

# **Отчет по лабораторной работе №4**

**Задание для самостоятельного выполнения**

Ильин Никита Евгеньевич

# Содержание

1	Цель работы	5
2	Задание	6
3	Ход работы	8
4	Выводы	18
5	Библиография	19

# List of Figures

3.1	Рис 1. Код модели(1) . . . . .	9
3.2	Рис 2. Код модели(2) . . . . .	10
3.3	Рис 3. Код для построения графиков . . . . .	11
3.4	Рис 4. Модель сети ns2 . . . . .	12
3.5	Рис 5. Изменение размера длины очереди и средней длины очереди на линке (R1–R2) . . . . .	13
3.6	Рис 6. Изменение размера окна TCP на линке 1-го источника . . . . .	14
3.7	Рис 7. Изменение размера длины очереди и средней длины очереди на линке (R1–R2) . . . . .	15
3.8	Рис 8. Изменение размера средней длины очереди на линке (R1–R2) . . . . .	15
3.9	Рис 9. Изменение размера длины очереди на линке (R1–R2) . . . . .	16
3.10	Рис 10. Изменение размера окна TCP на линке 1-го источника . . . . .	16
3.11	Рис 11. Изменение размера окна TCP на всех источниках . . . . .	17

## List of Tables

# 1 Цель работы

Построить модель сети, и построить графики на ее основе, используя Xgraph и GNUpot.

## 2 Задание

1. Для приведённой схемы разработать имитационную модель в пакете NS-2.

Сеть имеет следующий вид:

- сеть состоит из  $N$  TCP-источников,  $N$  TCP-приёмников, двух маршрутизаторов  $R1$  и  $R2$  между источниками и приёмниками ( $N$  — не менее 20);

- между TCP-источниками и первым маршрутизатором установлены дуплексные соединения с пропускной способностью 100 Мбит/с и задержкой 20 мс очередью типа DropTail;

- между TCP-приёмниками и вторым маршрутизатором установлены дуплексные соединения с пропускной способностью 100 Мбит/с и задержкой 20 мс очередью типа DropTail;

- между маршрутизаторами установлено симплексное соединение ( $R1-R2$ ) с пропускной способностью 20 Мбит/с и задержкой 15 мс очередью типа RED, размером буфера 300 пакетов; в обратную сторону — симплексное соединение ( $R2-R1$ ) с пропускной способностью 15 Мбит/с и задержкой 20 мс очередью типа DropTail;

- данные передаются по протоколу FTP поверх TCP Reno;

- параметры алгоритма RED:  $q_{min} = 75$ ,  $q_{max} = 150$ ,  $q_w = 0,002$ ,  $p_{max} = 0.1$ ;

- максимальный размер TCP-окна 32; размер передаваемого пакет

2. Построить график изменения размера окна TCP (в Xgraph и в GNUPlot);

3. Построить график изменения длины очереди и средней длины очереди на первом маршрутизаторе.

4. Оформить отчёт о выполненной работе.

## 3 Ход работы

1. Создаю файл .tcl:

```
touch lab04_1.tcl
```

Записываю в него описание модели по заданию:



```

1 #создаю симулятор
2 set ns [new Simulator]
3 set nf [open out.nam w]
4 $ns namtrace-all $nf
5 # открываю на запись файл трассировки
6 set f [open out.tr w]
7 $ns trace-all $f
8
9 set N 20
10 set r1 [$ns node]
11 set r2 [$ns node]
12
13 $ns simplex-link $r1 $r2 20Mb 15ms RED
14 $ns queue-limit $r1 $r2 300
15 $ns simplex-link $r2 $r1 15Mb 20ms DropTail
16
17 set redq [[$ns link $r1 $r2] queue]
18 $redq set tresh_ 75
19 $redq set maxtresh_ 150
20 $redq set q_weight_ 0.002
21 $redq set lineterm_ 10
22
23 set tchan_ [open all.q w]
24 $redq trace curq_
25 $redq trace ave_
26 $redq attach $tchan_
27
28 for {set i 0} {$i < $N} {incr i} {
29     set n1($i) [$ns node]
30     $ns duplex-link $n1($i) $r1 100Mb 20ms DropTail
31     set n2($i) [$ns node]
32     $ns duplex-link $n2($i) $r2 100Mb 20ms DropTail
33     set tcp($i) [$ns create-connection TCP/Reno $n1($i) TCPSink $n2($i) 1]
34     $tcp($i) set window_ 32
35     set ftp($i) [$tcp($i) attach-source FTP]
36     $tcp($i) set class_ 1
37 }
38
39 $ns color 1 Red
40
41 # Мониторинг размера окна TCP:
42 set windowVsTimeall [open windall w]
43 set windowVsTime [open wind w]
44 set qmon [$ns monitor-queue $r1 $r2 [open qm($i).out w] 0.1];
45 [$ns link $r1 $r2] queue-sample-timeout;
46
47 proc plotWindow {tcpSource file} {
48     global ns
49     set time 0.01
50     set now [$ns now]
51     set cwnd [$tcpSource set cwnd_]
52     puts $file "$now $cwnd"
53     $ns at [expr $now+$time] "plotWindow $tcpSource $file"
54 }
55 for {set i 0} {$i < $N} {incr i} {
56     $ns at 0.0 "$ftp($i) start"
57     $ns at 0.0 "plotWindow $tcp($i) $windowVsTimeall"
58 }

```

Figure 3.1: Рис 1. Код модели(1)

```

58 }
59 proc finish {} {
60     global ns f nf
61     $ns flush-trace
62     close $f
63     close $nf
64     global tchan_
65     # подключение кода AWK:
66     set awkCode {
67     {
68         if ($1 == "Q" && NF>2) {
69             print $2, $3 >> "temp.q";
70             set end $2
71         }
72         else if ($1 == "a" && NF>2) {
73             print $2, $3 >> "temp.a";
74         }
75     }
76 }
77
78 set f [open temp.queue w]
79 puts $f "TitleText: Изменение размера окна tcp"
80
81 if { [info exists tchan_] } {
82     close $tchan_
83 }
84
85 exec rm -f temp.q temp.a
86 exec touch temp.a temp.q
87
88 # выполнение кода AWK
89 exec awk $awkCode all.q
90 puts $f "\"queue
91 exec cat temp.q >@ $f
92 puts $f \"n\"ave_queue
93 exec cat temp.a >@ $f
94 close $f
95
96 # графики окна TCP и очередей
97 exec xgraph -bb -tk -x time -t "wind(1)" wind &
98 exec xgraph -bb -tk -x time -t "wind(all)" windall &
99 exec xgraph -bb -tk -x "X line" time -y "Y line" queue temp.queue &
100 exec nam out.nam &
101 exit 0
102 }
103
104 $ns at 0.0 "plotWindow $tcp(0) $windowVsTime"
105 $ns at 20.0 "finish"
106 # запуск модели
107 $ns run

```

Figure 3.2: Рис 2. Код модели(2)

2. Создаю новый файл:

```
touch graph
```

И записываю в него код для построения графиков:

```

1 #!/usr/bin/gnuplot -persist
2
3 # задаём текстовую кодировку,
4 # тип терминала, тип и размер шрифта
5 set encoding utf8
6 set term pdfcairo font "Arial,9"
7 # задаём выходной файл графика
8 set out 'qm.pdf'
9
10 # задаём название графика
11 set title "График средней длины очереди"
12 # задаём стиль линии
13 set style line 2
14
15 # оси
16 set xlabel "t"
17 set ylabel "Пакеты"
18
19 set title "средняя длина очереди"
20 plot "temp.q" using ($1):($2) with lines title "Размер очереди",\
21 "temp.a" using ($1):($2) with lines title "Размер очереди (средний)"
22
23 plot "temp.q" using ($1):($2) with lines title "Размер очереди (пакет)"
24
25 set title "длина очереди"
26 plot "temp.a" using ($1):($2) with lines title "Размер очереди"
27
28 # задаём название графика
29 set title "График окна TCP"
30 # задаём стиль линии
31 set style line 2
32
33 # оси
34 set xlabel "t"
35 set ylabel "window"
36
37 plot "wind" using ($1):($2) with lines title "Размер окна"
38 set title "размер окна для всех источников"
39
40 plot "windall" using 1:2 with line title "Размер окна на всех источниках"

```

Figure 3.3: Рис 3. Код для построения графиков

Делаю файл исполняемым:

```
chmod +x graph
```

3. Запускаю файл для построения графиков:

```
ns lab03.tcl
```

Получаю модель сети в ns2:

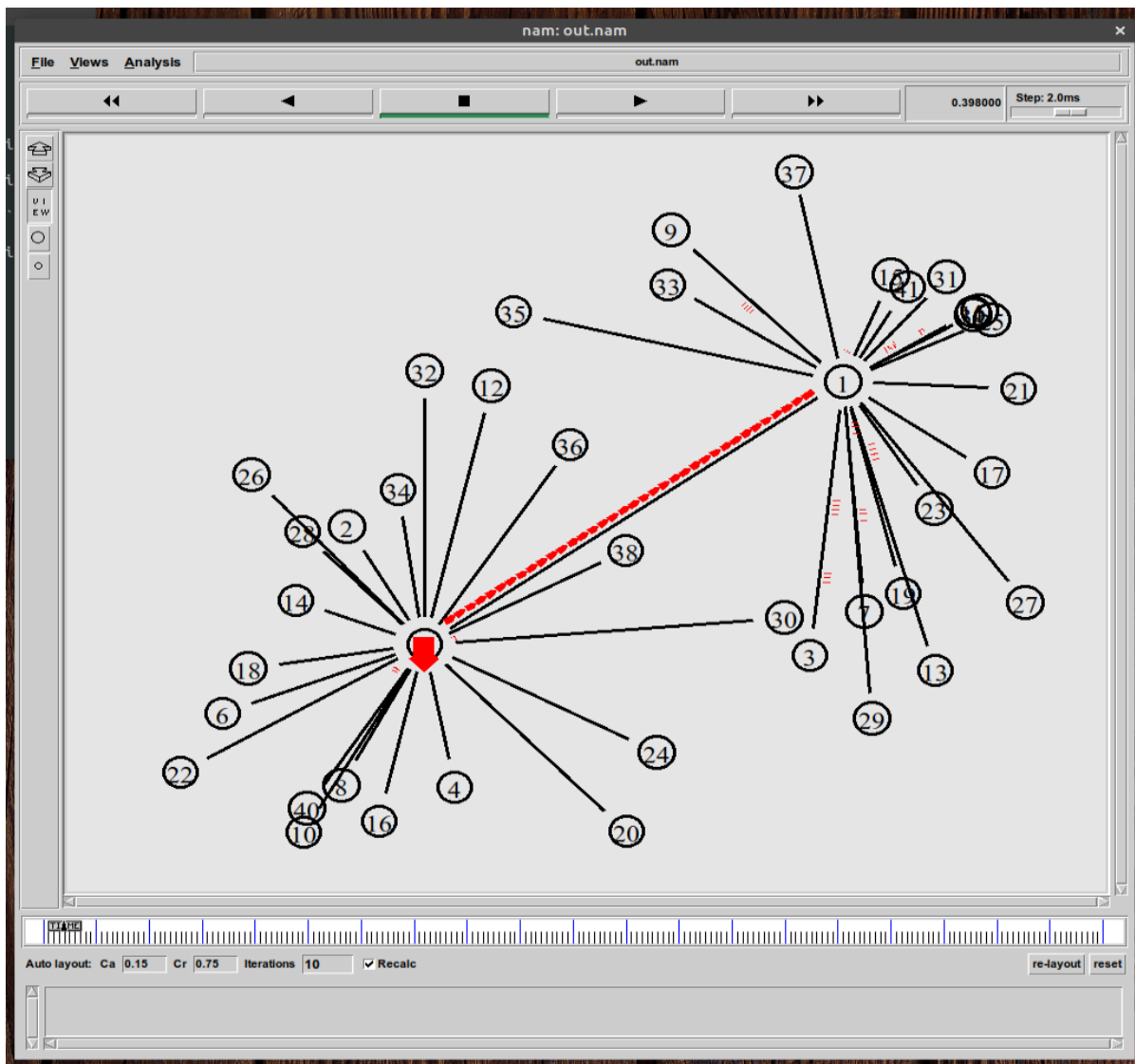


Figure 3.4: Рис 4. Модель сети ns2

Также получаю 2 графика:

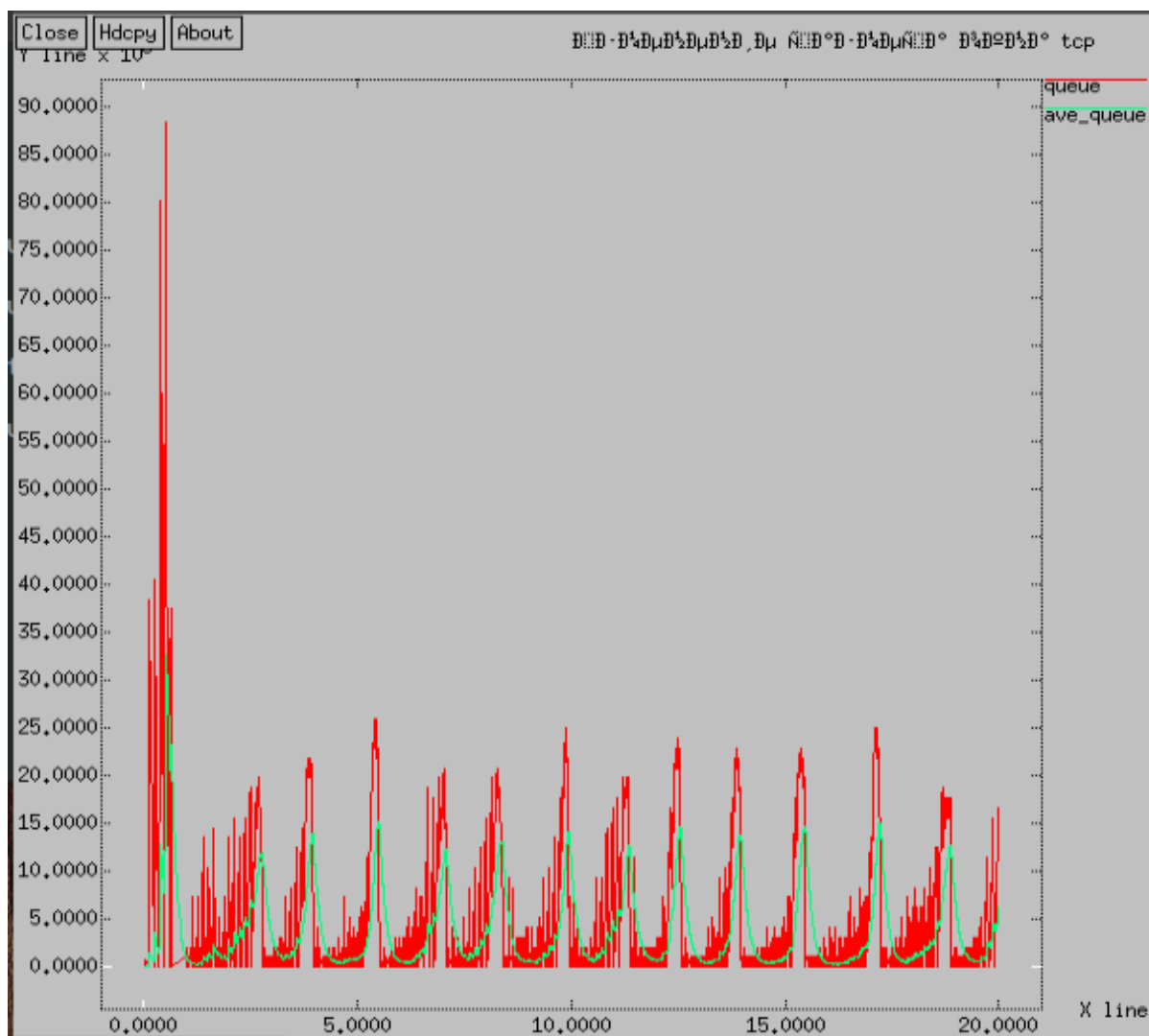


Figure 3.5: Рис 5. Изменение размера длины очереди и средней длины очереди на линке (R1–R2)

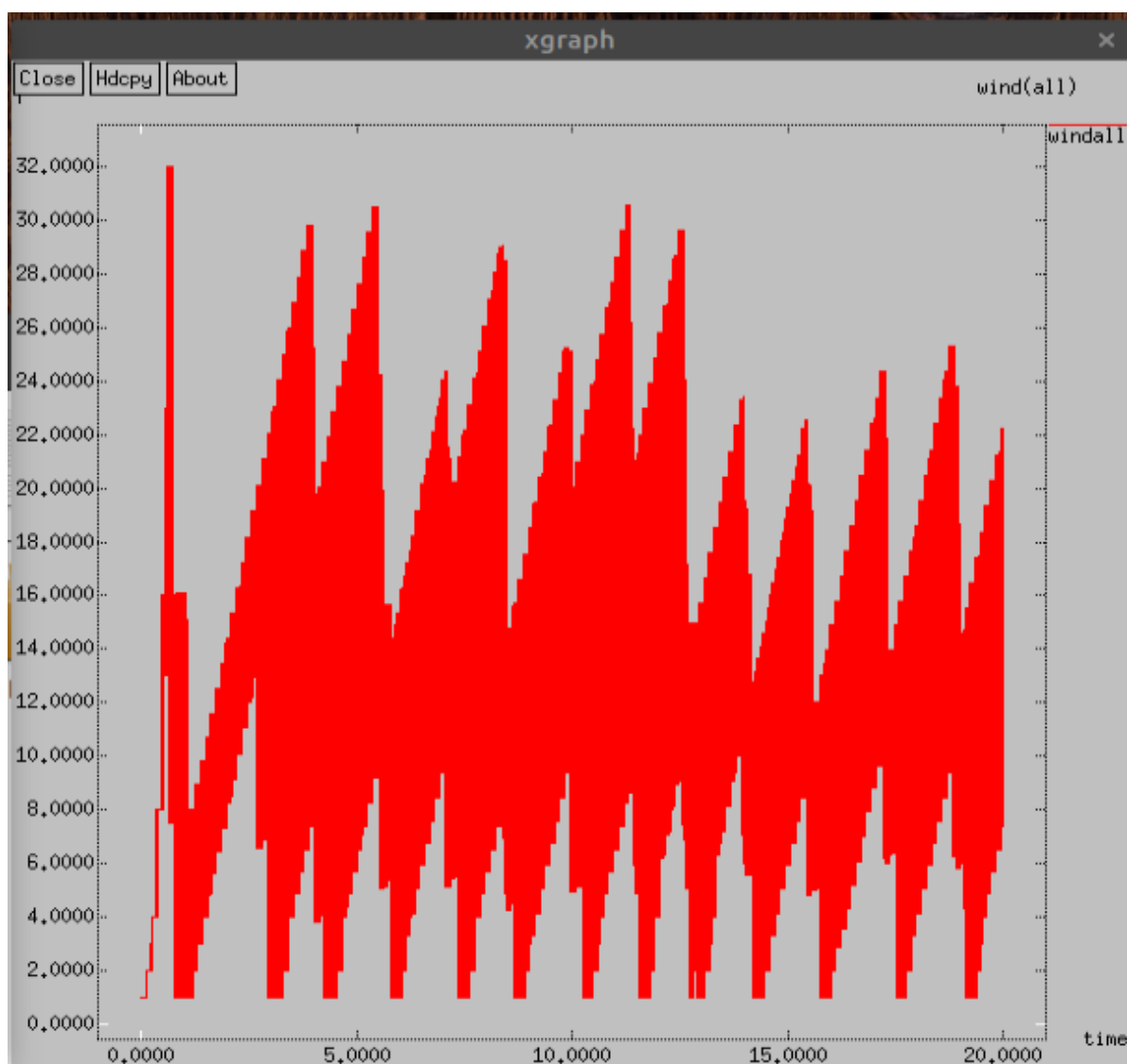


Figure 3.6: Рис 6. Изменение размера окна TCP на линке 1-го источника

4. Запускаю файл graph:

```
./graph
```

Получаю файл pdf, в который сохранены следующие графики:

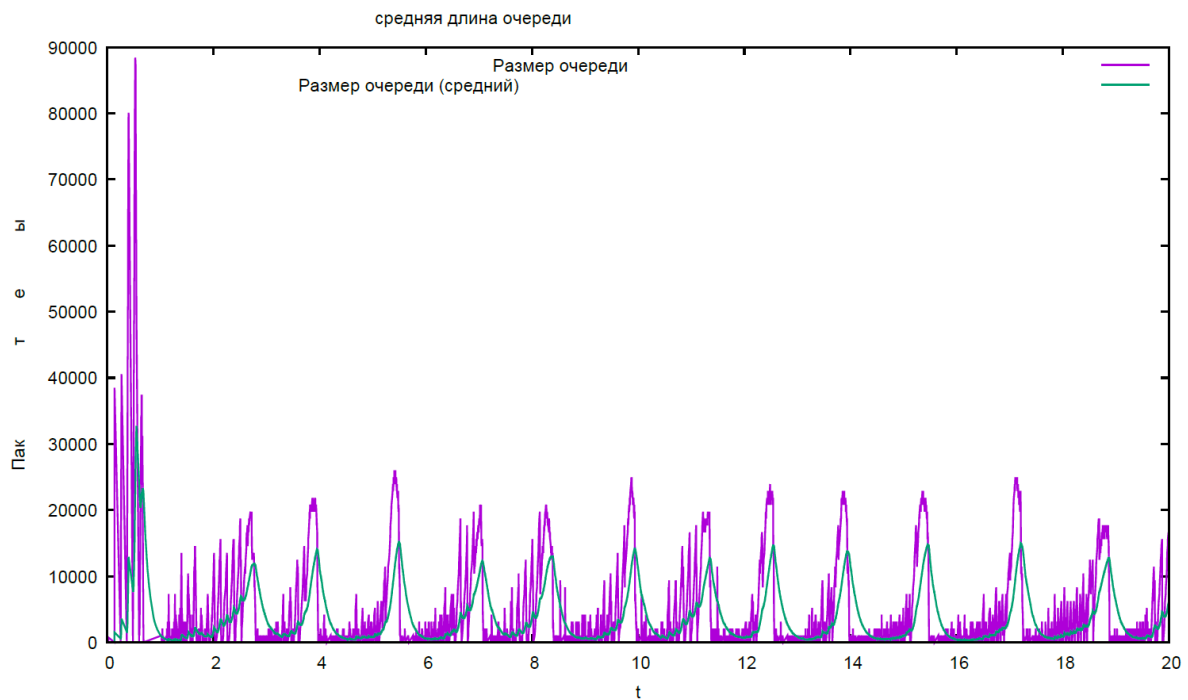


Figure 3.7: Рис 7. Изменение размера длины очереди и средней длины очереди на линке (R1–R2)

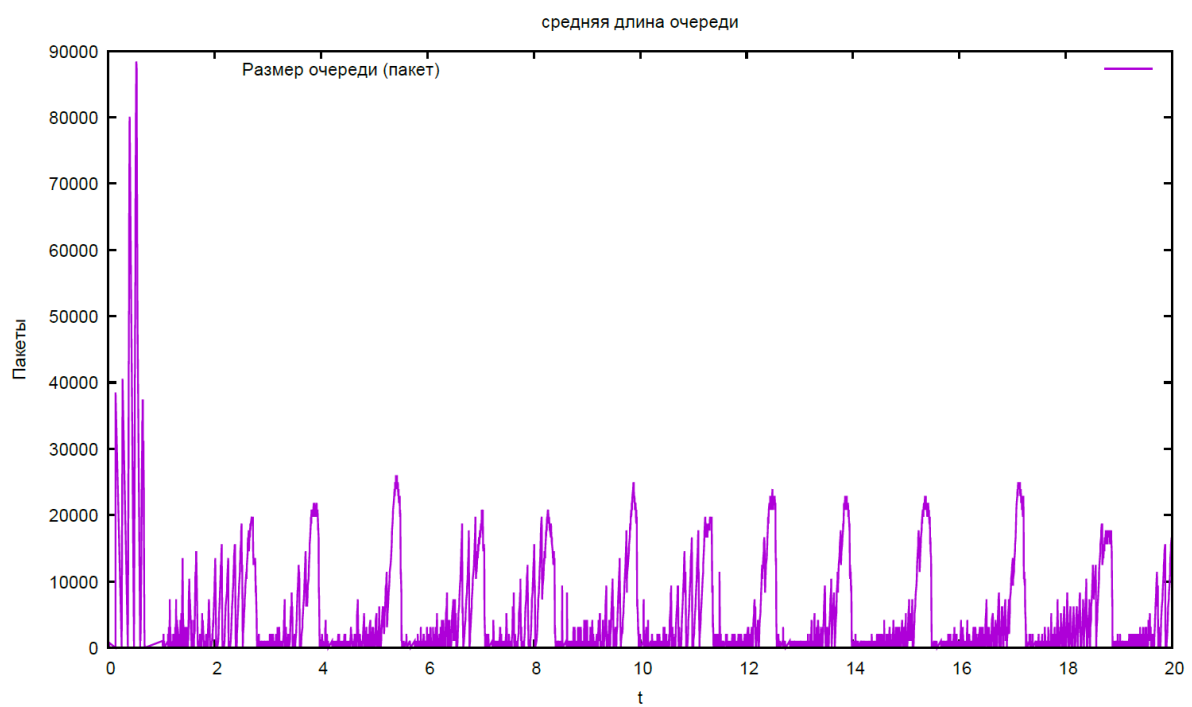


Figure 3.8: Рис 8. Изменение размера средней длины очереди на линке (R1–R2)

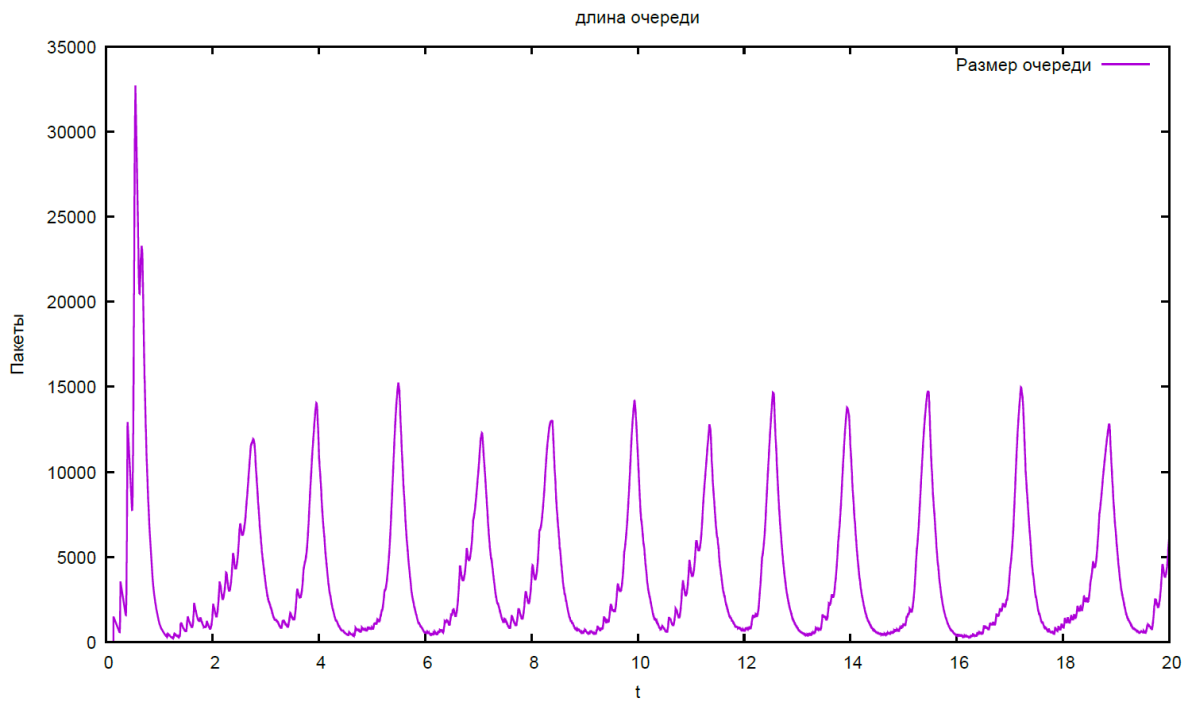


Figure 3.9: Рис 9. Изменение размера длины очереди на линке (R1–R2)

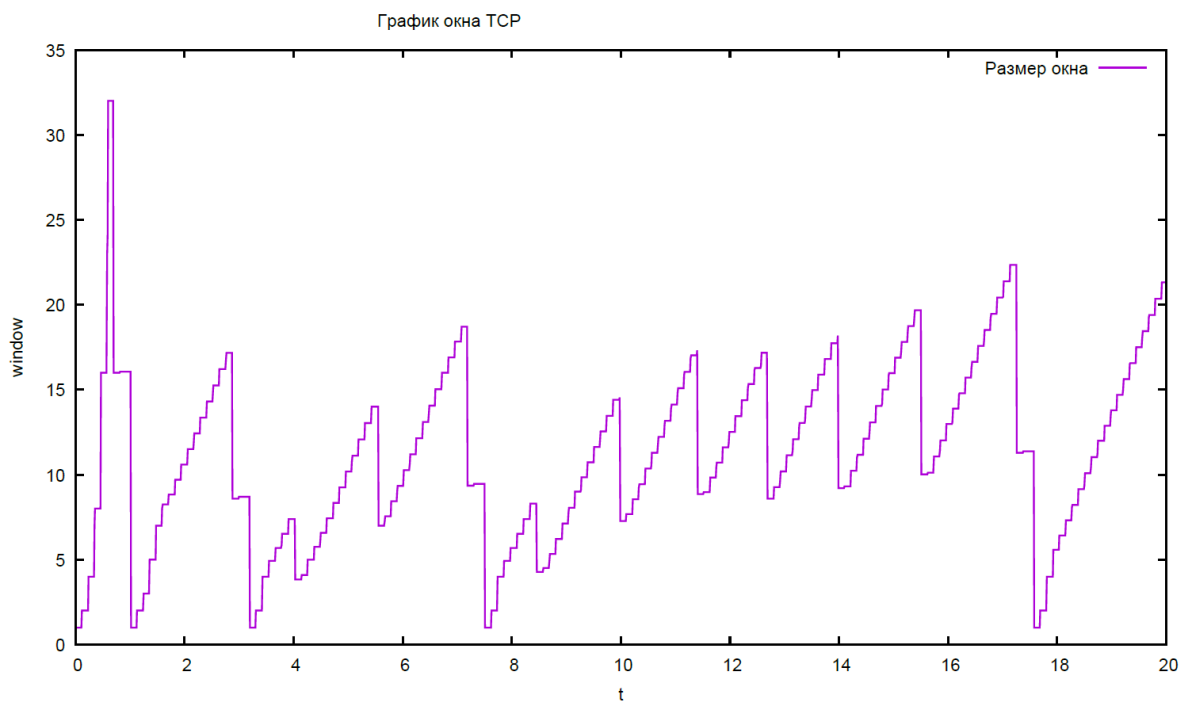


Figure 3.10: Рис 10. Изменение размера окна TCP на линке 1-го источника



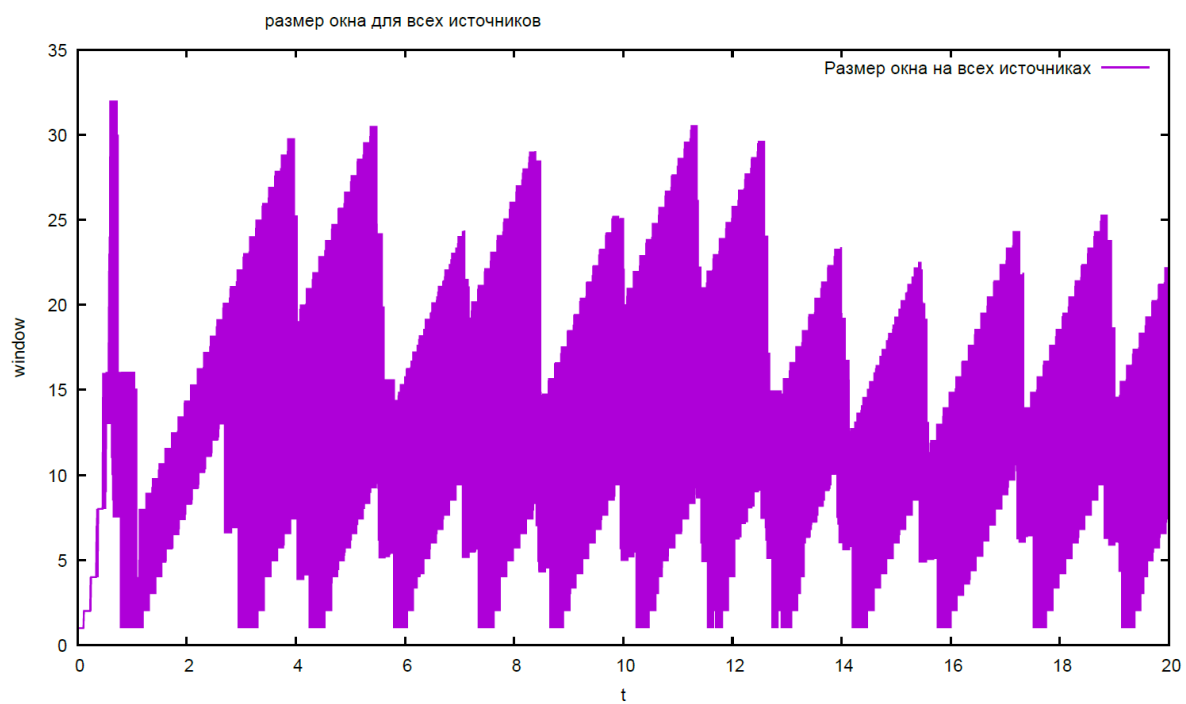


Figure 3.11: Рис 11. Изменение размера окна TCP на всех источниках

## 4 Выводы

В ходе работы были закреплены навыки моделирования сетей, а также были реализованы модель на ns2 и графики в XGraph и GNUpot.

## **5 Библиография**

1. Методические материалы курса