

Лабораторная работа №3

Имитационное моделирование

Александрова Ульяна Вадимовна

Содержание

1	Цель работы	5
2	Теоретическое введение	6
2.1	М М 1	6
3	Выполнение лабораторной работы	7
3.1	Реализация модели на NS-2 и GNU	7
4	Выводы	13

Список иллюстраций

3.1	Вывод программы	10
3.2	График средней длины очереди	11

Список таблиц

1 Цель работы

Целью данной работы является освоение моделирования стохастических процессов.

2 Теоретическое введение

2.1 М | М | 1

М/М/1 — однолинейная СМО с накопителем бесконечной ёмкости. Поступающий поток заявок — пуассоновский с интенсивностью (λ) . Времена обслуживания заявок — независимые в совокупности случайные величины, распределённые по экспоненциальному закону с параметром (μ) .

3 Выполнение лабораторной работы

3.1 Реализация модели на NS-2 и GNU

Для того, чтобы произвести реализацию данной модели, необходимо создать два файла: исполняемый для построения графика и файл для NS-2. В моем случае это будут `graph_plot` и `lab.tcl` соответственно.

Сначала пропишу код для NS-2 с расчетом теоретической вероятности потери и средней длины очереди (рис. 3.1).

```
# создание объекта Simulator
set ns [new Simulator]

# открытие на запись файла out.tr для регистрации событий
set tf [open out.tr w]
$ns trace-all $tf

# задаём значения параметров системы
set lambda 30.0
set mu 33.0

# размер очереди для M|M|1 (для M|M|1|R: set qsize R)
set qsize 100000

# устанавливаем длительность эксперимента
```

```

set duration 1000.0

# задаём узлы и соединяем их симплексным соединением
# с полосой пропускания 100 Кб/с и задержкой 0 мс,
# очередь с обслуживанием типа DropTail
set n1 [$ns node]
set n2 [$ns node]

# создаём симплексное соединение и сохраняем ссылку на него
set link_ [$ns simplex-link $n1 $n2 100Kb 0ms DropTail]

# наложение ограничения на размер очереди:
$ns queue-limit $n1 $n2 $qsize

# задаём распределения интервалов времени
# поступления пакетов и размера пакетов
set InterArrivalTime [new RandomVariable/Exponential]
$InterArrivalTime set avg_ [expr 1/$lambda]
set pktSize [new RandomVariable/Exponential]
$pktSize set avg_ [expr 100000.0/(8*$mu)]

# задаём агент UDP и присоединяем его к источнику,
# задаём размер пакета
set src [new Agent/UDP]
$src set packetSize_ 100000
$ns attach-agent $n1 $src

# задаём агент-приёмник и присоединяем его
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

# процедура finish закрывает файлы трассировки
proc finish {} {
    global ns tf
    $ns flush-trace
    close $tf
    exit 0
}

# процедура случайного генерирования 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
}

# планировщик событий
$ns at 0.0001 "sendpacket"
$ns at $duration "finish"

# расчет загрузки системы и вероятности потери пакетов

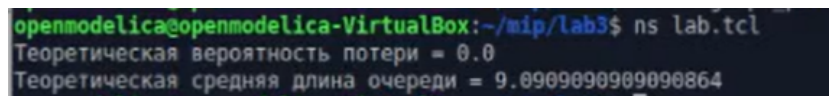
```

```

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 модели
$ns run

```



```

openmodelica@openmodelica-VirtualBox:~/mip/lab3$ ns lab.tcl
Теоретическая вероятность потери = 0.0
Теоретическая средняя длина очереди = 9.8909090909090864

```

Рис. 3.1: Вывод программы

То есть, можно сказать, что пакеты никогда не теряются (что достаточно странно), а средняя длина очереди равна 9.

Теперь напишем код для GNU-plot (рис. 3.2).

```

#!/usr/bin/gnuplot -persist
# задаём текстовую кодировку,
# тип терминала, тип и размер шрифта

set encoding utf8
set term pdfcairo font "Arial,9"

# задаём выходной файл графика

set out 'qm.pdf'

# задаём название графика

```

```

set title "График средней длины очереди"

# задаём стиль линии

set style line 2

# подписи осей графика

set xlabel "t"
set ylabel "Пакеты"

# построение графика, используя значения
# 1-го и 5-го столбцов файла 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 "

```

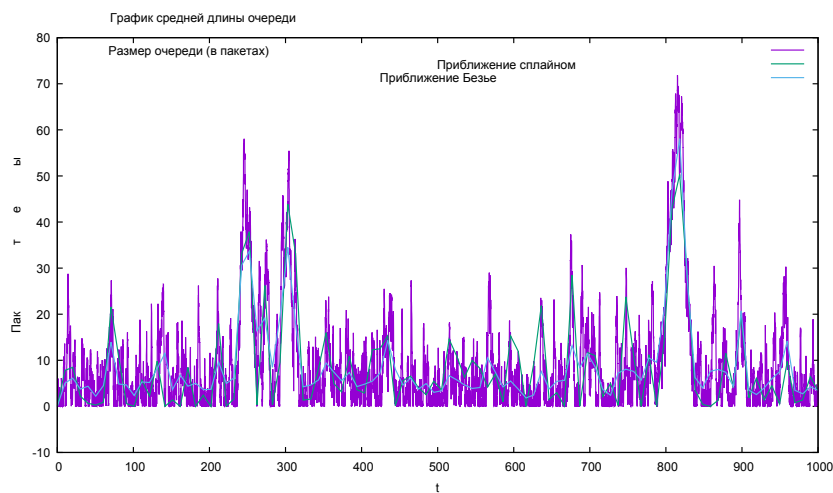


Рис. 3.2: График средней длины очереди

На графике мы можем видеть очередь с течением времени (в пакетах), а так-

же ее приблизегте сплайном и Безье. Мы можем наблюдать скачки, при этом максимальный скачок наблюдается в точке 800 с примерным показателем 70.

4 Выводы

Я научилась моделировать стохастические процессы.