

Моделирование информационных процессов
Лабораторная работа №3

Ибрагимов Улугбек

Российский университет дружбы народов, Москва, Россия

Информация

- Ибрагимов Улугбек
- НФИбд-02-20
- 1032204510
- Российский Университет Дружбы Народов
- 1032204510@pfur.ru
- <https://github.com/gkwd>

Вводная часть

- Приобрести необходимые в современном научном сообществе навыки моделирования информационных процессов.
- Освоить средство моделирования NS-2.

Цель

- Приобретение навыков моделирования стохастических процессов с помощью средства имитационного моделирования NS-2.
- Приобрести навыки построения графиков в GNUplot.

Выполнение работы

СМО M|M|n|R

```
1
2  # создание объекта Simulator
3  set ns [new Simulator]
4  # открытие на запись файла out.tr для регистрации событий
5  set tf [open out.tr w]
6  $ns trace-all $tf
7  # задаём значения параметров системы
8  set lambda 30.0
9  set mu 33.0
10 # размер очереди для M|M|1 (для M|M|1|R: set qsize R)
11 set qsize 100000
12 # устанавливаем длительность эксперимента
13 set duration 1000.0
14 # задаём узлы и соединяем их симплексным соединением
15 # с полосой пропускания 100 Кб/с и задержкой 0 мс,
16 # очередью с обслуживанием типа DropTail
17 set n1 [$ns node]
18 set n2 [$ns node]
19
20 set link [$ns simplex-link $n1 $n2 100kb 0ms DropTail]
21 # наложение ограничения на размер очереди:
22 $ns queue-limit $n1 $n2 $qsize
23 # задаём распределения интервалов времени
24 # поступления пакетов и размера пакетов
25 set InterArrivalTime [new RandomVariable/Exponential]
26 $InterArrivalTime set avg_ [expr 1/$lambda]
27 set pktSize [new RandomVariable/Exponential]
28 $pktSize set avg_ [expr 100000.0/(8*$mu)]
29 # задаём агент UDP и присоединяем его к источнику,
30 # задаём размер пакета
31 set src [new Agent/UDP]
32 $src set packetSize_ 100000
33 $ns attach-agent $n1 $src
```

```
34 # задаём агент-приёмник и присоединяем его
35 set sink [new Agent/Null]
36 $ns attach-agent $n2 $sink
37 $ns connect $src $sink
38 # мониторинг очереди
39 set qmon [$ns monitor-queue $n1 $n2 [open qm.out w] 0.1]
40 $link queue-sample-timeout
41 # процедура finish закрывает файлы трассировки
42 proc finish {} {
43     global ns tf
44     $ns flush-trace
45     close $tf
46     exit 0
47 }
48 # процедура случайного генерирования пакетов
49 proc sendpacket {} {
50     global ns src InterArrivalTime pktSize
51     set time [$ns now]
52     $ns at [expr $time + [$InterArrivalTime value]] "sendpacket"
53     set bytes [expr round ([$pktSize value])]
54     $src send $bytes
55 }
56 # планировщик событий
57 $ns at 0.0001 "sendpacket"
58 $ns at $duration "finish"
59 # расчет загрузки системы и вероятности потери пакетов
60 set rho [expr $lambda/$mu]
61 set ploss [expr (1-$rho)*pow($rho,$qsize)/(1-pow($rho,($qsize+1))
62 puts "Теоретическая вероятность потери = $ploss"
63 set aveq [expr $rho*$rho/(1-$rho)]
64 puts "Теоретическая средняя длина очереди = $aveq"
65 # запуск модели
66 $ns run
```


GNUplot

```
1  #!/usr/bin/gnuplot -persist
2  # задаём текстовую кодировку,
3  # тип терминала, тип и размер шрифта
4  set encoding utf8
5  set term pdfcairo font "Arial,9"
6  # задаём выходной файл графика
7  set out 'qm.pdf'
8  # задаём название графика
9  set title "График средней длины очереди"
10 # задаём стиль линии
11 set style line 2
12 # подписи осей графика
13 set xlabel "t"
14 set ylabel "Пакеты"
15 # построение графика, используя значения
16 # 1-го и 5-го столбцов файла qm.out
17 plot "qm.out" using ($1):($5) with lines title "Размер очереди (в пакетах)", \
18      "qm.out" using ($1):($5) smooth csplines title " Приближение сплайном ", \
19      "qm.out" using ($1):($5) smooth bezier title " Приближение Безье "
```

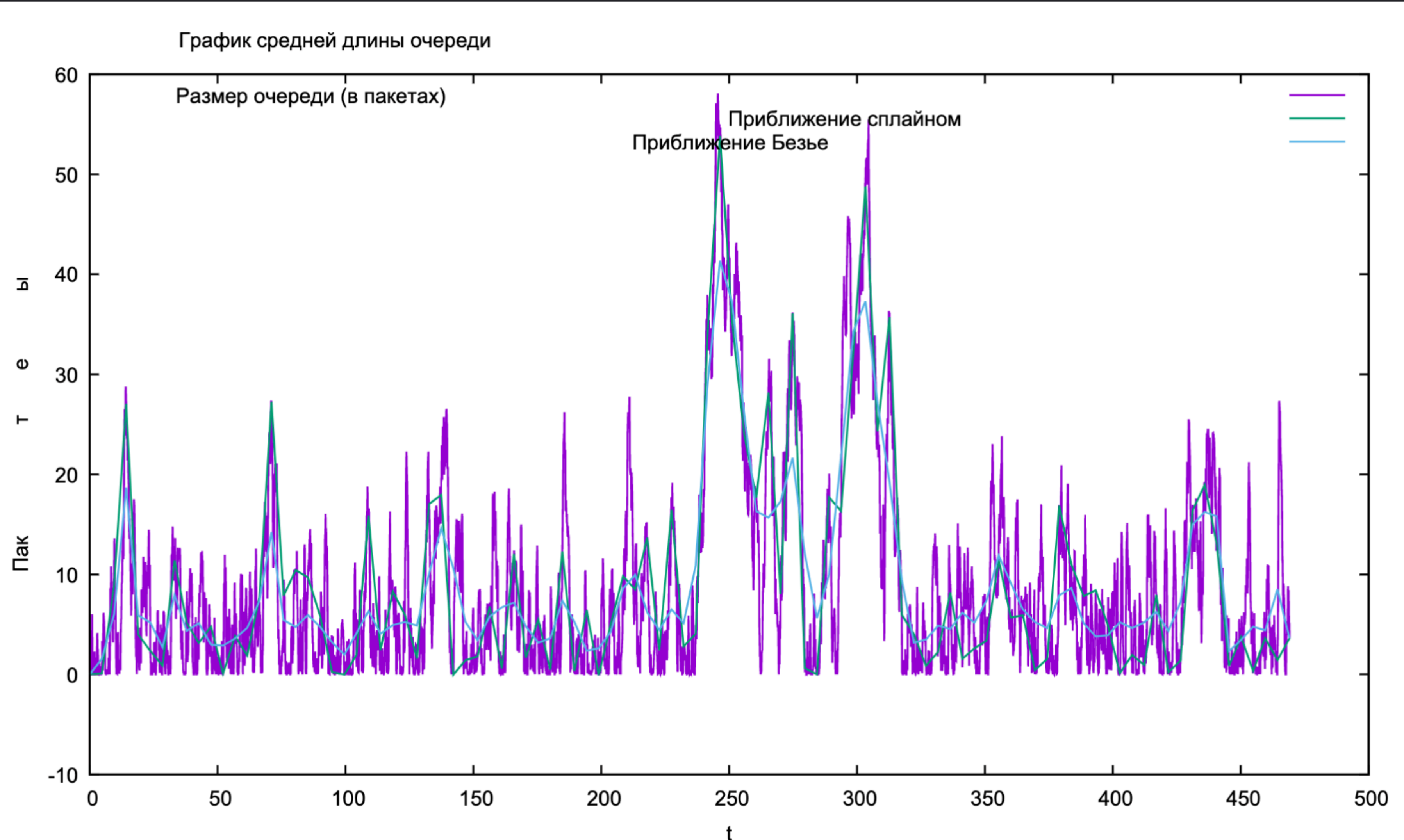


Рис. 1. График поведения длины очереди

Результаты

- В результате выполнения работы, были получены и улучшены практические навыки моделирования на NS-2.
- Произведено моделирование стохастических процессов.
- Приобретены навыки построения графиков при помощи GnuPlot