Беспалов Владимир Михайлович, группа 3-1

Лабораторная работа № 3

**Вариант № 3**

Моделирование систем массового обслуживания (Q-систем) в

Simulink с использованием библиотеки SimEvents

**Цель работы**

Практическое изучение технологий визуального программирования имитационных моделей систем с использованием подсистемы Simulink, построение систем массового обслуживания, а также оценка различных показателей эффективности с помощью библиотеки SimEvents.**Задание**

Междугородный переговорный пункт имеет четыре телефонных аппарата. В среднем за сутки поступает 320 заявок на переговоры. Средняя длительность переговоров составляет 5 мин. Длина очереди не должна превышать 6 абонентов. Потоки заявок и обслуживаний простейшие. Определить характеристики обслуживания переговорного пункта в стационарном режиме: вероятность простоя каналов, вероятность отказа клиенту в обслуживании, среднее число занятых каналов, среднее число заявок в очереди.**Код программы**

clear all;

% Theoretical values;

% http://window.edu.ru/resource/208/29208/files/samiit225.pdf

% queue\_len

m = 6;

% Channel\_num

n = 4;

fprintf('========= Theoretical values =========\n\n');

% Service request per second

lambda = 320 / 24 / 60 / 60; % by default, per second

fprintf('Mean value of calls per second: %f \n', lambda);

% Service amount, per second

mu = 1 / (5 \* 60); % by default, per second

fprintf("Service amount per second: %f \n", mu);

% traffic intensity

rho = lambda / mu;

fprintf('Traffic intensity: %f\n', rho);

p\_0 = 1;

for i=1:n

p\_0 = p\_0 + rho.^i / factorial(i);

end

p\_0 = 1 / (p\_0 + rho.^(n + 1) / (n \* factorial(n)) \* ((1 - rho / n).^m / (1 - rho / n)));

% Idle probability

fprintf('Idle probability: %f\n', p\_0);

% Reject probability

p\_reject = p\_0 \* (rho.^(n + m) / (n.^m \* factorial(n)));

fprintf('Reject probability: %f\n', p\_reject);

% Relative throughput

q = 1 - p\_reject;

fprintf('Relative throughput: %f\n', q);

% Absolute throughput

Q = q \* lambda;

fprintf('Absolute throughput: %f calls per second\n', Q);

% Mean amount of occupied phones

n\_mean = Q / mu;

fprintf('Mean amount of occupied phones: %f\n', n\_mean);

% mean amount of waiting requests in queue

e\_wait = (rho.^(n + 1)) / (n \* factorial(n)) \* p\_0 \* (1 - (rho / n).^m \* (1 + m \* (1 - rho / n))) / (1 - rho / n).^2;

fprintf('mean amount of waiting requests in queue: %f\n\n', e\_wait);

% Experimental values

day\_amount = 10;

fprintf('========= Experimental values, days: %d =========\n\n', day\_amount);

time\_per\_day = 24 \* 3600; % day in seconds

total\_time = time\_per\_day \* day\_amount;

simOut = sim('lab3', total\_time);

generated\_values = get(simOut, 'GeneratedEntities');

fprintf("Total generated: %d, mean generated per day %f\n", max(generated\_values), max(generated\_values) / day\_amount);

call\_time = get(simOut, 'CallTime');

fprintf("Mean call time: %f\n", mean(call\_time));

phone\_ocupancy = get(simOut, 'PhoneOccupancy');

fprintf("Mean phone ocupancy: %f\n", mean(phone\_ocupancy));

p\_0\_exp = nnz(~phone\_ocupancy.Data) / numel(phone\_ocupancy.Data);

fprintf("Idle probability: %f\n", p\_0\_exp);

queue\_ocupancy = get(simOut, 'NumberOfEntitiesInQueue');

fprintf("Mean amount of entities in queue : %f\n", mean(queue\_ocupancy.Data));

processed\_values = get(simOut, 'ProcessedCalls');

fprintf('Reject probability: %f\n', 1 - numel(processed\_values.Data) / numel(generated\_values.Data));

fprintf("Processed all calls: %d\n\n", sum(processed\_values) == sum(generated\_values));

**Результаты выполнения задания**

1. Сравнение теоретических значений с практическими.

|  |  |  |
| --- | --- | --- |
|  | Теоретические значения | Практические значения |
| вероятность простоя каналов (p\_0) | 0.330689 | 0.156138 |
| вероятность отказа клиенту в обслуживании (p\_reject) | 0.000010 | 0.000313 (Потеря последнего клиента вызвана остановкой симуляции) |
| среднее число занятых каналов | 1.111100 | 1.189518 |
| среднее число заявок в очереди | 0.011157 | 0.000000 |

1. Полученные графики:

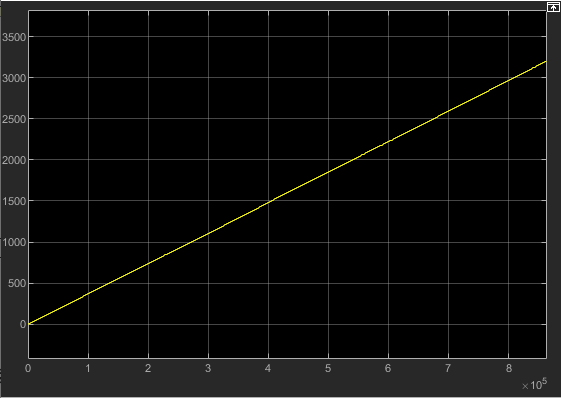


Рисунок 1 – график зависимости числа сгенерированных сущностей от времени (в сек)

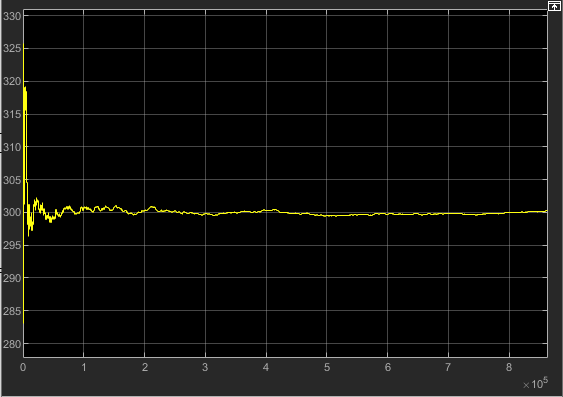


Рисунок 2 – средняя продолжительность звонка в секундах.

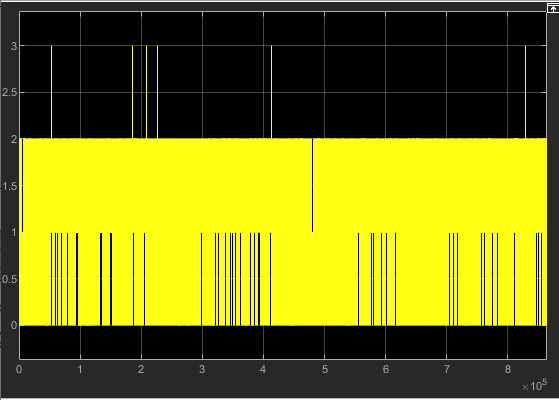


Рисунок 3 – число занятых телефонов в зависимости от времени (в сек)

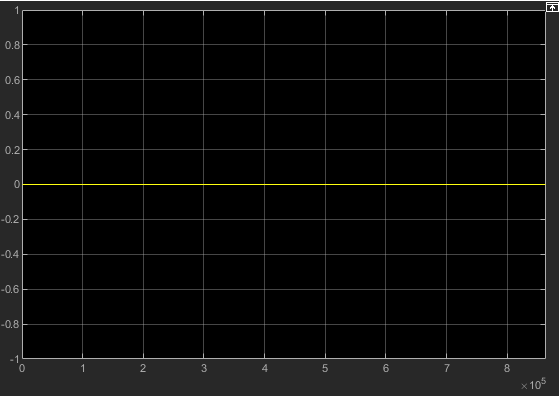


Рисунок 4 – Средняя длина очереди (в сек)

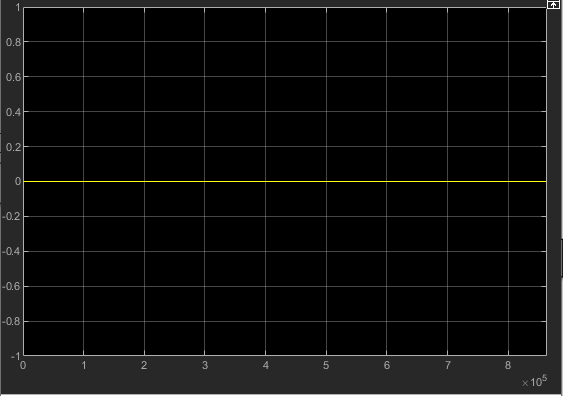


Рисунок 5 – среднее время ожидания в очереди (в сек)

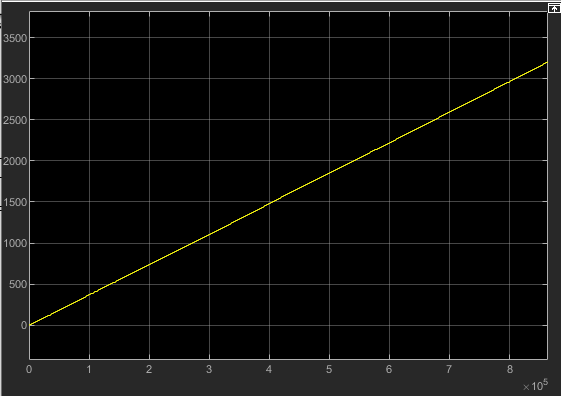


Рисунок 6 – зависимость числа обработанных заявок от времени (сек)

**Вывод:**

Полученная экспериментальная модель, в целом, адекватна теоретической.

Существующие неточности экспериментальной модели, а именно, число отказов появилось ввиду резкой остановки моделирования. В свою очередь, неточности теоретической модели (вероятность отказа, не равная 0) вызвана переносом разрядов и характером хранения числа с плавающей запятой в памяти.

Ввиду высокого числа каналов для заданного числа входящих заявок, необходимость в очереди отсутствует (в случае, если число заявок распространено равномерно)