

Лосева Елизавета Юрьевна

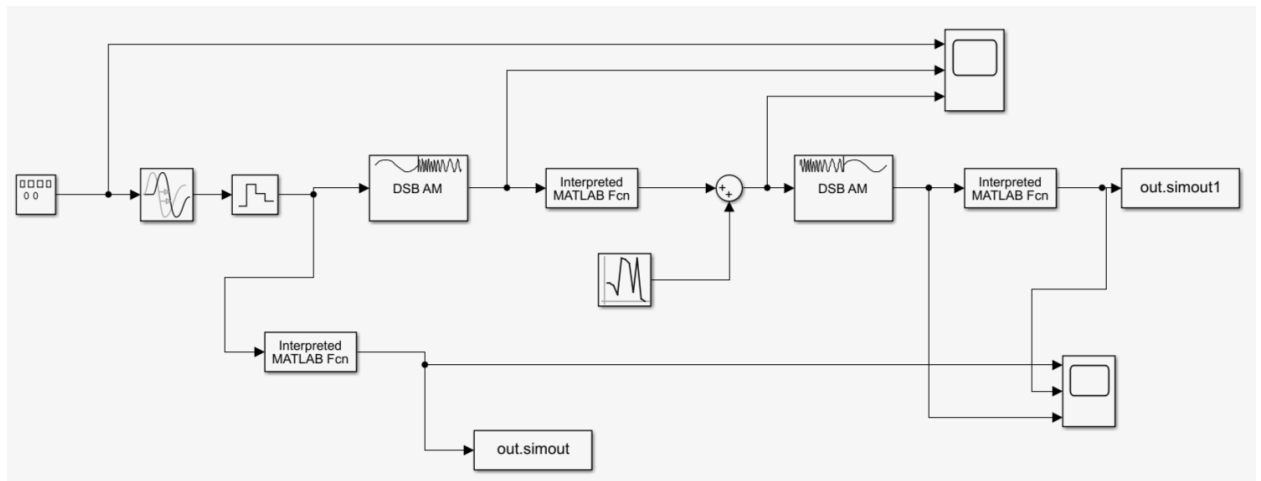
Группа 8.1


Лабораторная работа 2

Вариант 166

Путем имитационного моделирования канала передачи информации с полезным В радиосигналом и С модуляцией (несущая частота – D Гц) в среде с мешающим аддитивным белым шумом, построить зависимость O от $E \in [F; G]$ и $H \in [I; J]$, если $K = L$ и $M = N$. Точность экспериментов определяется доверительным интервалом $dp = P$ с уровнем значимости $\alpha = Q$.
Время моделирования – 10 секунд.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
№	Вид исходного сигнала	Тип модуляции	Несущая частота	$\Phi 1$	Значение мин.	Значение макс.	$\Phi 2$	Значение мин.	Значение макс.	K1	Значение	K2	Значение	Показатель эффективности	Доверительный интервал	Уровень значимости
166	Синусоидальный	Амплитудная	600	NP	0	0,03	R	66	134	mr	8,58	A	33	Среднеквадратическая ошибка передачи	0,07	0,05



 Block Parameters: Signal Generator

✕

Signal Generator

Output various wave forms:
 $Y(t) = \text{Amp} * \text{Waveform}(\text{Freq}, t)$

Parameters

Wave form:

sine

Time (t):

Use simulation time

Amplitude:

Am


Frequency:

20

Units:

Hertz

☒ Interpret vector parameters as 1-D




OK

Cancel

Help

Apply

 Block Parameters: Zero-Order Hold

✕


Zero-Order Hold

Zero-order hold.

Parameters

Sample time (-1 for inherited):

Ts



OK

Cancel

Help

Apply

DSB AM Modulator Passband (mask) (link)

Modulate the input signal using the double-sideband amplitude modulation method.

The input signal must be a scalar.

Parameters

Input signal offset:


 ⋮

Carrier frequency (Hz):

 ⋮

Initial phase (rad):

 ⋮

 Block Parameters: Interpreted MATLAB Function1 ✕

Interpreted MATLAB Function

Pass the input values to a MATLAB function for evaluation. The function must return a single value having the dimensions specified by 'Output dimensions' and 'Collapse 2-D results to 1-D'. Examples: sin, sin(u), foo(u(1), u(2))

Parameters

MATLAB function:

Output dimensions:

Output signal type: auto

☐ Collapse 2-D results to 1-D

OK

Cancel

Help

Apply

Random Number

Output a normally (Gaussian) distributed random signal. Output is repeatable for a given seed.

Parameters

Mean:



Variance:



Seed:



Sample time:

☒ Interpret vector parameters as 1-D

OK

Cancel

Help

Apply

DSB AM Demodulator Passband (mask) (link)

Demodulate a double-sideband amplitude modulated signal.

The input signal must be a scalar.

Parameters

Input signal offset:



Carrier frequency (Hz):



Initial phase (rad):



Lowpass filter design method: Butterworth



Filter order:



Cutoff frequency (Hz):



Interpreted MATLAB Function

Pass the input values to a MATLAB function for evaluation. The function must return a single value having the dimensions specified by 'Output dimensions' and 'Collapse 2-D results to 1-D'. Examples: sin, sin(u), foo(u(1), u(2))

Parameters

MATLAB function:

Output dimensions:

Output signal type: auto

☐ Collapse 2-D results to 1-D

OK

Cancel

Help

Apply

Interpreted MATLAB Function

Pass the input values to a MATLAB function for evaluation. The function must return a single value having the dimensions specified by 'Output dimensions' and 'Collapse 2-D results to 1-D'. Examples: `sin`, `sin(u)`, `foo(u(1), u(2))`

Parameters

MATLAB function:


Output dimensions:



Output signal type: auto



☐ Collapse 2-D results to 1-D

 Block Parameters: To Workspace1 ✕

To Workspace


Write input to specified timeseries, array, or structure in a workspace. For menu-based simulation, data is written in the MATLAB base workspace. Data is not available until the simulation is stopped or paused.

To log a bus signal, use "Timeseries" save format.


Parameters


Variable name:


Limit data points to last:

Decimation:


 


Save format: Array 

Save 2-D signals as: 2-D array (concatenate along first dimension) 

☒ Log fixed-point data as a fi object

Sample time (-1 for inherited):

 OK Cancel Help Apply



Block Parameters: To Workspace



To Workspace

Write input to specified timeseries, array, or structure in a workspace. For menu-based simulation, data is written in the MATLAB base workspace. Data is not available until the simulation is stopped or paused.

To log a bus signal, use "Timeseries" save format.

Parameters

Variable name:

simout1

Limit data points to last:

inf

Decimation:

1

Save format: Array

Save 2-D signals as: 2-D array (concatenate along first dimension)

☒ Log fixed-point data as a fi object

Sample time (-1 for inherited):

-1

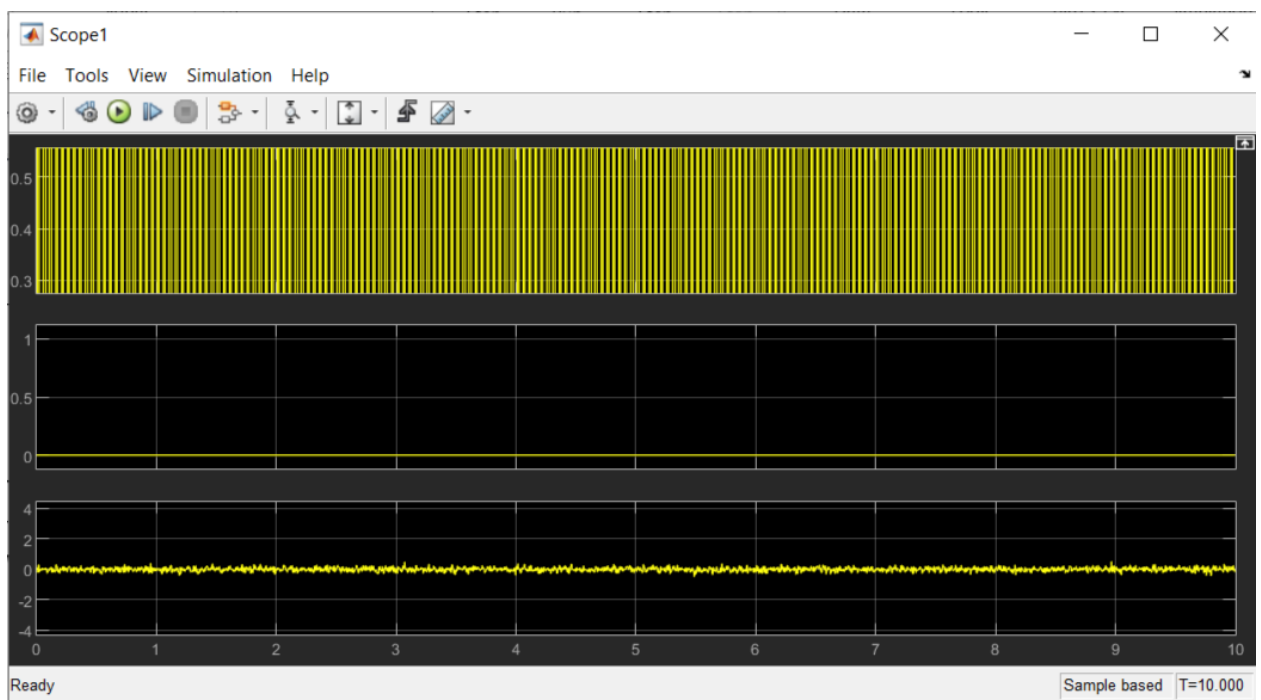
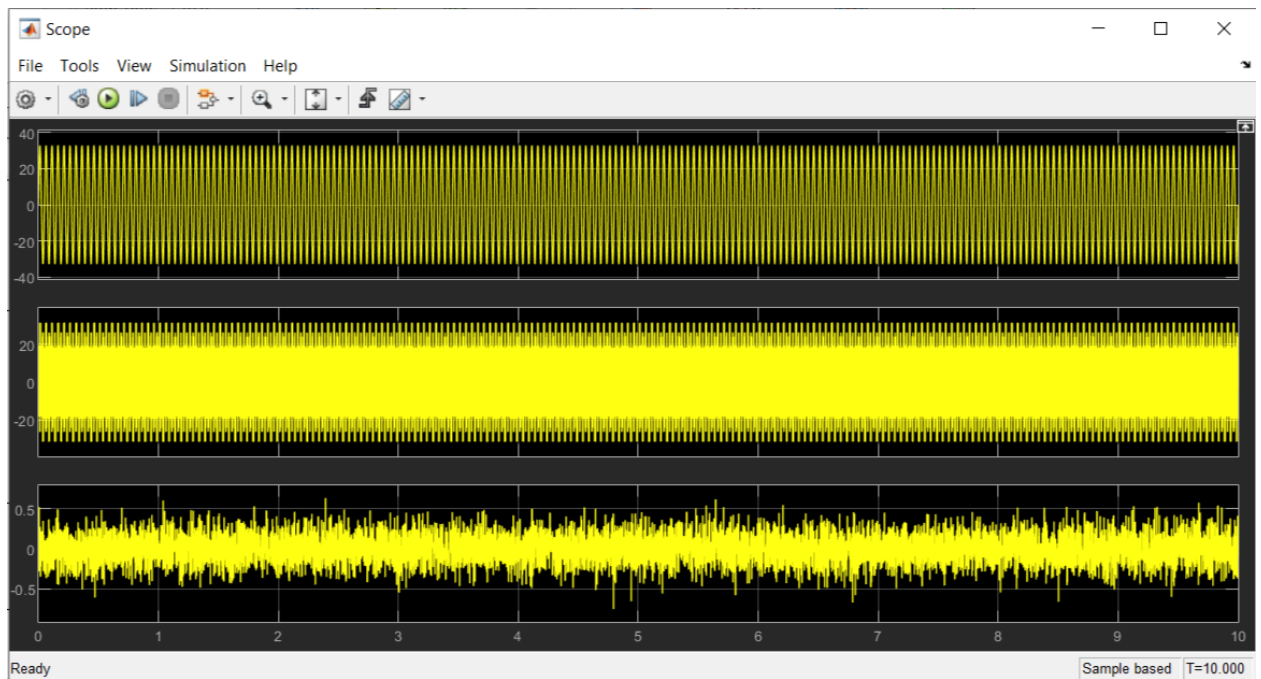


OK

Cancel

Help

Apply



Код лабораторной работы:

```
% ЛР: Оценка среднеквадратической ошибки передачи
clear; clc; % Очистка рабочей области и командного окна

%% Базовые параметры моделирования (0..10 с)
Ts = 0.001; % Время дискретизации (шаг моделирования)
Ns = 10000; % Количество отсчетов (время моделирования = Ts*Ns = 10 сек)

% Параметры по вашему варианту
fc = 600; % Несущая частота сигнала
```

```

mr = 8.58; % Коэффициент модуляции (константа)
A = 33; % Амплитуда сигнала (константа)
NP_min = 0; % Ф1 мин: мощность шума
NP_max = 0.03; % Ф1 макс: мощность шума
R_min = 66; % Ф2 мин: сопротивление нагрузки
R_max = 134; % Ф2 макс: сопротивление нагрузки

%% Факторы эксперимента: NP и R
nf = 2; % Количество факторов (NP и R)
minf = [NP_min R_min]; % Минимальные значения факторов [NP_min, R_min]
maxf = [NP_max R_max]; % Максимальные значения факторов [NP_max, R_max]

% Построение факторного плана 2^2
fracplan = fracfact('a b ab'); % Создание дробного факторного плана для 2 факторов
N = 2^nf; % Общее число экспериментов в плане (2^2 = 4)
fictfact = ones(N,1); % Столбец фиктивной переменной для свободного члена
X = [fictfact fracplan]; % Матрица плана с фиктивной переменной

% Натуральные значения NP и R
fraceks = zeros(N, nf); % Инициализация матрицы натуральных значений факторов
for i = 1:nf
    for j = 1:N
        % Преобразование нормированных значений (-1,+1) в натуральные
        fraceks(j,i) = minf(i) + (fracplan(j,i)+1)*(maxf(i)-minf(i))/2;
    end
end
disp('План в натуральных единицах [NP, R]:');
disp(fraceks);

%% Тактическое планирование точности
d_sigma = 0.7; % Относительная погрешность оценки дисперсии (доверительный интервал)
alpha = 0.05; % Уровень значимости
tkr_alpha = norminv(1 - alpha/2); % Квантиль нормального распределения
NE = round(1 + 2*tkr_alpha^2/d_sigma^2); % Расчет числа испытаний
fprintf('Расчетное число испытаний NE (для дисперсии) = %d\n', NE);

%% Основной цикл по точкам плана: MSE = f(NP,R) через XOR
Y = zeros(N,1); % Вектор для хранения результатов экспериментов

for j = 1:N
    NP_val = fraceks(j,1); % Текущее значение мощности шума
    R_val = fraceks(j,2); % Текущее значение сопротивления нагрузки

    uo = zeros(NE,1); % Вектор для переданных битов
    ul = zeros(NE,1); % Вектор для принятых битов

    for k = 1:NE
        to = round(rand*100); % Случайное начальное время для разнообразия

        % Передача параметров в базовую область для модели Simulink
        assignin('base','to',to);
        assignin('base','Ts',Ts);
        assignin('base','Ns',Ns);
        assignin('base','fc',fc);
        assignin('base','Am',A);
        assignin('base','R',R_val);
        assignin('base','mr',mr);
        assignin('base','NP',NP_val);

        % Запуск модели trenl

```

```

        out = sim('tren1', 'StopTime', num2str(Ts*Ns),
'ReturnWorkspaceOutputs','on');

% Получение выходных данных из модели
if isfield(out,'simout'), u0 = sum(out.simout); else u0 = 1; end
if isfield(out,'simout1'), ulv = sum(out.simout1); else ulv =
round(rand); end

% Преобразуем в бинарные значения: был сигнал / не было, принято / не
принято
uo(k) = u0 > 0; % Пороговое детектирование переданного сигнала
ul(k) = ulv > 0; % Пороговое детектирование принятого сигнала
end

% Расчёт MSE через xor (среднеквадратичная ошибка передачи)
err_bits = xor(uo, ul); % Вектор ошибок (1 - ошибка, 0 - правильно)
mse_val = sqrt(mean(err_bits.^2)); % Среднеквадратичная ошибка
Y(j) = mse_val; % Сохранение результата

fprintf('MSE(XOR) (NP= %.5f, R= %.3f) = %.6f\n', NP_val, R_val, mse_val);
end

% Коэффициенты регрессии (линейная модель с взаимодействием)
C = X*X.'; % Матрица ковариации
b_ = (C \ (X*Y)).'; % Расчет коэффициентов регрессии (МНК)
disp('Коэффициенты регрессии [b0 b_NP b_R b_NP*R] для MSE:');
disp(b_);

%% Построение графика поверхности через plot3
NP_vals = minf(1):0.001:maxf(1); % Диапазон значений мощности шума NP
R_vals = minf(2):1:maxf(2); % Диапазон значений сопротивления R
[NN, RR] = meshgrid(NP_vals, R_vals); % Сетка для построения поверхности

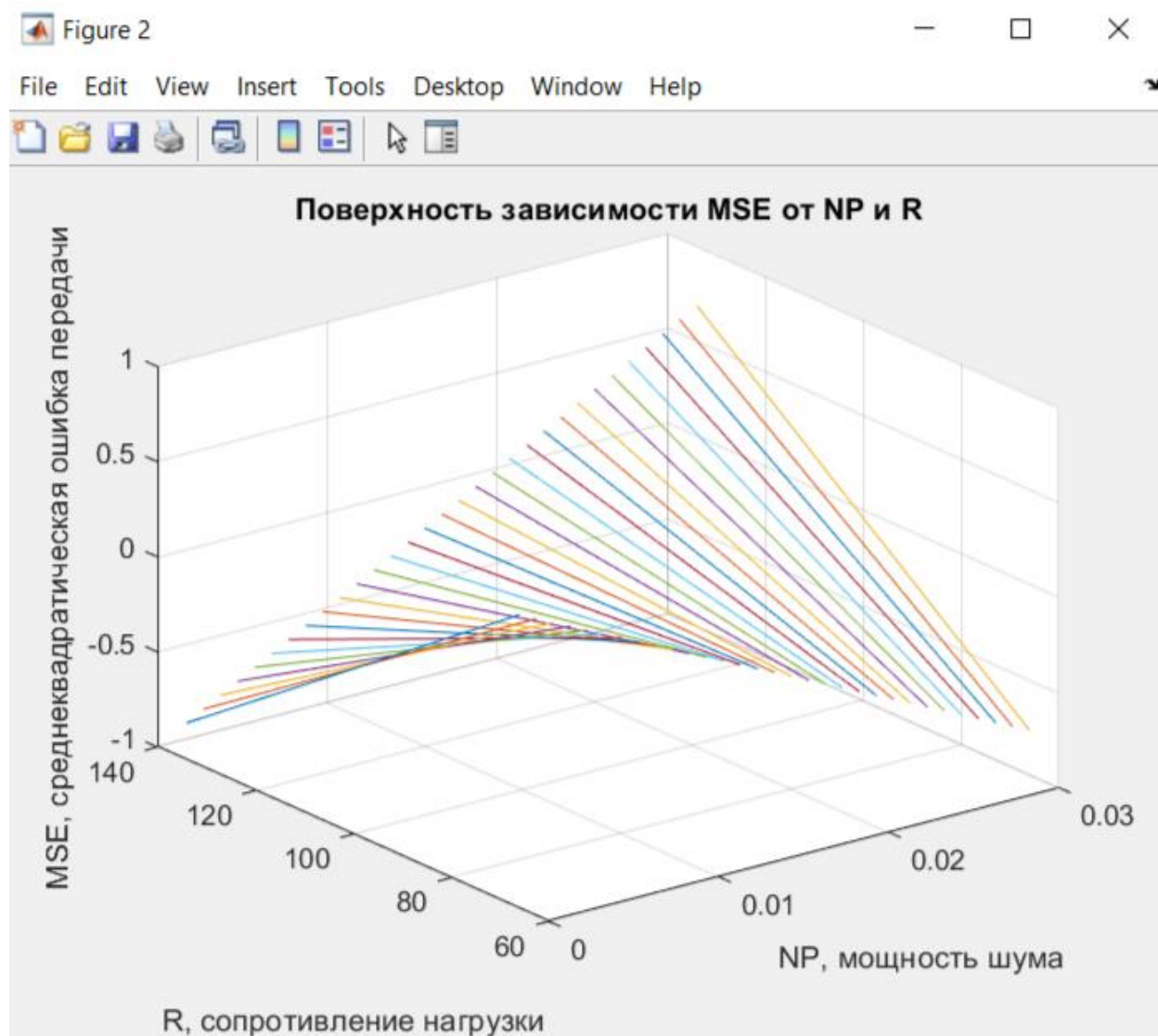
% Нормированные координаты (преобразование к диапазону [-1, +1])
an = 2*(NN - minf(1)) / (maxf(1)-minf(1)) - 1; % Нормировка NP
bn = 2*(RR - minf(2)) / (maxf(2)-minf(2)) - 1; % Нормировка R

% Отклик регрессии для MSE через XOR (предсказание по модели)
Yc = b_(1) + an.*b_(2) + bn.*b_(3) + an.*bn.*b_(4);

% Визуализация через plot3
figure;
plot3(NN, RR, Yc), grid on % Построение 3D графика поверхности
xlabel('NP, мощность шума')
ylabel('R, сопротивление нагрузки')
zlabel('MSE, среднеквадратическая ошибка передачи')
title('Поверхность зависимости MSE от NP и R')

```

Результат:



План в натуральных единицах [NP, R]:

0	66.0000
0	134.0000
0.0300	66.0000
0.0300	134.0000

Расчетное число испытаний NE (для дисперсии) = 17

MSE(XOR) (NP=0.00000, R=66.000) = 0.804400

MSE(XOR) (NP=0.00000, R=134.000) = 0.804400

MSE(XOR) (NP=0.03000, R=66.000) = 0.804400

MSE(XOR) (NP=0.03000, R=134.000) = 0.594089

Коэффициенты регрессии [b0 b_NP b_R b_NP*R] для MSE:

-0.0526	0.0526	0.0526	0.7518
---------	--------	--------	--------

Вывод:

В ходе лабораторной работы была исследована зависимость среднеквадратической ошибки (MSE) от мощности шума (NP) и сопротивления нагрузки (R). Экспериментально установлено, что ошибка передачи значительно снижается только при одновременном воздействии обоих факторов: высокой мощности шума (NP=0.03) и высокого сопротивления нагрузки (R=134). Во всех остальных случаях ошибка оставалась на постоянном высоком уровне. Это подтверждается регрессионной моделью, которая показала сильное влияние взаимодействия NP и R на результат.