Цель работы

Получение геометрического представления сигналов. Промоделировать работу приемника дискретных сигналов в канале с аддитивным белым гауссовским шумом.

Исходные данные

Вариант 3.3.

Выполнение работы

1. Данные из прошлых работ.
   1.1. В случае квадратурной амплитудной модуляции, базисными будут являться следующие функции:

(1)

В таком случае размерность базиса D = 2.

   1.2. Для заданного варианта КАМ, q = 4. Для коэффициентов разложения s_{ij} используются следующие формулы:

(2)

Где i_k – числа в представлении числа i.

   1.3. Сигнал же представляется линейной комбинацией D базисных функций

(3)

   1.4. Изобразим сигнальное созвездие.

Таблица 1 - Значение коэффициентов i_k

i_1	i_2	(s_{i1},s_{i2})
0	0	(1, 1)
0	1	(1, -1)
1	0	(-1, 1)
1	1	(-1, -1)

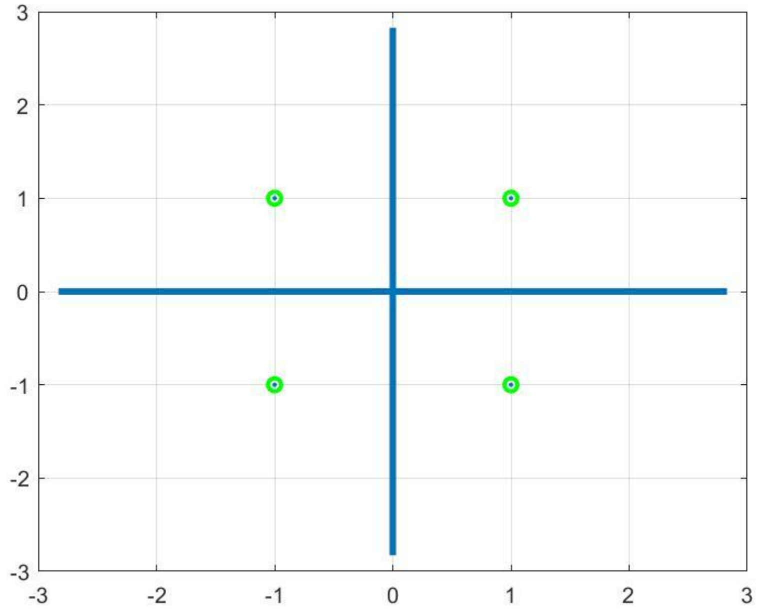

Рисунок 1 - Сигнальное созвездие КАМ 4

Как и предполагалось, элементы сигнального созвездия расположены с равномерным шагом в интервале [-A, A].

2. Алгоритм моделирования передачи по каналу оптимального приёма.

Алгоритм состоит из следующих шагов:

2.1.1. Случайный равновероятный выбор номера сигнала i, подлежащего передаче, i = 0, 1, …, q-1.

2.1.2. Получение сигнала на выходе сигнала, согласно равенству , где n(t) – АБГШ со спектральной плотностью мощностью

2.1.3. Вычисление вектора с компонентами

2.1.4. Формирование решение по правилу

(4)

3. Вывод теоретического выражения вероятности ошибки

3.1. Формула для вероятности ошибки выглядит следующим образом

(5)

3.2. Для нашего варианта формула сокращается к

(6)

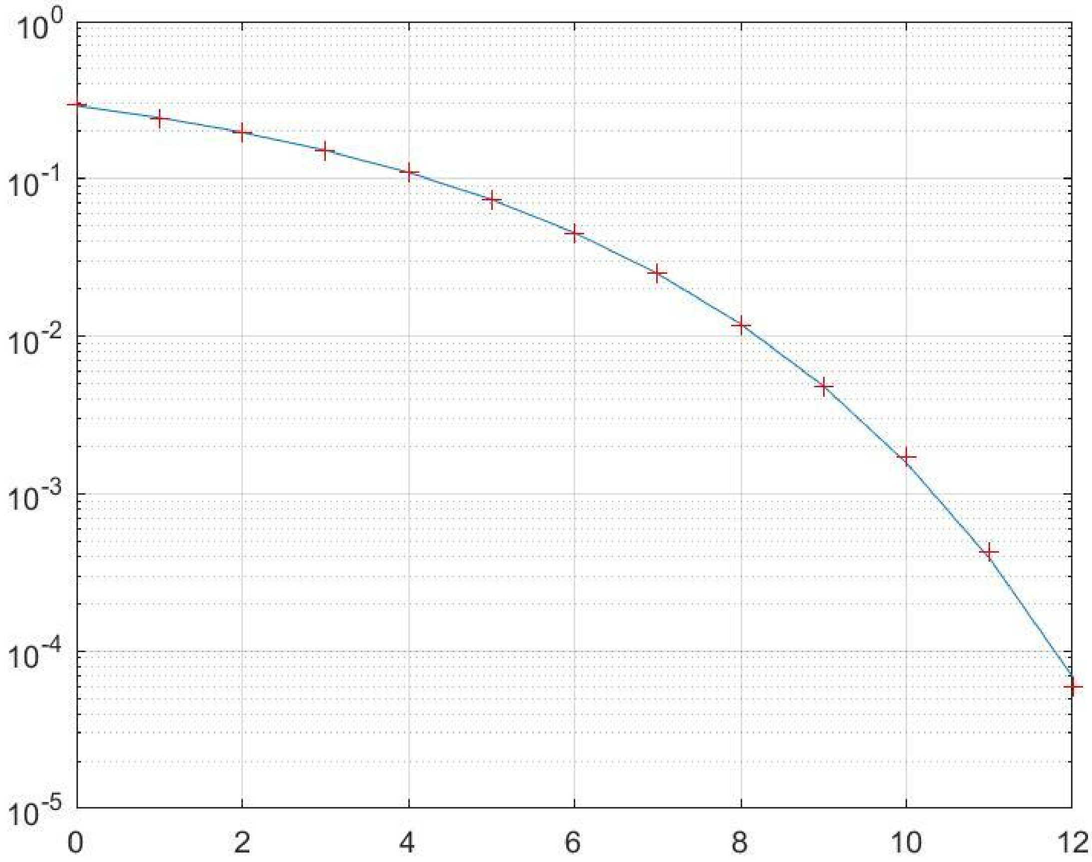

Рисунок 2 – График зависимости и результаты моделирования.

Теоретический расчёт частично совпал с моделированным результатом.

Вывод

В ходе выполнения лабораторной работы была промоделирована работа приемника дискретных сигналов в канале с аддитивным белым гауссовским шумом. Были рассчитаны вероятности ошибок в работе приемника при различных значениях SNR. Данные вероятности ошибок были сравнены с полученными в ходе моделирования вероятностями ошибок (Рисунок 2). Так эти вероятности оказались приблизительно равны.

Листинг

```
clear;
close all;

q = 4;
i1 = [0, 0, 1, 1];
i2 = [0, 1, 0, 1];
X = zeros(0, q);
Y = zeros(0, q);
for i = 1 : q
    X(i) = (1-(2*i1(i)/(sqrt(q)-1)));
    Y(i) = (1-(2*i2(i)/(sqrt(q)-1)));
end
f = 2400;
T = 1/600;

step = T/1000;
t = 0: step : T - 10^(-8);
s = zeros(q, 1000);

b = zeros(2, 1000);
b(1, :) = (2/T)^(1/2)*cos(2*pi*f*t);
b(2, :) = (2/T)^(1/2)*sin(2*pi*f*t);

sig_noise = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10];

n = zeros(1000);

for i = 1:q
    s(i, :) = X(i)
*((2/T)^(1/2))*cos(2*pi*f*t)+Y(i)*((2/T)^(1/2))*sin(2*pi*f*t);
end

g = matrix(sig_noise);
e = matrix(sig_noise);
Pe = matrix(sig_noise);
P = matrix(sig_noise);
Pr = matrix(sig_noise);
E = norm(s(1,:))^2;

for sn = 1:length(sig_noise)
    g(sn) = 10^(sig_noise(sn)/10);
    N0 = E/g(sn);
    for i = 1:100000
        n = sqrt(N0/2)*randn(1, 1000);
        I = randi([1, 4]);
        r = s(I, :)+n;
        mind = Inf;
        n1 = sum(r .* b(1, :))*step;
        n2 = sum(r .* b(2, :))*step;
        newI = 0;
        for j = 1:q
           if ((n1 - X(j))^2 + (n2 - Y(j))^2 < mind)
              mind = (n1 - X(j))^2 + (n2 - Y(j))^2;
              newI = j;
           end
        end
        if (newI ~= I)
            e(sn) = e(sn) + 1;
        end
    end
```

```
    Pr(sn) = e(sn)/100000;
    P(sn) = qfunc(sqrt(((3*E)/N0) * (1/(q-1))));
    Pe(sn) = 2*P(sn) - P(sn)^2;
end

figure(1);
semilogy(sig_noise, Pe);
hold on;
for i = 1:length(sig_noise)
    plot(sig_noise(i), Pr(i), '+', 'Color', 'r');
end
grid on;

function m = matrix(snr)
    m = zeros(length(snr), 1);
end
```