения к формату **double**.

При помощи команды stem получим изображение случайного процесса.

% ПОСТРОЕНИЕ ГРАФИКА ПРОЦЕССА

A=double(I); % преобразование типов – беззнакового целого uint8 к double для

% обеспечения возможности выполнения арифметических операций

variable = A(:,1); % выбираем 1 столбец для формирования вектора случайного процесса

figure(2);

stem(variable);

title('PROCES');

ylabel('Y');

xlabel('N');

При помощи функции **hist** построить гистограмму случайного процесса.

% ПОСТРОЕНИЕ ГИСТОГРАММЫ

n=length(variable); % получаем длину вектора случайного процесса

k=round(sqrt(n)); % определение оптимального количества интервалов гистограммы

figure(3);

hist(variable, k); % построение гистограммы процесса

title('HISTOGRAMMA'); ylabel('Q') xlabel('N');

Используя функцию MATLAB **pwelch (psd, periodogramm)** рассчитать спектральную плотность случайного процесса согласно приведённому ниже примеру.

% ПОСТРОЕНИЕ СП ПРИ ПОМОЩИ ПРОЦЕДУРЫ PSD

% [s, f]=psd(x, nfft, Fmax), где: x - вектор заданных значений процесса, nfft - число элементов этого вектора, Fmax= 1/Ts – частота дискретизации сигнала, f - вектор значений частот, которые соответствуют найденные значения СП. В общем случае длина s и f равна nfft/2.

% сформируем массив частот где: df - дискрет частоты, Fmax – величина диапазона частот

fsp=250; % правая граница выводимого вектора частот для СП

df=1/T; Fmax=1/Ts; f=-Fmax/2:df:Fmax/2; dovg=length(f);

[c, f]=psd(variable, dovg, Fmax);

figure(4);

stem(f(1:fsp), c(1:fsp));grid; title('PSD'); ylabel('SP');

xlabel('frequency');

Применив функцию MATLAB **xcorr** произвести автоковариацию случайного процесса согласно представленному ниже программному коду.

% ПОСТРОЕНИЕ АКФ СЛУЧАЙНОГО ПРОЦЕССА

% tau - сдвиг

R=xcorr(variable); % расчёт автоковариационной функции

tau=-1.98:0.01:1.98;

figure(5);

plot( tau, R); grid;

title('AKVF');

label('Bcov');

xlabel('tau');

Применив функцию MATLAB **xcov** произвести автокорреляцию случайного процесса согласно представленному ниже программному коду.

R1=xcov(variable); % расчёт автокорреляционной функции

tau=-1.98:0.01:1.98;

figure(5);

plot( tau, R1); grid;

title('AKRF');

label('Bcor');

xlabel('tau');

Согласно формуле рассчитать числовые характеристики случайного процесса.

Создать М-файл программы на языке MATLAB.

На рисунке 4.1 представлен рисунок, который будет исследоваться.



Рисунок 4.1 – Исследуемый рисунок

**4.3 Ход работы**

С помощью примера кода программы после ознакомления с функциями пакета MATLAB, uigetfile, hist, xcov, plot, imread, Imshow, double, stem, xcorr, mean, std была написана программа выполняющая действия требуемые от студента. Результаты представлены на рисунках 4.2 – 4.7.

% ДОБАВЛЕНИЕ ИЗОБРАЖЕНИЯ В MATLAB

clear all; % очистка рабочего пространства

close all; % закрываем все созданные фигуры

Ts=0.01; % шаг во времени (с) (частота квантования)

T=100; % длительность процесса (с)

% тут вводится путь к изображению

[F\_Name,PathName]=uigetfile('\*.png','Выберите имя файла с изображением');

I=imread([PathName F\_Name]);

figure(1); % создание области figure 1

imshow(I); % отображение картинки в figure 1

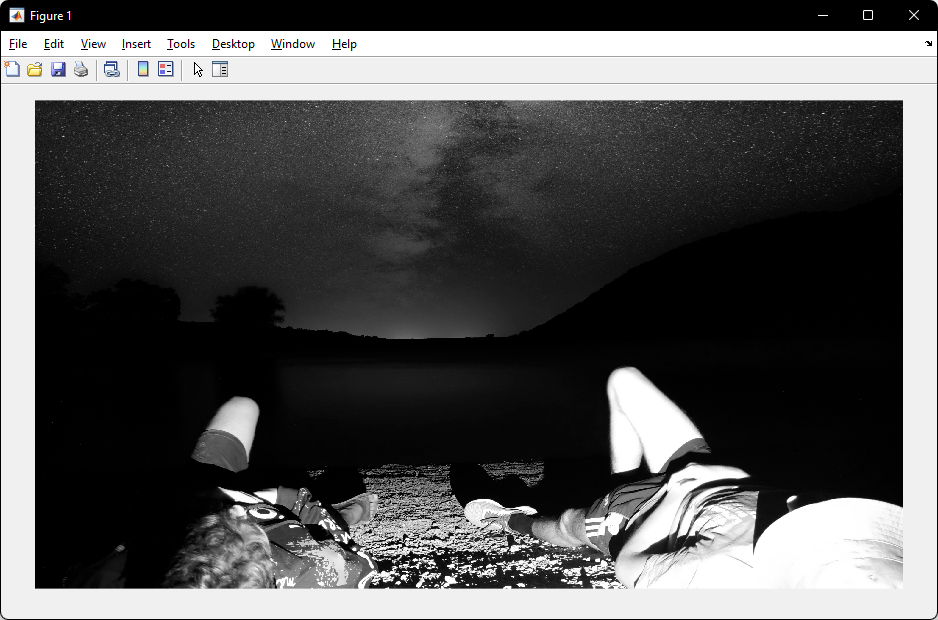


Рисунок 4.2 – Исследуемое изображение

% ПОСТРОЕНИЕ ГРАФИКА ПРОЦЕССА

A=double(I); % преобразование типов – беззнакового целого uint8 к double для обеспечения возможности выполнения арифметических операций

variable = A(:,1); % выбираем 1 столбец для формирования вектора случайного процесса

figure(2);

stem(variable);

title('PROCES');

ylabel('Y');

xlabel('N');

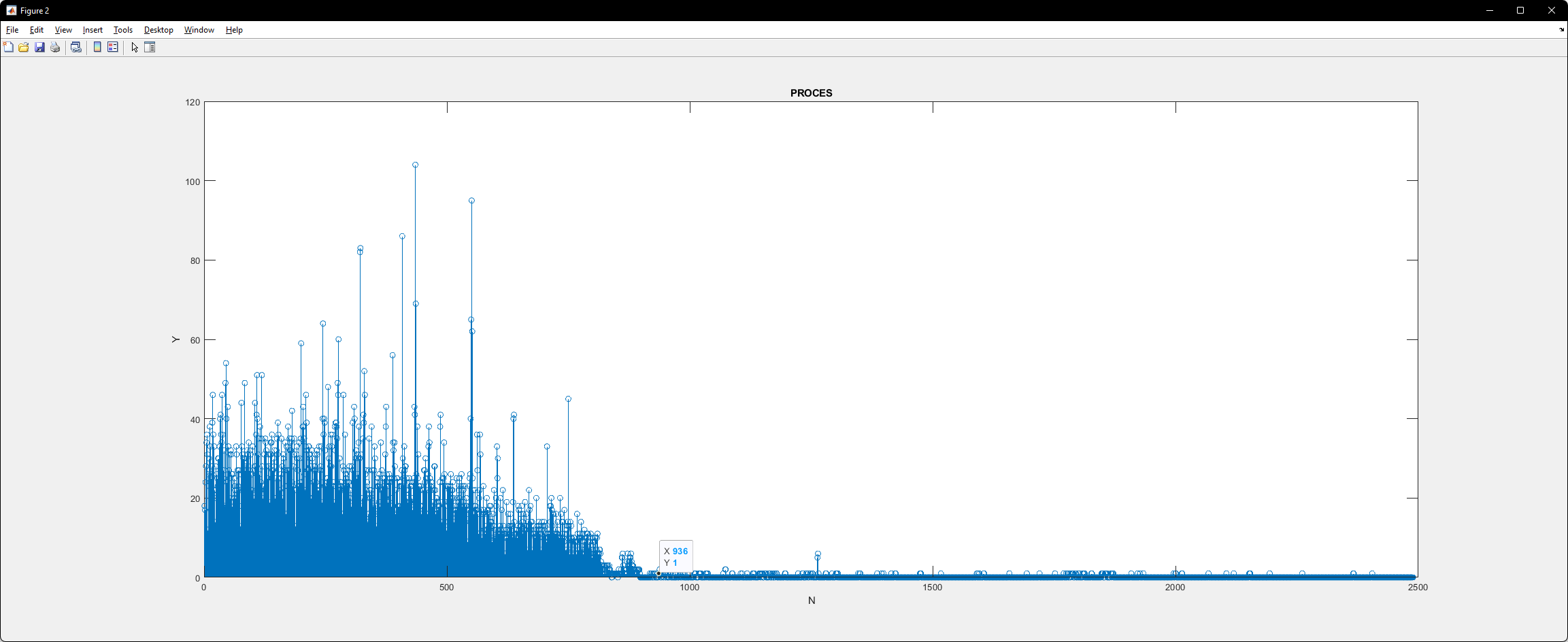


Рисунок 4.3 – График случайного процесса, полученного из столбца матрицы введённого изображения. Y–величина яркости, N–номер отсчёта.

% ПОСТРОЕНИЕ ГИСТОГРАММЫ

n=length(variable); % получаем длину вектора случайного процесса

k=round(sqrt(n)); % определение оптимального количества интервалов гистограммы

figure(3);

hist(variable, k); % построение гистограммы процесса

title('HISTOGRAMMA');

ylabel('Q');

xlabel('N');

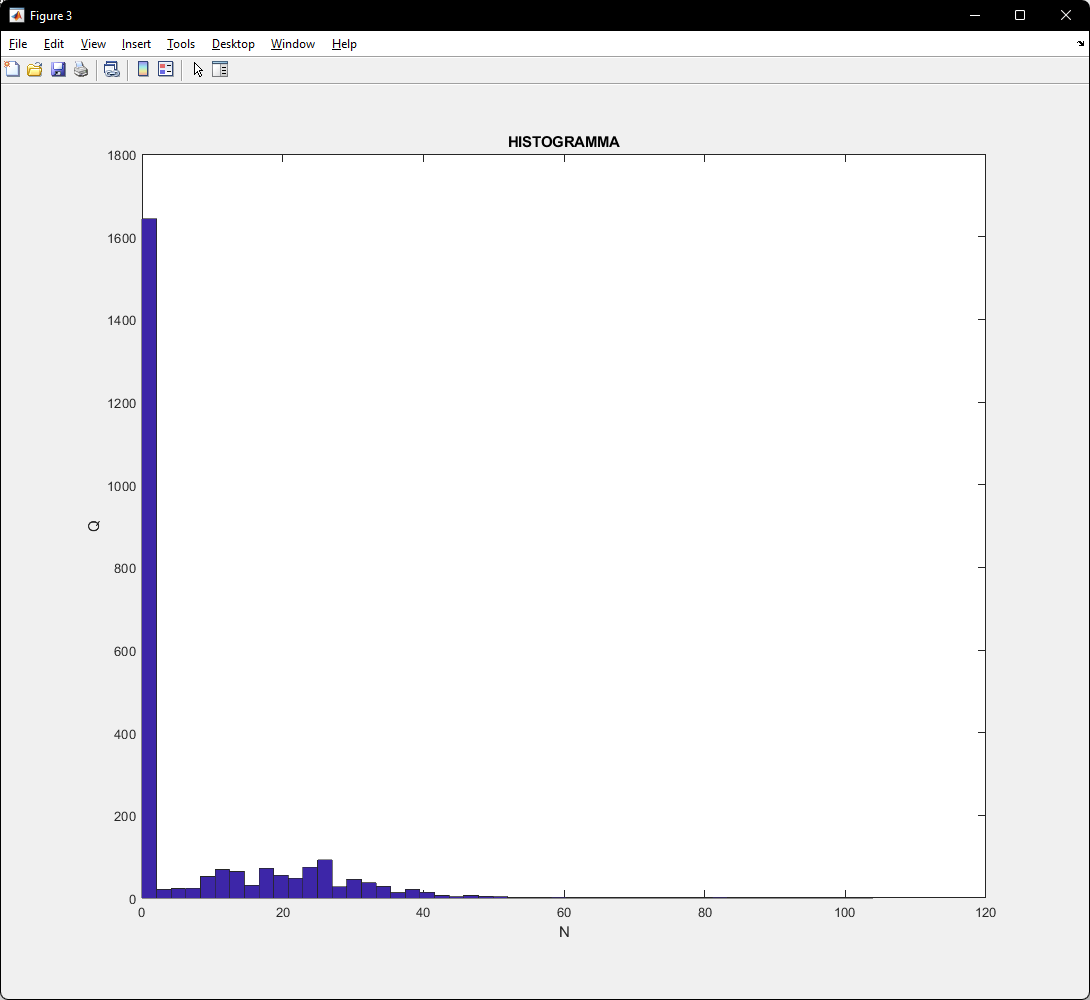


Рисунок 4.4 – Гистограмма случайного процесса. Y–величина яркости, Q – частота попадания случайной величины в заданный интервал.

% ПОСТРОЕНИЕ СП ПРИ ПОМОЩИ ПРОЦЕДУРЫ PSD

fsp=250;

df=1/T;

Fmax=1/Ts;

f=-Fmax/2:df:Fmax/2;

dovg=length(f);

[c, f] = pwelch(variable, [], [], dovg, Fmax);

figure(4);

stem(f(1:fsp), c(1:fsp));

grid;

title('PSD');

ylabel('SP');

xlabel('frequency');

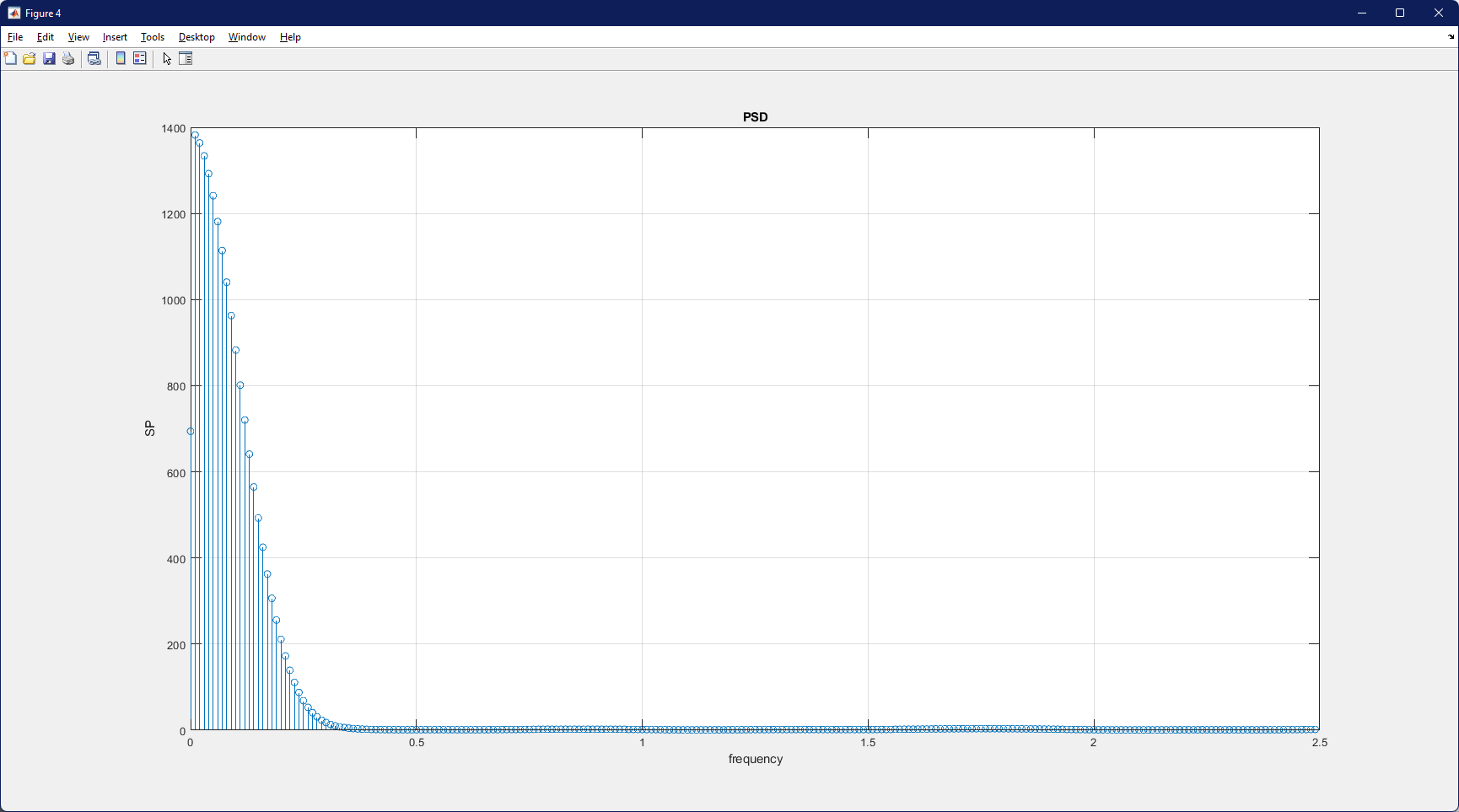


Рисунок 4.5 – График функции спектральной плотности случайного процесса

SP – спектральная плотность случайного процесса, ось абсцисс – частота

% ПОСТРОЕНИЕ АКФ СЛУЧАЙНОГО ПРОЦЕССА

R=xcorr(variable); % расчёт автоковариационной функции

tau = -(n/100 - 0.01):0.01:(n/100 - 0.01);

figure(5);

plot(tau, R);

grid;

title('AKVF');

ylabel('Bcov');

xlabel('tau');

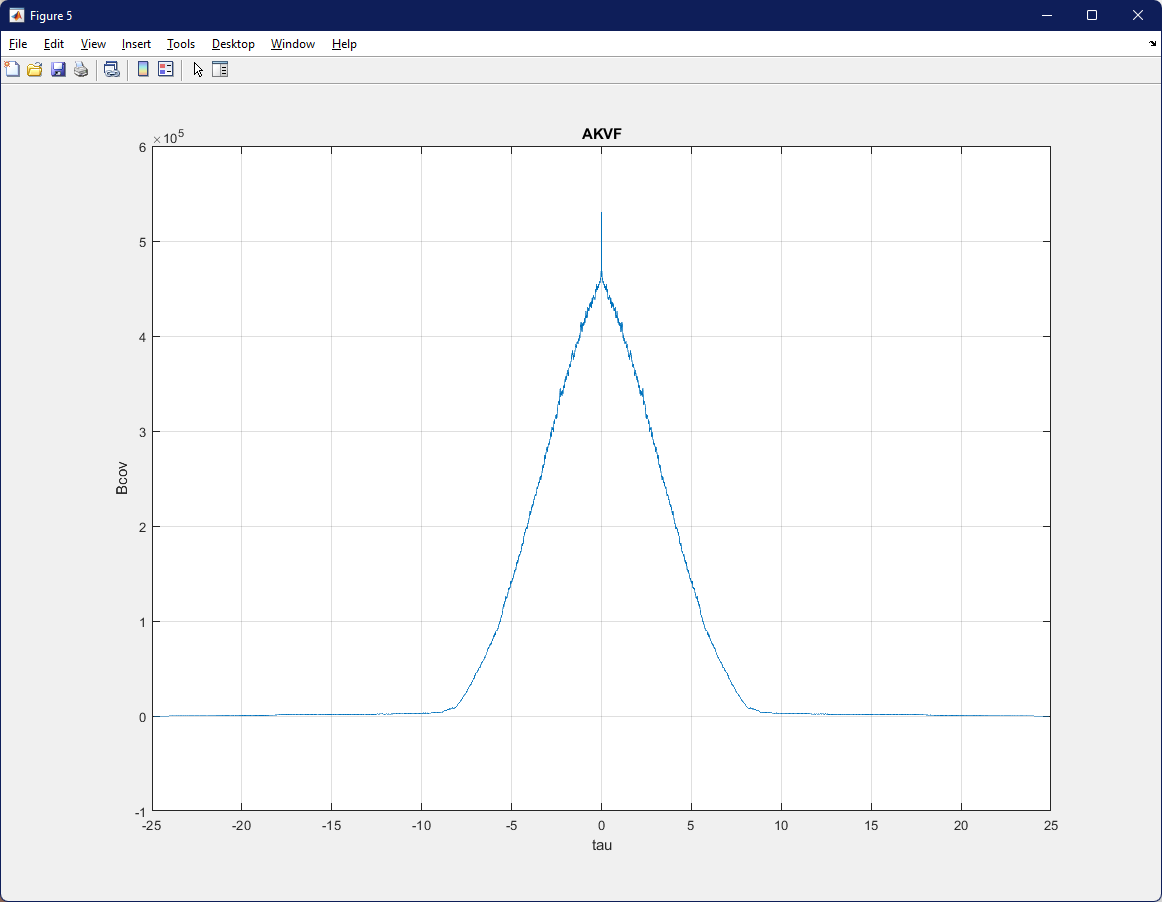


Рисунок 4.6 – График автоковариационной функции случайного процесса

Bcov – автоковариационная функция случайного процесса, tau – временной сдвиг

R1=xcov(variable);

figure(6);

plot(tau,R1);

grid;

title('AKRF');

ylabel('Bcor');

xlabel('tau');

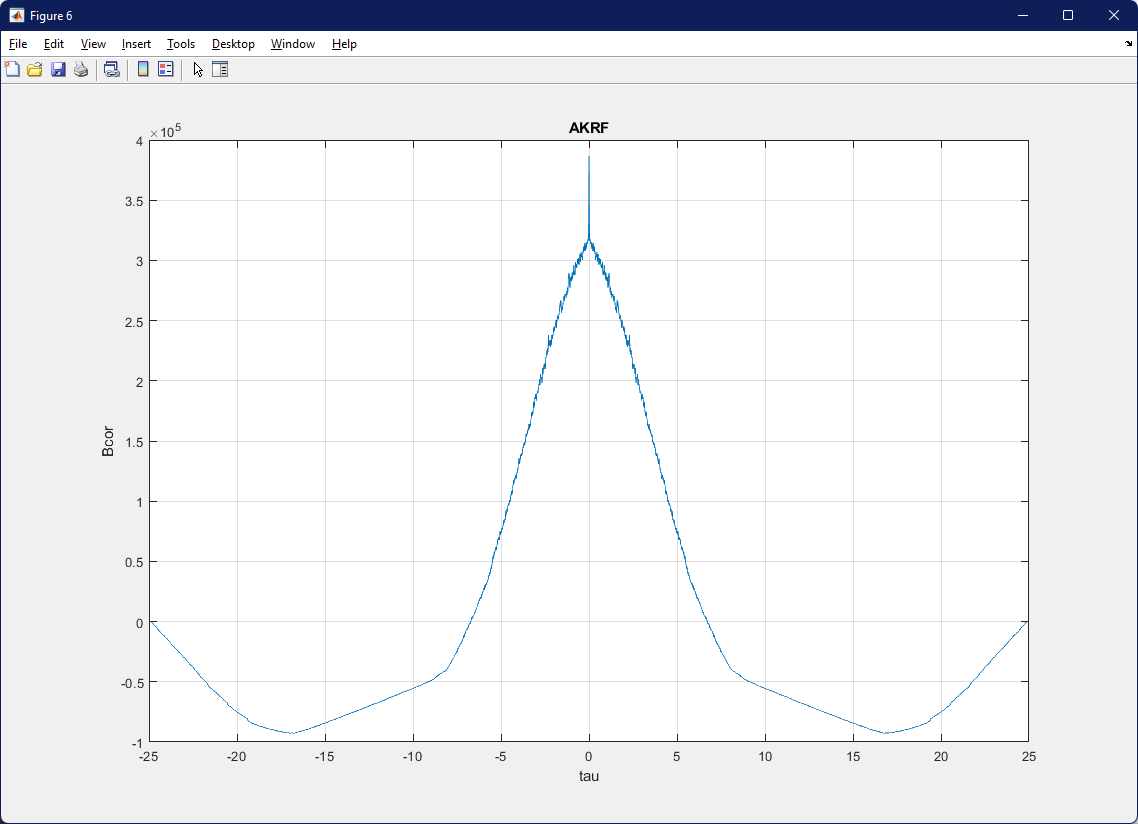


Рисунок 4.7 – График автокорреляционной функции случайного процесса

Bcor – автокорреляционная функция случайного процесса, tau – временной сдвиг.

function B = meanearch(A, n);

sum = 0;

B = zeros(1,n);

for i = 1:n

sum = sum + A(i);

B(i) = sum / i;

end

% оценки численных характеристик

R = variable;

n = length(variable);

M1 = meanearch(R, n);

fprintf('Оценка математического ожидания: %g\n', M1(n));

mu = zeros(4, n);

for i = 1:4

mu(i, :) = meanearch( (R - M1(n)) .^ i, n);

fprintf('Оценка центрального момента %d-го порядка случайной величины: %g\n', i, mu(i, n));

end

y = zeros(2, n);

y(1, :) = mu(3, :) ./ (mu(2, :) .^ (3/2));

y(2, :) = mu(4, :) ./ (mu(2, :) .^ 2) - 3;

fprintf('\nОценка дисперсии: %g\n', mu(2, n));

fprintf('Оценка среднеквадратического значения: %g\n', sqrt(mu(2, n)));

fprintf('Оценка коэффициента асимметрии: %g\n', y(1, n));

fprintf('Оценка коэффициента эксцесса: %g\n', y(2, n));

Результаты конечной части программы:

Оценка математического ожидания: 7.59502

Оценка центрального момента 1-го порядка случайной величины: 1.42672e-13

Оценка центрального момента 2-го порядка случайной величины: 155.248

Оценка центрального момента 3-го порядка случайной величины: 3719.28

Оценка центрального момента 4-го порядка случайной величины: 192406

Оценка дисперсии: 155.248

Оценка среднеквадратического значения: 12.4598

Оценка коэффициента асимметрии: 1.92274

Оценка коэффициента эксцесса: 4.98305

**Выводы**

В ходе выполнения лабораторной работы были изучены методы статического описания случайных величин, методы нахождения числовых характеристик случайных величин, освоены способы программного моделирования случайных процессов. Результаты моделирования и теоретические расчеты приведены в отчете.

Чтобы определить числовые характеристики случайного процесса, необходимо знать, как он ведёт себя в любой момент времени и является ли он стационарным. Если процесс эргодичен, можно найти характеристики не по ансамблю реализаций, а по одной из большого количества реализаций.