САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

ПЕТРА ВЕЛИКОГО

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Институт компьютерных наук и технологий

Высшая школа интеллектуальных систем и суперкомпьютерных технологий

**РАСЧЁТНОЕ ЗАДАНИЕ**

**«Лабораторная работа 4»**

по дисциплине «Системный анализ и принятие решений»

Выполнил:

студент гр. 5130901/10201

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Ухов А.Д.

(подпись)

Преподаватель:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Сиднев А.Г.

(подпись)

«\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2024 г.

Санкт-Петербург

2024

14. Исследование маршрутизатора глобальной компьютерной сети, обеспечивающего обмен данными по протоколу. Входящие пакеты записываются в буферную память маршрутизатора, обрабатываются процессором и передаются по каналу в соответствующем направлении. Копия принятого пакета хранится в буферной памяти до тех пор, пока не будет получена квитанция о безошибочной доставке пакета адресату. Отсутствие квитанции-подтверждения в течение времени time-out вынуждает маршрутизатор к повторной передаче пакета по тому же каналу в направлении адресата. При отсутствии места в буферной памяти входящие в маршрутизатор пакеты получают отказ.

Требуется построить модель маршрутизатора в форме сети массового обслуживания. При этом следует иметь в виду, что поступивший в маршрутизатор пакет занимает буферную память в течение следующих интервалов времени:

1) ввод в маршрутизатор и запись в буфер со скоростью канала связи,

2) ожидание в очереди на обработку в процессоре,

3) обработка в процессоре,

4) ожидание освобождения канала в направлении выхода из маршрутизатора,

5) передача из маршрутизатора по выходному каналу,

6) ожидание квитанции об удачной доставке с возможным повторением п. 5 (по истечении интервала time-out).

Каждая из отмеченных временных задержек пакета в маршрутизаторе обеспечивается ожиданием и обслуживанием в соответствующем узле сети массового обслуживания. Условия задачи приводят к разомкнутой сети с блокировками, которая может быть заменена стандартной замкнутой сетью массового обслуживания, характеризуемой тем же марковским процессом перехода из состояния в состояние [47].

Определить следующие характеристики маршрутизатора:

а) вероятность отказа в приёме пакета в буферную память,

б) среднее время пребывания пакета в буферной памяти,

в) среднее число пакетов в маршрутизаторе,

г) среднюю интенсивность потока пакетов, занимающих буферную память маршрутизатора,

д) зависимость указанных показателей от размера буферной памяти маршрутизатора.

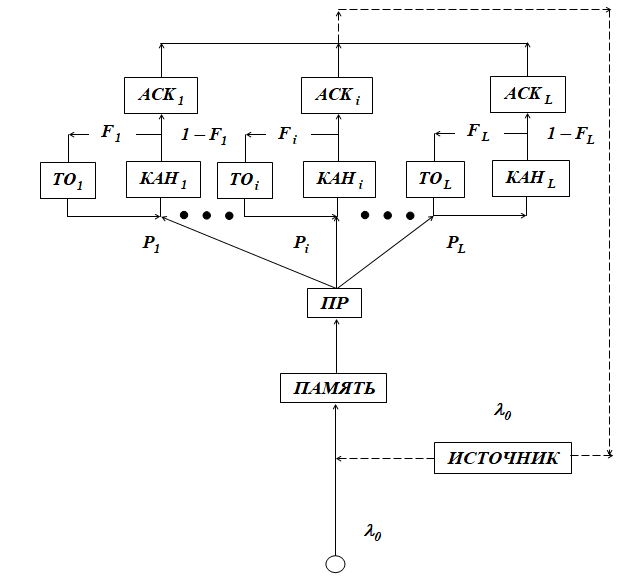
Варианты заданий приведены в табл. 12.20.

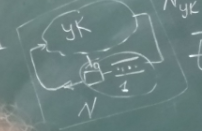
*Примечание.* Предполагается равновероятная передача пакета по любому из **** каналов, среднее время time-out принимается равным , время успешной доставки квитанции равно . Все временные задержки в маршрутизаторе, в том числе интервал time-out, и время доставки квитанции считаются случайными величинами, распределёнными по показательному закону. Входной поток пакетов — простейший.

Таблица 12.21

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № задания | Интенсивность  поступления  пакетов | Размер  буфера | Число каналов ***L*** | Интенсивность  **μ*пер***  передачи  пакетов  по каналу | Интенсивность **μ*пр***  обработки пакета в процессоре | Вероятность  неуспешной передачи пакета по каналу |
|  |  |  |
| 2 | 8 | 25 | 18 | 1,0 | 10 | 0,10 |

Как сказано в задании мы переведем разомкнутую сеть к замкнутой

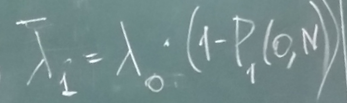




УК – узел коммутации

N=25 (размер буфера) в данном случае - число заявок в сети

Интенсивность поступления пакетов в закрытой системе:



****=8 (интенсивность поступления пакета с источника в разомкнутой системе)

P(0,N) - вероятность отсутствия пакетов на источнике

L = 18 –число каналов

- вероятность передачи пакета по каналу I (Исходит из условия задачи) Для i= 1,2….,L

**** = (Среднее время time out)

TU = 

F = 0,1 (вероятность неуспешной передачи пакета по каналу)

Характеристики узлов

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Источник | Память | Процессор | Канал передачи(КАНi) | ACKi | TOi |
| μ |  |  | μ\_пр | μ\_пер | 1/TU | 1/TO |
| ω | 1 | 1 | 1 |  |  |  |
| Число каналов(m) | 1 | N | 1 | 1 | N | N |

Программа Alab4.m (Входные данные и вызов функции)

N = 25;

P = zeros(N+1, N+1); % создание матрицы из нулей

P(0+1,0+1) = 1; %P(0,0)

l0 = 8;

mu\_pam = l0;

mu\_per = 1; %интенсивность передачи пакетов по каналу

mu\_pr = 10;% интенсивность обработки пакета в процессоре

L=18; %Число каналов по условию задачи

nodes = 3\*(L+1);

mu = zeros(1,nodes);

TO = 2/mu\_per; %среднее время тайм оут

L\_TO = 1/TO; % интенсивность

TU = 0.1 / mu\_per; % время успешной доставки квитанции

L\_TU = 1/TU;% интенсивность

mu\_per\_ar = mu\_per \* ones(1, L);

L\_TO\_ar = L\_TO \* ones(1, L);

L\_TU\_ar = L\_TU \* ones(1, L);

mu = [l0, mu\_pr,mu\_pam,mu\_per\_ar,L\_TU\_ar,L\_TO\_ar ];

m\_mu\_per\_ar = 1 \* ones(1, L);

m\_L\_TO\_ar = N \* ones(1, L);

m\_L\_TU\_ar = N \* ones(1, L);

m = [1,N,1,m\_mu\_per\_ar,m\_L\_TU\_ar,m\_L\_TO\_ar]; %число каналов

%расчет w

P\_k = (1/L) \* ones(1, L);

F = 0.1 \* ones(1, L);

w\_l = zeros(1, L);

w\_m = zeros(1, L);

w\_r = zeros(1, L);

w\_k0 = 1;

w\_l0 = 1;

w\_m0 = 1;

for i = 1:L

w\_l(i) = P\_k(i) / (1-F(i));

w\_m(i) = P\_k(i);

w\_r(i) = F(i) \* P\_k(i) / (1-F(i));

end

w = zeros(1,nodes);

w = [w\_k0,w\_l0,w\_m0, w\_l, w\_m, w\_r];

[t\_array, P\_array] = mhcnCFR(mu,w, m, N, nodes );

P\_0\_N = P\_array{1}(0+1, N+1);

disp(P\_0\_N); % Вывод результата

n1 = N \* (1-P\_0\_N(1));

l1 = l0\*(1-P\_0\_N(1));

N\_uk = N-n1;

t = N\_uk / (l0 \* (1-P\_0\_N(1)));

disp("a)вероятность отказа в приёме пакета в буферную память");

fprintf("P(0,N) =%.2f\n", P\_0\_N(1))

disp("б)среднее время пребывания пакета в буферной памяти");

fprintf("t = %.2f\n", t)

disp("в) среднее число пакетов в маршрутизаторе");

fprintf("N\_uk = %.2f\n", N\_uk)

disp("г) средняя интенсивность потока пакетов, занимающих буферную память маршрутизатора");

fprintf("l1 = %.2f\n", l1)

Программа mhcnCFR.m (рекурсивная процедура)

% Многоканальные однородные замкнутые сети СМО. На выход

% t\_array - массив массивов t(i)(N) формат

% P\_array - массив матриц

%Входы mu - массив мю для каждого узла

%w - массив омега для каждого узла

% m - массив числа каналов для каждого узла

% N - число заявок в сети

% nodes - число узлов

function [t\_array, P\_array] = mhcnCFR(mu,w, m, N, nodes )

w\_sum = sum(w);

t\_array = cell(1, nodes);

P\_array = cell(1, nodes);

P = zeros(N+1, N+1); % создание матрицы из нулей

P(0+1,0+1) = 1; %P(0,0)

for i = 1:nodes

mu\_i = zeros(1,N);

%расчет mu

for n = 1:N

if n < m(i)

mu\_i(n) = n \* mu(i);

else

mu\_i(n) = m(i) \* mu(i);

end

end

t=zeros(1,N);

l1 = zeros(1,N);

for r = 1:N

%шаг 1

for n = 1:r

t(r) = t(r) + n/mu\_i(n) \* P(n-1+1, r-1+1);

end

%шаг 2

l1(r)= r / (w\_sum \* t(r) / w(1));

%шаг 3

sum\_P = 0;

for n = 1:r

%неизвестно какое w писать

P(n+1, r+1) = w(i) \* l1(r) \* P(n-1+1, r-1+1)/(w(1) \* mu\_i(n));

sum\_P = sum\_P + P(n+1,r+1);

end

P(0+1,r+1) = 1-sum\_P;

end

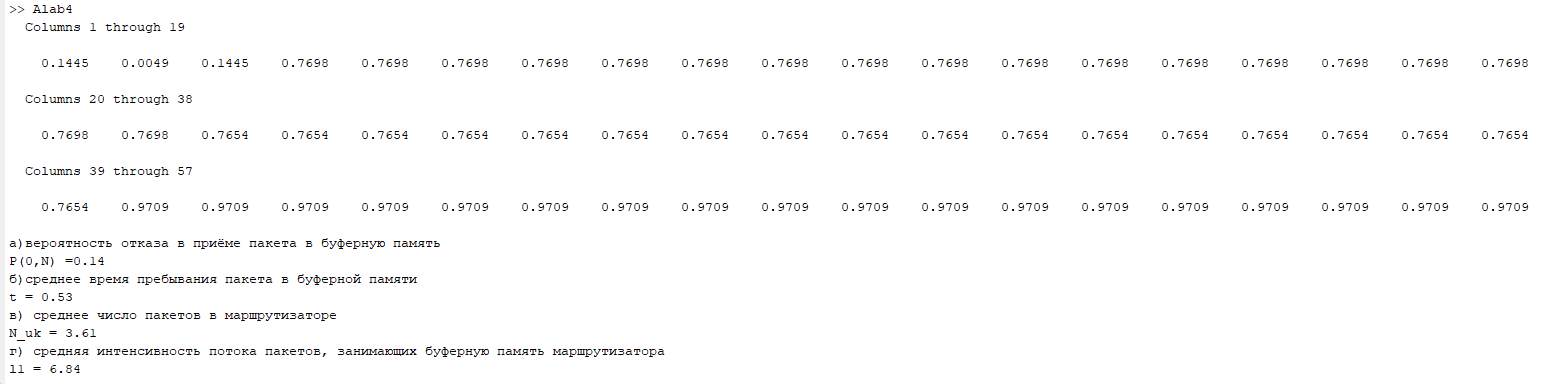
P\_array{i} = P;

t\_array{i} = t;

end

end

Результаты



Графики

д) зависимость указанных показателей от размера буферной памяти маршрутизатора.

max\_bs = 50;

n1 = zeros(1,max\_bs);

l1 = zeros(1,max\_bs);

N\_uk = zeros(1,max\_bs);

t = zeros(1,max\_bs);

P\_0\_N = zeros(1,max\_bs);

for N = 1:max\_bs

P = zeros(N+1, N+1); % создание матрицы из нулей

P(0+1,0+1) = 1; %P(0,0)

l0 = 8;

mu\_pam = l0;

mu\_per = 1; %интенсивность передачи пакетов по каналу

mu\_pr = 10;% интенсивность обработки пакета в процессоре

L=18; %Число каналов по условию задачи

nodes = 3\*(L+1);

mu = zeros(1,nodes);

TO = 2/mu\_per; %среднее время тайм оут

L\_TO = 1/TO; % интенсивность

TU = 0.1 / mu\_per; % время успешной доставки квитанции

L\_TU = 1/TU;% интенсивность

mu\_per\_ar = mu\_per \* ones(1, L);

L\_TO\_ar = L\_TO \* ones(1, L);

L\_TU\_ar = L\_TU \* ones(1, L);

mu = [l0, mu\_pr,mu\_pam,mu\_per\_ar,L\_TU\_ar,L\_TO\_ar ];

m\_mu\_per\_ar = 1 \* ones(1, L);

m\_L\_TO\_ar = N \* ones(1, L);

m\_L\_TU\_ar = N \* ones(1, L);

m = [1,N,1,m\_mu\_per\_ar,m\_L\_TU\_ar,m\_L\_TO\_ar]; %число каналов

%расчет w

P\_k = (1/L) \* ones(1, L);

F = 0.1 \* ones(1, L);

w\_l = zeros(1, L);

w\_m = zeros(1, L);

w\_r = zeros(1, L);

w\_k0 = 1;

w\_l0 = 1;

w\_m0 = 1;

for i = 1:L

w\_l(i) = P\_k(i) / (1-F(i));

w\_m(i) = P\_k(i);

w\_r(i) = F(i) \* P\_k(i) / (1-F(i));

end

w = zeros(1,nodes);

w = [w\_k0,w\_l0,w\_m0, w\_l, w\_m, w\_r];

[t\_array, P\_array] = mhcnCFR(mu,w, m, N, nodes );

%disp(P\_0\_N); % Вывод результата

P\_0\_N(N) = P\_array{1}(0+1, N+1);

n1(N) = N \* (1-P\_0\_N(N));

l1(N) = l0\*(1-P\_0\_N(N));

N\_uk(N) = N-n1(N);

t(N) = N\_uk(N) / (l1(N));

end

N = 1:max\_bs; % Значения N от 1 до 50

% Построение первого графика в первом окне

figure; % Создание нового окна

plot(N, P\_0\_N\_N, 'LineWidth', 2);

xlabel('N');

ylabel('P(0,N)');

title('График P(0,N)');

grid on;

% Построение второго графика во втором окне

figure; % Создание нового окна

plot(N, t, 'LineWidth', 2);

xlabel('N');

ylabel('t');

title('График t');

grid on;

% Построение первого графика в первом окне

figure; % Создание нового окна

plot(N, N\_uk, 'LineWidth', 2);

xlabel('N');

ylabel('N\_uk');

title('График N\_uk');

grid on;

% Построение второго графика во втором окне

figure; % Создание нового окна

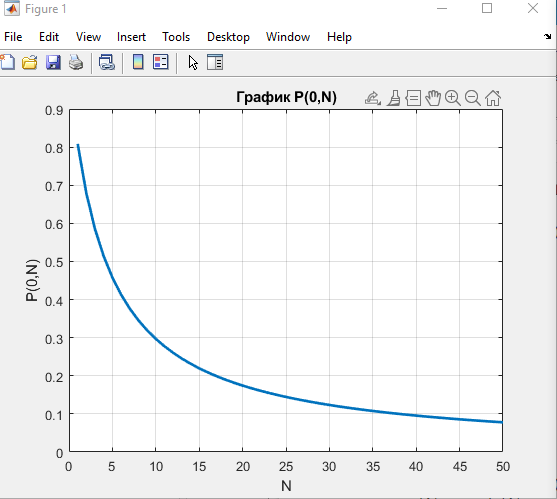
plot(N, l1, 'LineWidth', 2);

xlabel('N');

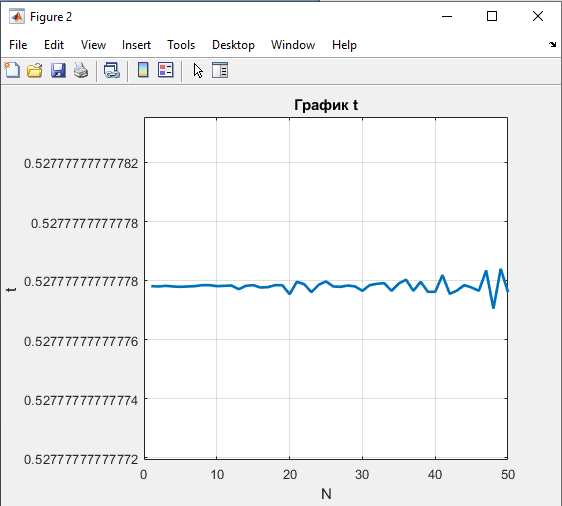
ylabel('l1');

title('График l1');

grid on;

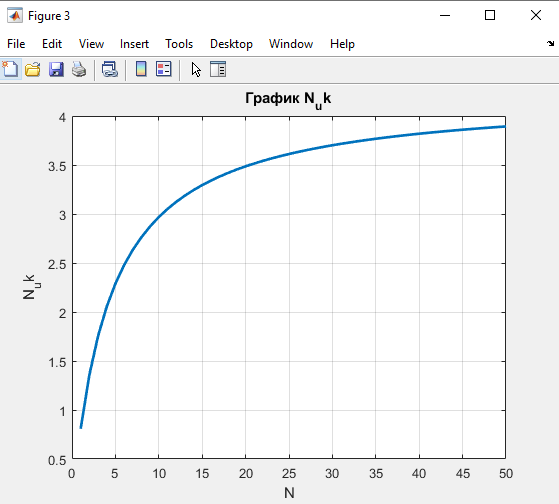
а) вероятность отказа в приёме пакета в буферную память,  


б) среднее время пребывания пакета в буферной памяти,

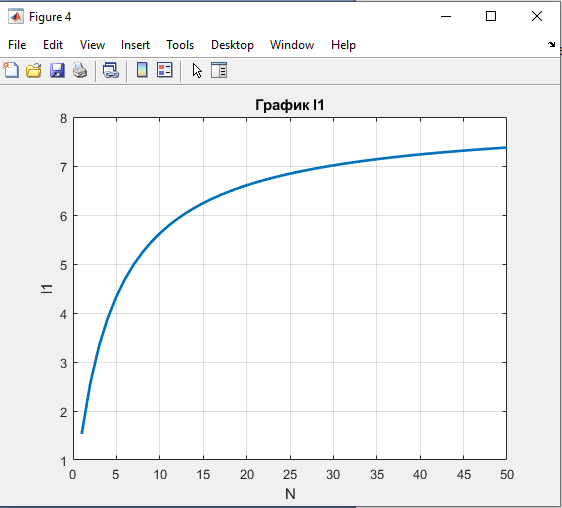


(из-за погрешности в вычислениях)

в) среднее число пакетов в маршрутизаторе,



г) средняя интенсивность потока пакетов, занимающих буферную память маршрутизатора



**Простейший расчет среднего времени пребывания пакета в буферной памяти. Я сложил время обработки в пр+время передачи по каналу связи + время записи в память со скоростью входного потока + время получения квитанции + время нахождения в таймауте (с вероятностью 0.1).**

**1/15+1/0,8+1/9+1/(0,1\*0,8)+0,1/(2\*0,8)=13.99**

**Я не учитывал неизбежные времена ожидания в узлах и посчитал только суммарное время обслуживания пакета во всех узлах сети.**