Отчёт по лабораторной работе №3

Дисциплина: Имитационное моделирование

Ганина Таисия Сергеевна, НФИбд-01-22

Содержание

# 1 Цель работы

Исследование работы системы массового обслуживания, моделирование передачи пакетов в сети с использованием симулятора NS-2, анализ зависимости вероятности потерь и длины очереди от интенсивности входного потока.

# 2 Задание

1. Реализация модели на NS-2.
2. График в GNUplot.

# 3 Теоретическое введение

M|M|1 — однолинейная СМО с накопителем бесконечной ёмкости. Поступающий поток заявок — пуассоновский с интенсивностью . Времена обслуживания заявок — независимые в совокупности случайные величины, распределённые по экспоненциальному закону с параметром .

Коэффициент загрузки системы определяется формулой:

где:

* – интенсивность поступления пакетов.
* – интенсивность обработки пакетов.

Вероятность потери пакетов вычисляется как:

Средняя длина очереди:

В данной лабораторной работе с помощью **NS-2** моделируется передача пакетов по каналу связи между двумя узлами. Анализируются потери пакетов и длина очереди при разных параметрах системы.

# 4 Выполнение лабораторной работы

1. Реализация модели на NS-2.

Результат выполнения кода: (рис. [1](#fig:001)). Так как очередь мы задали очень большую (set qsize 100000), а загружается она в среднем на 9 позиций, то и вероятность потери равна 0.0.

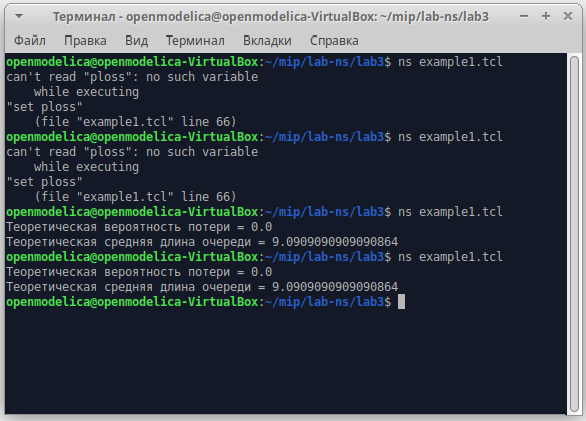


Figure 1: Теоретическая вероятность потери и средняя длина очереди

1. График в GNUplot.

В каталоге с проектом создала отдельный файл graph\_plot (рис. [2](#fig:002)): touch graph\_plot

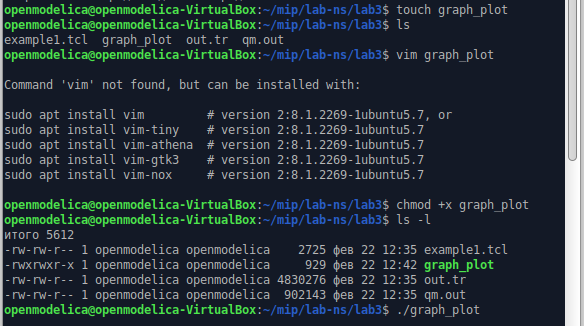


Figure 2: Создание файла, сделала его исполняемым

Открыла его на редактирование и добавила следующий код(рис. [3](#fig:003)), обращая внимание на синтаксис GNUplot.

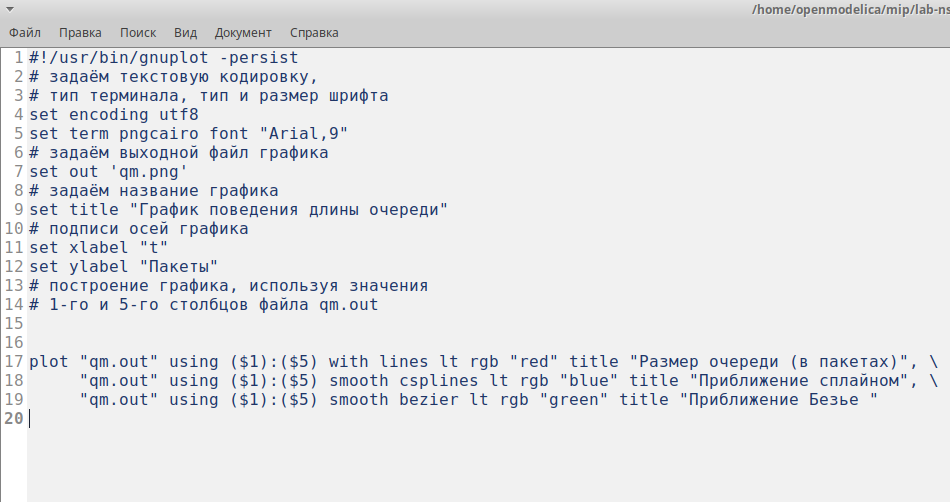


Figure 3: Код в файле

Результат выполнения кода видно на изображении. (рис. [4](#fig:004))

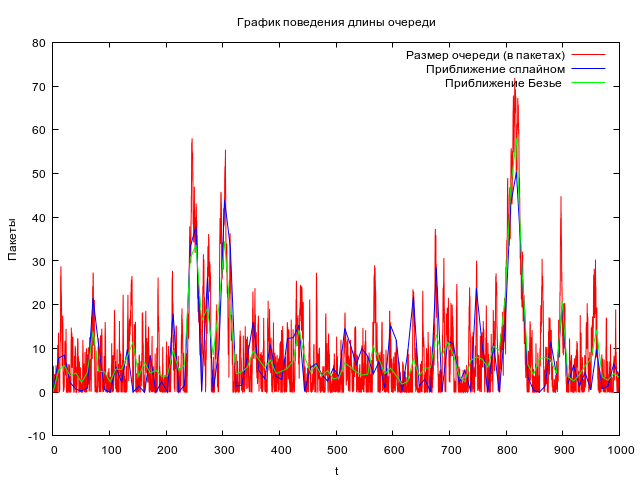


Figure 4: График

На графике изображена динамика изменения длины очереди в зависимости от времени.

* **Оси**:
  + Ось X (горизонтальная): “t” (время). Показывает время в условных единицах.
  + Ось Y (вертикальная): “Пакеты”. Показывает количество пакетов в очереди в данный момент времени.
* **Линии**:
  + Красная линия: “Размер очереди (в пакетах)”. Это фактический размер очереди в каждый момент времени, отражающий мгновенную загрузку очереди.
  + Синяя линия: “Приближение сплайном”. Это сглаженная версия данных о размере очереди, полученная с использованием сплайн-интерполяции. Сплайны помогают выделить общие тенденции и сгладить случайные колебания.
  + Зелёная линия: “Приближение Безье”. Еще один метод сглаживания данных, использующий кривые Безье. Как и сплайны, он помогает увидеть общую картину без резких колебаний.

График показывает, что длина очереди постоянно меняется. Есть периоды с низкой загрузкой (близко к 0), и периоды с высокой загрузкой (пики до 60-70 пакетов). Приближения сплайном и Безье показывают общую тенденцию изменения длины очереди. Они помогают отфильтровать краткосрочные колебания и увидеть более плавное изменение нагрузки.

# 5 Листинги

1. Реализация модели на NS-2.

# 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,  
# очередью с обслуживанием типа DropTail  
set n1 [$ns node]  
set n2 [$ns node]  
  
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

1. График в GNUplot.

#!/usr/bin/gnuplot -persist  
# задаём текстовую кодировку,  
# тип терминала, тип и размер шрифта  
set encoding utf8  
set term pngcairo font "Arial,9"  
# задаём выходной файл графика  
set out 'qm.png'  
# задаём название графика  
set title "График поведения длины очереди"  
# 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   
 lt rgb "red" title "Размер очереди (в пакетаx)", \  
 "qm.out" using ($1):($5) smooth csplines   
 lt rgb "blue" title "Приближение сплайном", \  
 "qm.out" using ($1):($5) smooth bezier   
 lt rgb "green" title "Приближение Безье "

# 6 Выводы

В ходе работы была смоделирована передача пакетов в сети с заданной пропускной способностью и ограниченной очередью. Было показано, что:

* При низкой загрузке системы потери минимальны, а средняя длина очереди мала.

# Список литературы

1. [Руководство к лабораторной работе №3](https://esystem.rudn.ru/mod/resource/view.php?id=1223336)