Федеральное агентство связи

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Сибирский государственный университет телекоммуникаций и

информатики» (СибГУТИ)

Отчет по

Лабораторной работе №5

по дисциплине «Теория массового обслуживания»

Тема: «Марковские цепи. Исследование эргодических свойств»

Вариант 1

Выполнил:

студент гр. ИА-232

Московских Дмитрий Петрович

Новосибирск 2024

Цель работы: исследовать свойства конечной дискретной, однородной цепи Маркова. Оценить параметры распределения числа коммутаций пакетов в сети.

Подготовка к лабораторной работе:

1. Повторить программирование в системе Mathcad.
2. Изучить свойства дискретной, конечной, однородной цепи Маркова.
3. Повторить определения основных операций с матрицами.
4. Изучить основы функционирования сетей передачи данных с коммутацией пакетов.

Краткая теория:

4.1 Обозначения и расчетные формулы

L – количество узлов в сети с коммутацией пакетов (число состояний цепи Маркова)

$P_{i,j}, \quad i,j=\overline{0,L}$ - маршрутная матрица (матрица перехода однородной цепи Маркова); $P_{i,j}$ - вероятность передачи пакета узлом i в узел j (из лабораторной работы №3).

$P_{i,j}^{(m)}, \quad i,j=\overline{0,L}, \quad m>0$ – вероятность пребывания пакета в узле j после m коммутаций при условии, что пакет поступил в сеть через узел j:

$$P^{(m)} = P^m. \tag{4.1}$$

$f_{i,j}^{(m)}, \quad i,j=\overline{0,L}, \quad m>0$ – вероятность первого перехода пакета в узел j из узла i после m коммутаций:

$$f_{i,j}^{(m)} = P_{i,j}^{(m)} \times \prod_{q=1}^{m-1}(1-P_{i,j}^{(q)}). \tag{4.2}$$

$M_{i,j}, \quad i,j=\overline{0,L}, \quad m>0$ – длина кратчайшего пути перехода пакета в узел j из узла i:

$$M_{i,j} = \min_{f_{i,j}^{(m)}>0} m. \tag{4.3}$$

$\overline{M}_{i,j}, \quad i,j=\overline{0,L}, \quad m>0$ – математическое ожидание длины пути перехода пакета в узел j из узла i:

$$\overline{M}_{i,j} = \sum_{m=1}^{\infty} m \times f_{i,j}^{(m)}. \tag{4.4}$$

$D_{i,j}, \quad i,j=\overline{0,L}, \quad m>0$ – дисперсия длины пути перехода пакета в узел j из узла *i:*

$$\overline{D}_{i,j} = \sum_{m=1}^{\infty} m^2 \times f_{i,j}^{(m)} - \overline{M}_{i,j}^2. \tag{4.5}$$

4.2 Функция для расчета траектории движения пакета по сети

```
MarkovTrajectory(P,N,s) := | E_0 ← s
                           | S ← cols(P) − 1
                           | for i ∈ 0..S
                           |   for j ∈ 1..S
                           |     P_{i,j} ← P_{i,j} + P_{i,j−1}
                           | for i ∈ 1..N
                           |   | r ← rnd(1)
                           |   | E_i ← S
                           |   | for j ∈ 0..S − 1
                           |   |   if r < P_{(E_{i−1}),j}
                           |   |      | E_i ← j
                           |   |      | break
                           | E
```

Здесь P – маршрутная матрица, N – длина траектории, s – начальное состояние цепи Маркова. Принцип стохастической маршрутизации описан в лабораторной работе №3.

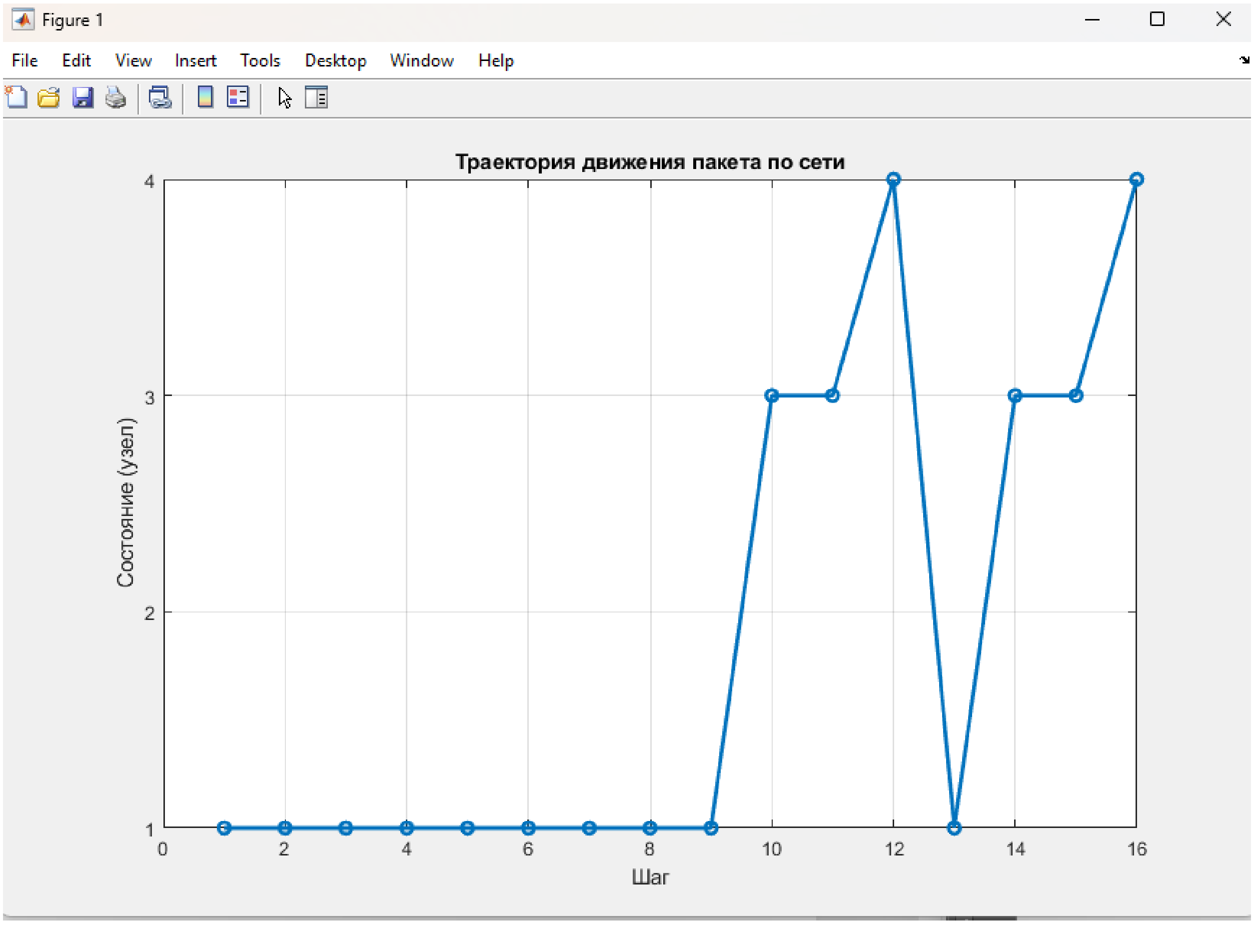
1. Реализация функции MarkovTrajectory

```
function E = MarkovTrajectory(P, N, s)
    E = zeros(1, N);

    E(1) = s;

    S = size(P,1);


    for i = 1:S
        for j = 2:S
            P(i, j) = P(i, j) + P(i, j - 1);
        end
    end
```

```
    for i = 2:N
        r = rand();
        E(i) = S;

        for j = 1:S
            if r < P(E(i-1), j)
                E(i) = j;
                break;
            end
        end
    end
end
```

Рассчёт траекторий:

```
trajectory = MarkovTrajectory(P,N,s);

figure;
    plot(1:N, trajectory, '-o', 'LineWidth', 2);
    title('Траектория движения пакета по сети');
    xlabel('Шаг');
    ylabel('Состояние (узел)');
    grid on;

    yticks(1:size(P, 1));
```



2. Расчёт величин

*вероятность пребывания пакета в узле j после m коммутаций при условии, что пакет поступил в сеть через узел i,

- вероятность первого перехода пакета в узел j из узла i после m коммутаций,
- длину кратчайшего пути перехода пакета в узел j из узла i,
- математическое ожидание длины пути перехода пакета в узел j из узла i,
- дисперсию длины пути перехода пакета в узел j из узла i,

Заданная точность:

```
epsilon = 1e-3;
```

Реализация функций:

```
% Функция для вычисления математического ожидания длины пути
function exp_length = expected_path_length(P, i, j, epsilon)
    m = 1;
    exp_length = 0;
    while true
        prob = first_passage_probability(P, i, j, m);
        exp_length = exp_length + m * prob;
        if prob <= epsilon
            break;
        end
        m = m + 1;
    end
end
```

```
function prob = first_passage_probability(P, i, j, m)
    if m == 1
        prob = P(i, j);
    else
        prob = P(i, j) * (1 - P(i, j))^(m-1);
    end
end

% Функция для нахождения длины кратчайшего пути
function length = shortest_path_length(P, i, j)
    length = inf;
    for m = 1:100
        if first_passage_probability(P, i, j, m) > 0
            length = m;
            break;
        end
    end
end
```

```
% Функция для вычисления дисперсии длины пути
function var_length = variance_path_length(P, i, j, epsilon)
    m = 1;
    exp_length = expected_path_length(P, i, j, epsilon);
```

```
    var_length = 0;
    while true
        prob = first_passage_probability(P, i, j, m);
        var_length = var_length + m^2 * prob;
        if prob <= epsilon
            break;
        end
        m = m + 1;
    end
    var_length = var_length - exp_length^2;
end
```

```
% Вычисления
m = N;  % Количество шагов
probs = zeros(L, L);
shortest_paths = zeros(L, L);
expected_lengths = zeros(L, L);
variances = zeros(L, L);
probs_prebi = 0;

for i = 1:L
    for j = 1:L
        probs(i, j) = first_passage_probability(P, i, j, m);
        shortest_paths(i, j) = shortest_path_length(P, i, j);
        expected_lengths(i, j) = expected_path_length(P, i, j, epsilon);
        variances(i, j) = variance_path_length(P, i, j, epsilon);
        probs_prebi = P.^i;
    end
end

i = 1; j = 3;  % Начальное и конечное состояния
disp(['Вероятность первого перехода: ', num2str(first_passage_probability(P, i, j,
N))]);
disp(['Длина кратчайшего пути: ', num2str(shortest_path_length(P, i, j))]);
disp(['Математическое ожидание длины пути: ', num2str(expected_path_length(P, i, j,
epsilon))]);
disp(['Дисперсия длины пути: ', num2str(variance_path_length(P, i, j, epsilon))]);
disp(['Вероятность пребывания пакета: ', num2str(probs_prebi(i,j))]);
```

Результат

```
Вероятность первого перехода: 0.020589
Длина кратчайшего пути: 1
Математическое ожидание длины пути: 9.52
Дисперсия длины пути: 72.1827
Вероятность пребывания пакета: 0.0001
```

3. Зависимости

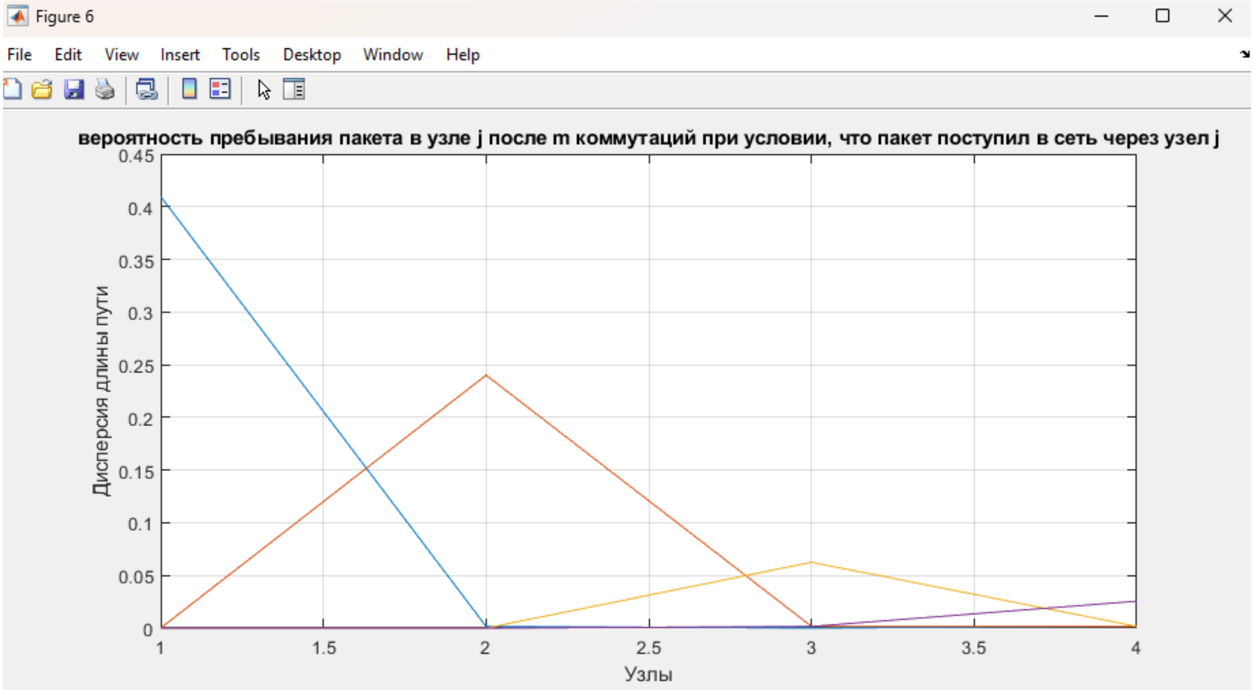


Figure 6
File Edit View Insert Tools Desktop Window Help
вероятность пребывания пакета в узле j после m коммутаций при условии, что пакет поступил в сеть через узел j
Дисперсия длины пути
Узлы
0 0.05 0.1 0.15 0.2 0.25 0.3 0.35 0.4 0.45
1 1.5 2 2.5 3 3.5 4

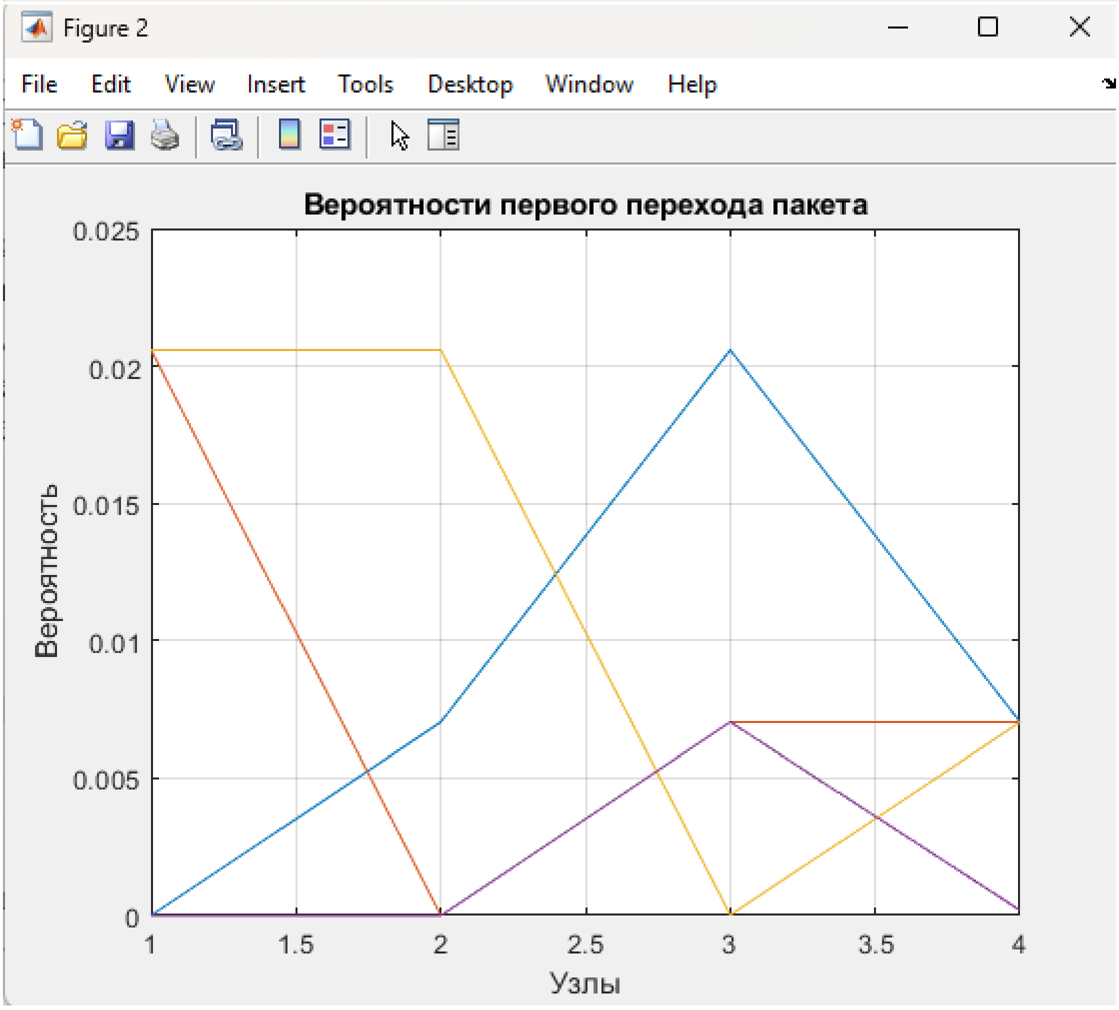


Figure 2
File Edit View Insert Tools Desktop Window Help
Вероятности первого перехода пакета
Вероятность
Узлы
0 0.005 0.01 0.015 0.02 0.025
1 1.5 2 2.5 3 3.5 4

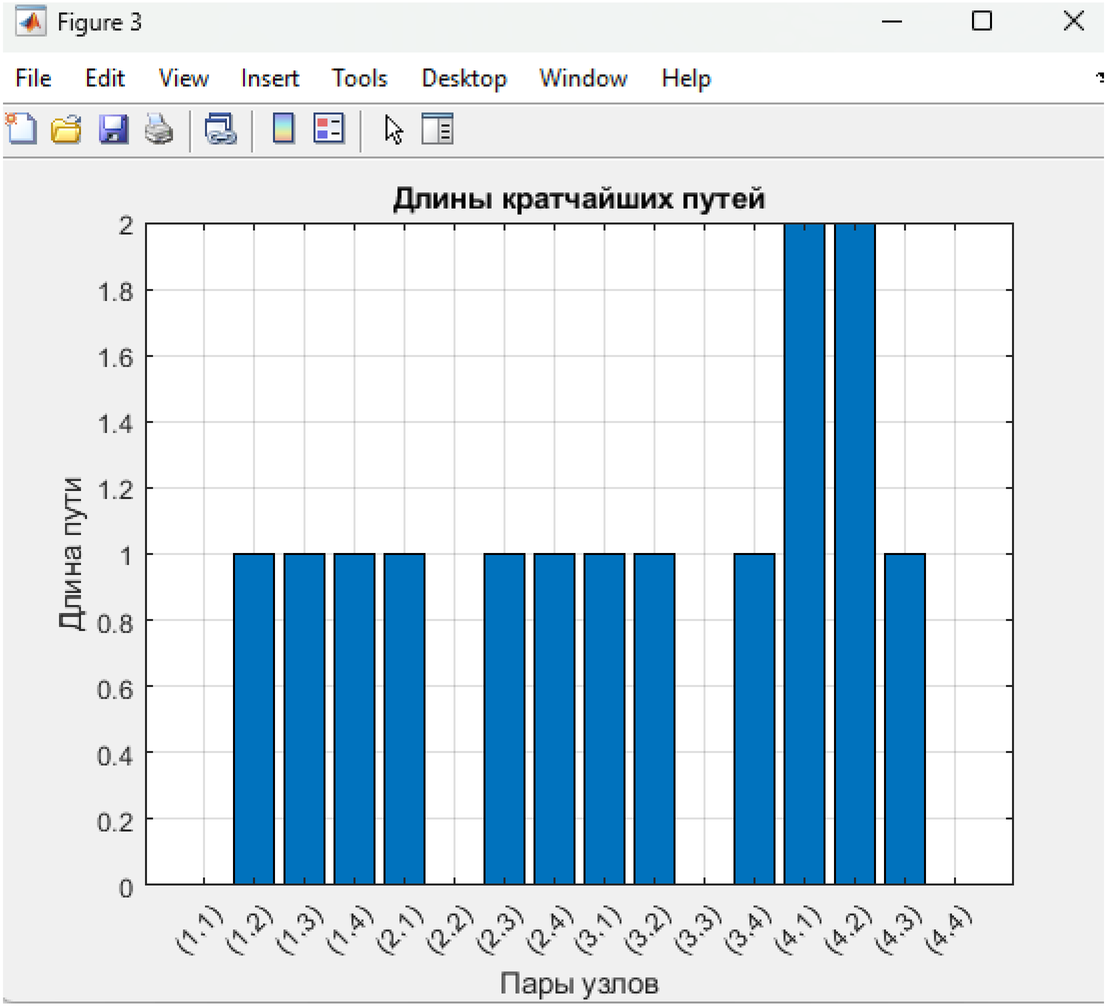


Figure 3
File Edit View Insert Tools Desktop Window Help
Длины кратчайших путей
Длина пути
Пары узлов
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2
(1,1)
(1,2)
(1,3)
(1,4)
(2,1)
(2,2)
(2,3)
(2,4)
(3,1)
(3,2)
(3,3)
(3,4)
(4,1)
(4,2)
(4,3)
(4,4)

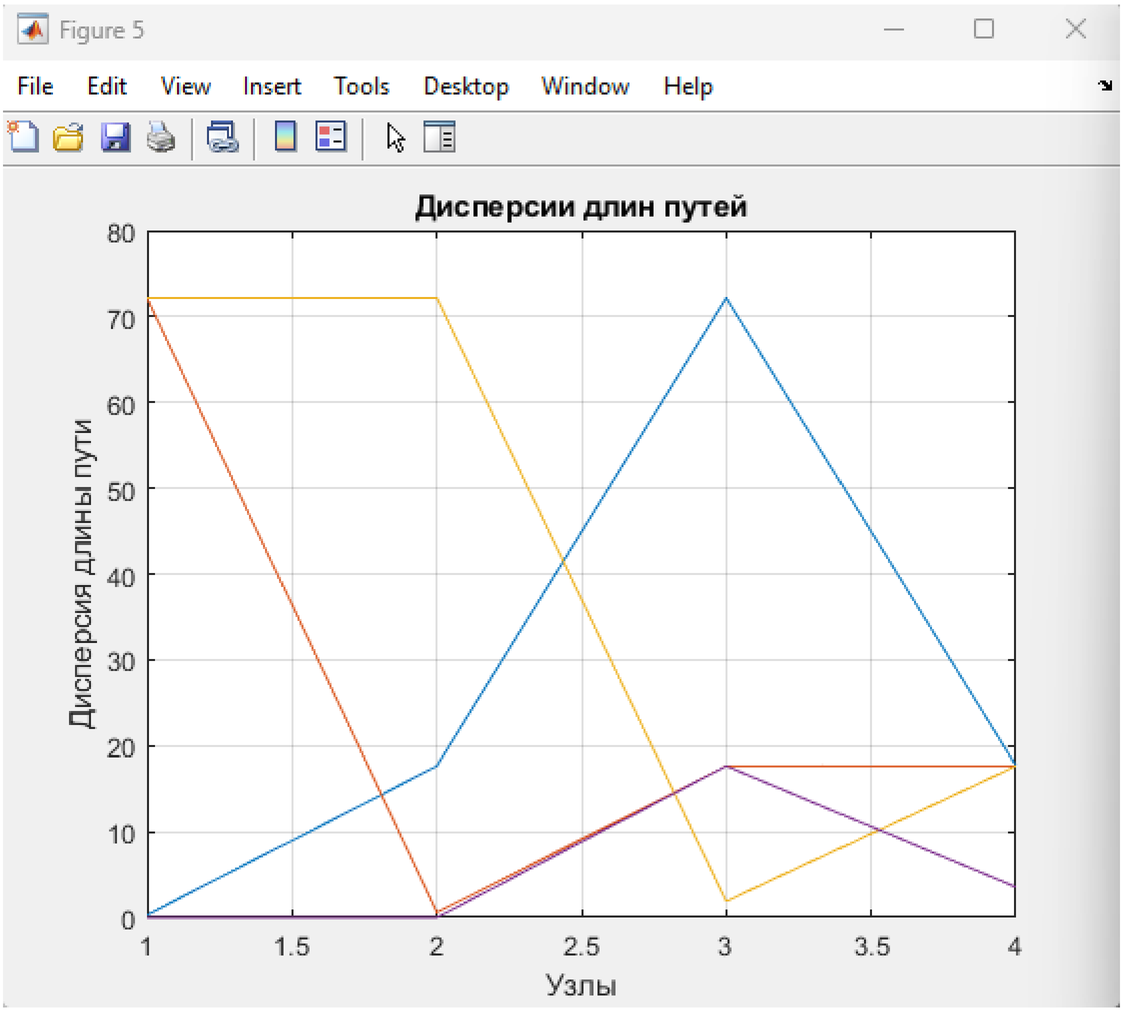


Figure 5
File
Edit
View
Insert
Tools
Desktop
Window
Help
Дисперсии длин путей
Дисперсия длины пути
Узлы
80
70
60
50
40
30
20
10
0
1
1.5
2
2.5
3
3.5
4

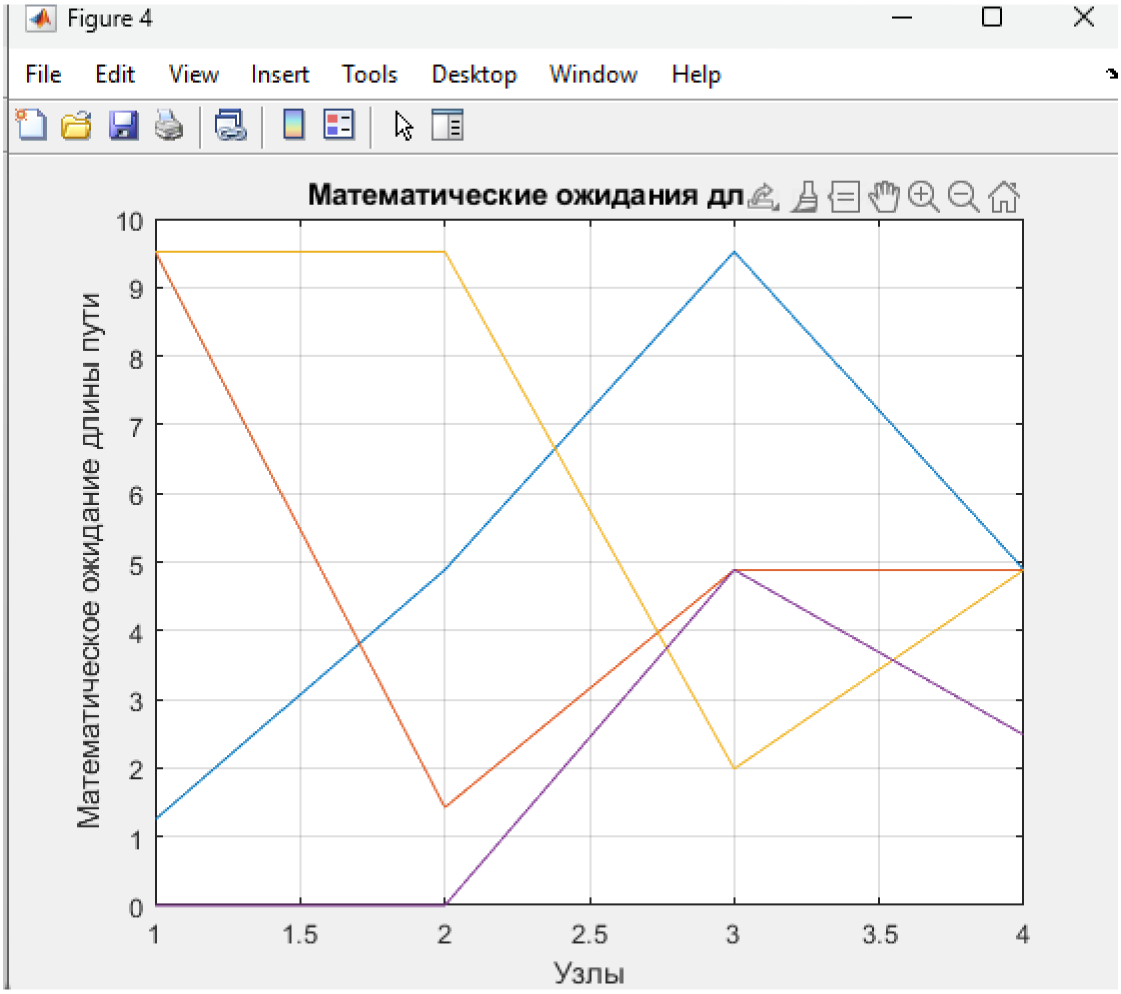


Figure 4
File Edit View Insert Tools Desktop Window Help
Математические ожидания дл
Математическое ожидание длины пути
Узлы
10
9
8
7
6
5
4
3
2
1
0
1
1.5
2
2.5
3
3.5
4

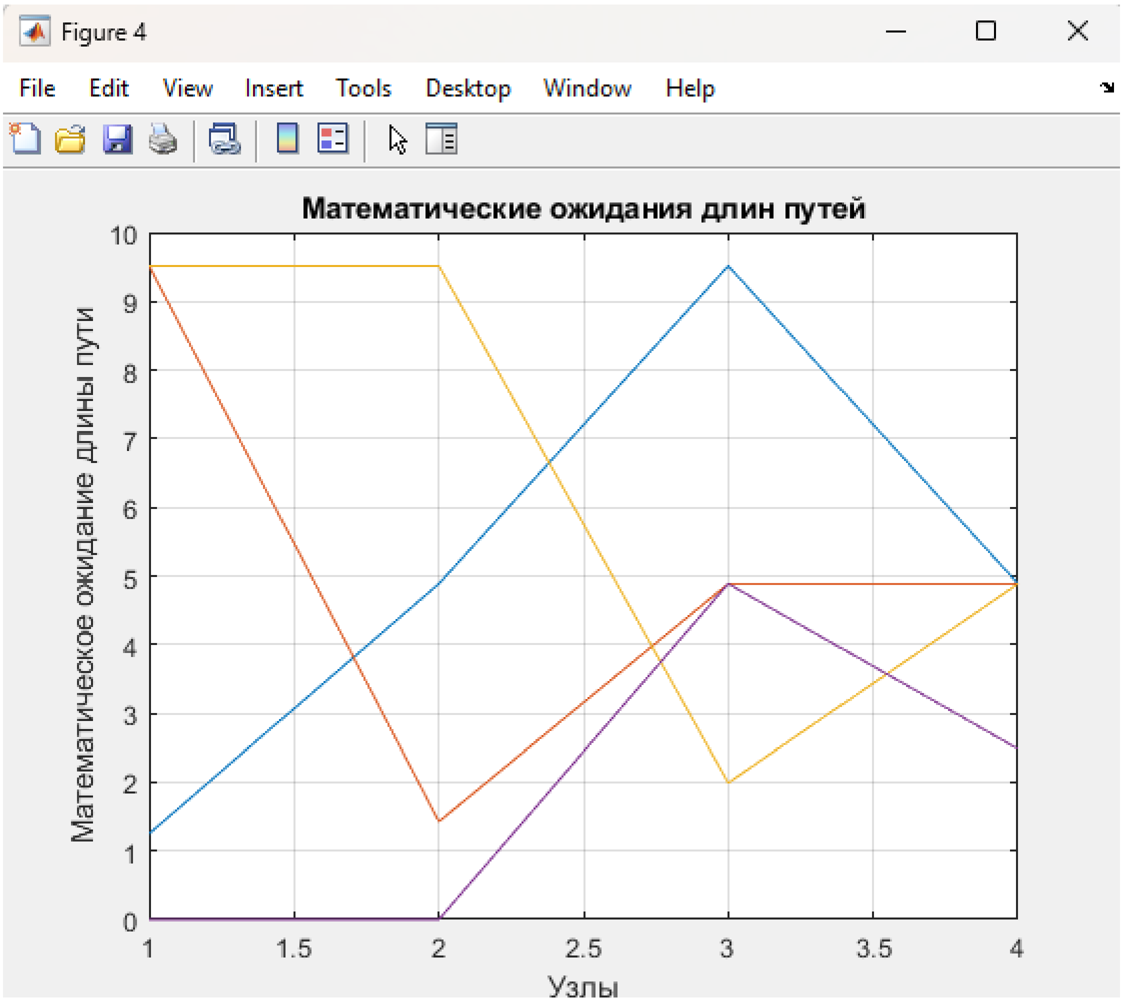
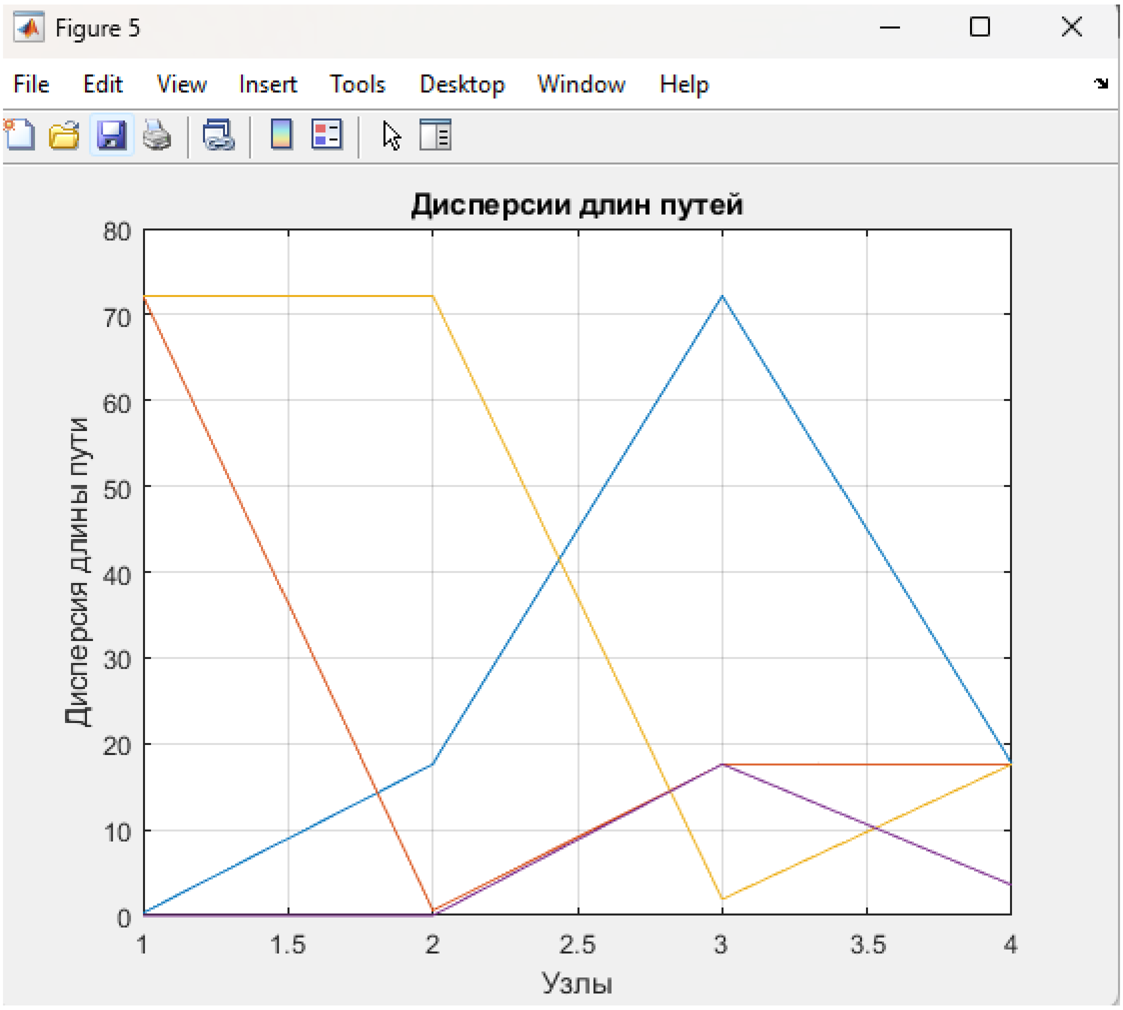


Figure 4
File Edit View Insert Tools Desktop Window Help
Математические ожидания длин путей
Математическое ожидание длины пути
Узлы
0
1
2
3
4
5
6
7
8
9
10
1
1.5
2
2.5
3
3.5
4




```
% Вероятности первого перехода пакета
figure;
plot(probs);
title('Вероятности первого перехода пакета');
xlabel('Узлы');
ylabel('Вероятность');
grid on;

% Создаем матрицу кратчайших путей (у вас уже есть это как shortest_paths)
n = size(shortest_paths, 1);
pairs = cell(n * n, 1);  % массив для пар узлов
index = 1;

% Формируем метки вида (i,j)
for i = 1:n
    for j = 1:n
        pairs{index} = sprintf('(%d,%d)', i, j);
        index = index + 1;
    end
end

% Построение графика
figure;
bar(shortest_paths(:));  % столбчатая диаграмма
title('Длины кратчайших путей');
```

```
xlabel('Пары узлов');
ylabel('Длина пути');
set(gca, 'XTick', 1:(n*n), 'XTickLabel', pairs);  % установка меток оси X
xtickangle(45);  % поворот меток оси X для удобства чтения
grid on;

% Математические ожидания длин путей
figure;
plot(expected_lengths);
title('Математические ожидания длин путей');
xlabel('Узлы');
ylabel('Математическое ожидание длины пути');
grid on;

% Дисперсии длин путей
figure;
plot(variances);
title('Дисперсии длин путей');
xlabel('Узлы');
ylabel('Дисперсия длины пути');
grid on;

% Дисперсии длин путей
figure;
plot(probs_prebi);
title('вероятность пребывания пакета в узле j после m коммутаций при условии, что пакет поступил в сеть через узел j');
xlabel('Узлы');
ylabel('Дисперсия длины пути');
grid on;
```

Контрольные вопросы:

Определение цепи Маркова

Цепь Маркова — это стохастический процесс, в котором вероятность перехода в следующее состояние зависит только от текущего состояния, а не от предыдущих состояний.

Классификация цепей Маркова

Цепи Маркова можно классифицировать по следующим признакам:

- **Дискретные и непрерывные:** по виду времени (дискретное или непрерывное).
- **Конечные и бесконечные:** по количеству состояний.
- **Однородные и неоднородные:** по зависимости переходных вероятностей от времени.

Свойства цепей Маркова

- **Марковское свойство:** будущее состояние зависит только от текущего состояния.
- **Стационарность:** свойства цепи не изменяются со временем.
- **Эргодичность:** в долгосрочной перспективе система посещает все состояния.

Состояния цепи Маркова

Состояния могут быть:

- **Поглощающие:** состояние, из которого невозможно выйти.
- **Переходные:** состояния, из которых возможен переход в другие состояния.

Дискретные и непрерывные цепи Маркова

- **Дискретные цепи Маркова:** переходы происходят в дискретные моменты времени.

- **Непрерывные цепи Маркова:** переходы происходят в непрерывное время.

Вероятности перехода за m шагов

Вероятности перехода за m шагов можно вычислить, возводя матрицу переходных вероятностей в степень m.

Длины кратчайших путей перехода пакета

Кратчайшие пути можно найти с помощью алгоритмов, таких как алгоритм Дейкстры или алгоритм Беллмана-Форда, которые минимизируют суммарное расстояние или стоимость переходов.

Алгоритм функции вычисления траектории цепи Маркова

Алгоритм вычисления траектории может включать:

1. Определение начального состояния.
2. Вычисление вероятностей перехода.
3. Генерация случайного числа для определения следующего состояния.
4. Повторение шагов до достижения конечного состояния.

Общие понятия о вычислительных сетях с коммутацией пакетов и методах маршрутизации в них

В вычислительных сетях с коммутацией пакетов данные разбиваются на пакеты, которые отправляются независимо друг от друга. Методы маршрутизации включают статическую маршрутизацию, где маршруты фиксированы, и динамическую маршрутизацию, где маршруты изменяются в зависимости от текущих условий сети.

Вывод

В ходе рассмотрения цепей Маркова и их применений было выявлено, что эти модели позволяют эффективно анализировать и прогнозировать поведение систем с вероятностными переходами. Такие знания особенно полезны при анализе вычислительных сетей и разработке эффективных алгоритмов маршрутизации, что обеспечивает надежную и оптимальную передачу данных в сетях.