

Лабораторная работа № 3

Моделирование стохастических процессов

Мугари Абдеррахим

22 февраля 2025

Российский университет дружбы народов, Москва, Россия

Информация

- Анна Владиславовна Королькова
- доцент кафедры прикладной информатики и теории вероятностей РУДН;
- заведующий лабораторией кафедры прикладной информатики и теории вероятностей РУДН (по совместительству);
- программист I кат.
- Российский университет дружбы народов
- korolkova-av@rudn.ru

- Мугари Абдеррахим
- Студент третьего курса
- фундаментальная информатика и информационные технологии
- Российский университет дружбы народов
- 1032215692@rudn.ru
- <https://iragoum.github.io/>



Цель работы

- Цель данной лабораторной работы — изучение моделирования стохастических процессов в системах массового обслуживания (СМО) с использованием математических моделей и компьютерного моделирования в NS-2.

Предварительные сведения.

Система массового обслуживания (СМО) – это математическая модель, описывающая процесс поступления заявок, их обработку и возможные задержки. В данной работе рассматриваются два типа СМО:

- $M|M|1$ – одноканальная СМО с неограниченной очередью.
- $M|M|n|R$ – многоканальная СМО с конечной емкостью буфера.

Для обеих систем входной поток заявок распределен по пуассоновскому закону с интенсивностью (λ),

а время обслуживания заявок распределено по экспоненциальному закону с параметром (μ).

Для описания работы системы используются **уравнения Колмогорова**, которые описывают вероятности нахождения определенного количества заявок в системе в каждый момент времени.

Для M|M|1:

- Стационарное распределение вероятностей выражается формулой:

$$p_i = (1 - \rho)\rho^i, \quad \text{где} \quad \rho = \frac{\lambda}{\mu}$$

Здесь (ρ) – коэффициент загрузки системы.

- Среднее число заявок в системе:

$$N = \frac{\rho}{1 - \rho}$$

- Среднее время пребывания заявки в системе:

$$v = \frac{1}{\mu(1 - \rho)}$$

Выполнение лабораторной работы

- Для моделирования мы используем симулятор NS-2. В коде на Tcl задаются параметры системы:
 - Интенсивность поступления заявок ($\lambda = 30.0$)
 - Средняя скорость обслуживания ($\mu = 33.0$)
 - Размер очереди (100000 для неограниченной системы)
- В коде создаются два узла, соединенные каналом с пропускной способностью 100 Кб/с, и задается очередь DropTail. Для генерации трафика используется агент UDP, который передает пакеты случайного размера.

- Кроме того, реализована функция для мониторинга очереди и вычисления:
 - Теоретической вероятности потери пакетов
 - Средней длины очереди

Создание объекта симулятора

```
set ns [new Simulator]
```

Открытие файла трассировки

```
set tf [open out.tr w]
```

```
$ns trace-all $tf
```

Определение параметров системы

```
set lambda 30.0
```

```
set mu 33.0  
set qsize 100000  
set duration 1000.0
```

Создание узлов и соединения между ними

```
set n1 [$ns node]  
set n2 [$ns node]
```

```
set link [$ns simplex-link $n1 $n2 100kb 0ms DropTail]  
$ns queue-limit $n1 $n2 $qsize
```

Настройка случайных переменных

```
set InterArrivalTime [new RandomVariable/Exponential]  
$InterArrivalTime set avg_ [expr 1/$lambda]
```

```
set pktSize [new RandomVariable/Exponential]  
$pktSize set avg_ [expr 100000.0/(8*$mu)]
```

Создание агентов (источник и приемник)

```
set src [new Agent/UDP]
```

```
$src set packetSize_ 100000
```

```
$ns attach-agent $n1 $src
```

```
set sink [new Agent/Null]
```

```
$ns attach-agent $n2 $sink
```

```
$ns connect $src $sink
```


Мониторинг очереди

```
set qmon [$ns monitor-queue $n1 $n2 [open qm.out w] 0.1]  
$link queue-sample-timeout
```

Функция завершения симуляции

```
proc finish {} {  
    global ns tf  
    $ns flush-trace  
    close $tf  
    exit 0  
}
```

Функция генерации пакетов

```
proc sendpacket {} {  
    global ns src InterArrivalTime pktSize  
    set time [$ns now]  
    $ns at [expr $time + [$InterArrivalTime value]] "sendpacket"  
    set bytes [expr round([$pktSize value])]  
    $src send $bytes  
}
```

Запуск генерации пакетов и завершения симуляции

`$ns at 0.0001 "sendpacket"`

`$ns at $duration "finish"`

Вычисление характеристик системы

`set rho [expr $lambda/$mu]`

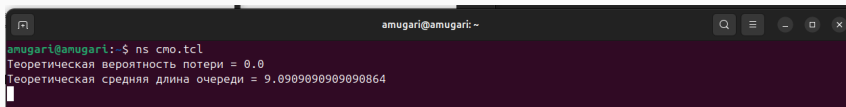
```
set ploss [expr (1-$rho)*pow($rho,$qsize)/(1-pow($rho,($qsize+1)))]  
puts "Теоретическая вероятность потери = $ploss"
```

```
set aveq [expr $rho*$rho/(1-$rho)]  
puts "Теоретическая средняя длина очереди = $aveq"
```

```
# Запуск симуляции
```

```
$ns run
```

- После выполнения кода мы получили
 - Теоретическая вероятность потери = 0.0
 - Средняя длина очереди = 9.09



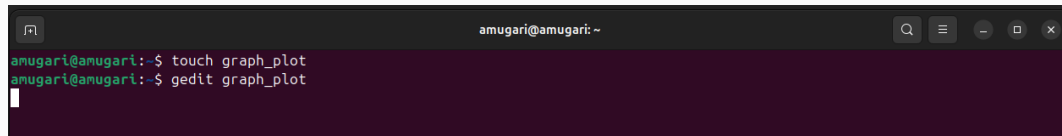
```
amugari@amugari: ~  
amugari@amugari:~$ ns smc.tcl  
Теоретическая вероятность потери = 0.0  
Теоретическая средняя длина очереди = 9.0909090909090864
```

Рис. 1: Анализ результатов

- Очередь никогда не переполняется.
- В среднем в системе ≈ 9 заявок ожидают обработки.
- Система стабильна, но работает с высокой нагрузкой.

Создание файла graph_plot

- мы создали отдельный файл в каталоге проекта с именем **graph_plot**

A terminal window with a dark background and light text. The title bar at the top reads 'amugari@amugari: ~'. On the left side of the title bar are icons for window management. On the right side are icons for search, menu, and window controls. The terminal shows two lines of commands: 'touch graph_plot' and 'gedit graph_plot', both preceded by the prompt 'amugari@amugari:~\$'. A cursor is visible at the end of the second line.

```
amugari@amugari:~$ touch graph_plot
amugari@amugari:~$ gedit graph_plot
```

Рис. 2: Создание файла graph_plot

- Открыли его для редактирования и добавили следующий код

```
#!/usr/bin/gnuplot -persist
# Устанавливаем кодировку и параметры вывода
set encoding utf8
set term pdfcairo font "Arial,9"
# Определяем выходной файл
set out 'qm.pdf'
# Название графика
set title "График средней длины очереди"
```

```
# Настройки линий
```

```
set style line 2
```

```
# Подписи осей
```

```
set xlabel "t"
```

```
set ylabel "Пакеты"
```

```
# Построение графика на основе данных из qm.out
```

```
plot "qm.out" using ($1):($5) with lines title "Размер очереди (в пакетах)",  
      "qm.out" using ($1):($5) smooth csplines title "Приближение сплайном", \  
      "qm.out" using ($1):($5) smooth bezier title "Приближение Безье"
```


Построение графика в Gnuplot

- Потом запустили его.
- После выполнения появится график `qm.pdf`, где можно увидеть, как изменяется длина очереди во времени

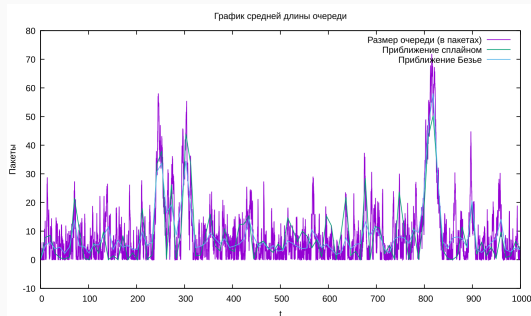


Рис. 3: График средней длины очереди

Выводы

- В данной работе изучены основы моделирования стохастических процессов в системах массового обслуживания с акцентом на модели $M|M|1$ и $M|M|n|R$, а также исследованы уравнения Колмогорова для описания поведения заявок. Реализация моделирования в NS-2 и а результаты, визуализированные через Gnuplot