

Лабораторная работа №4

Задание для самостоятельного выполнения

Алиева Милена Арифовна

Российский университет дружбы народов, Москва, Россия

Содержание

1. Цель
2. Задания
3. Порядок выполнения
4. Вывод

Цель работы

Разработать имитационную модель в пакете NS-2, построить графики изменения размера окна TCP, изменения длины очереди и средней длины очереди.

Задание

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

Порядок выполнения

1. Описание моделируемой сети:

- 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;

Порядок выполнения

- 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 пакета 500 байт; время моделирования — не менее 20 единиц модельного времени.

Порядок выполнения

Листинг программы:

```
# создание объекта Simulator
set ns [new Simulator]
# открытие на запись файла out.nam для визуализатора nam
set nf [open out.nam w]
# все результаты моделирования будут записаны в переменную nf
$ns namtrace-all $nf
```

Порядок выполнения

```
# открытие на запись файла трассировки out.tr
# для регистрации всех событий
set f [open out.tr w]
# все регистрируемые события будут записаны в переменную f
$ns trace-all $f
```

```
Agent/TCP set window_ 32
```

```
Agent/TCP set pktSize_ 500
```

Порядок выполнения

```
# процедура finish
proc finish {} {
    global tchan_
    # подключение кода AWK:
    set awkCode {
    {
        if ($1 == "Q" && NF>2) {
            print $2, $3 >> "temp.q";
            set end $2
        }
        else if ($1 == "a" && NF>2)
            print $2, $3 >> "temp.a";
        }
    }
}
```

Порядок выполнения

Порядок выполнения

```
exec rm -f temp.q temp.a
```

```
exec touch temp.a temp.q
```

```
exec awk $awkCode all.q
```

```
# Запуск xgraph с графиками окна TCP и очереди:
```

```
exec xgraph -fg pink -bg blue -bb -tk -x time -t "TCPrenoCWND" WindowVsTimeRe
```

```
exec xgraph -fg pink -bg blue -bb -tk -x time -t "TCPrenoCWND" WindowVsTimeRe
```

```
exec xgraph -bb -tk -x time -y queue temp.q &
```

```
exec xgraph -bb -tk -x time -y queue temp.a &
```

```
exec nam out.nam &
```

```
exit 0
```

```
}
```

Порядок выполнения

```
# Формирование файла с данными о размере окна TCP:
proc plotWindow {tcpSource file} {
    global ns
    set time 0.01
    set now [$ns now]
    set cwnd [$tcpSource set cwnd_]
    puts $file "$now $cwnd"
    $ns at [expr $now+$time] "plotWindow $tcpSource $file"
}
```

Порядок выполнения

```
set r1 [$ns node]
set r2 [$ns node]
$ns simplex-link $r1 $r2 20Mb 15ms RED
$ns simplex-link $r2 $r1 15Mb 20ms DropTail
$ns queue-limit $r1 $r2 300
```

Порядок выполнения

```
set N 30
for {set i 0} {$i < $N} {incr i} {
    set n1($i) [$ns node]
    $ns duplex-link $n1($i) $r1 100Mb 20ms DropTail
    set n2($i) [$ns node]
    $ns duplex-link $n2($i) $r2 100Mb 20ms DropTail
    set tcp($i) [$ns create-connection TCP/Reno $n1($i) TCPSink $n2($i) $i]
    set ftp($i) [$tcp($i) attach-source FTP]
}
```

Порядок выполнения

```
# Мониторинг размера окна TCP:
set windowVsTimeOne [open WindowVsTimeRenoOne w]
set windowVsTimeAll [open WindowVsTimeRenoAll w]

set qmon [$ns monitor-queue $r1 $r2 [open qm.out w] 0.1];
[$ns link $r1 $r2] queue-sample-timeout;
```

Порядок выполнения

```
# Мониторинг очереди:
set redq [[ $ns link $r1 $r2 ] queue]
$redq set thresh_ 75
$redq set maxthresh_ 150
$redq set q_weight_ 0.002
$redq set linterm_ 10

set tchan_ [open all.q w]
$redq trace curq_
$redq trace ave_
$redq attach $tchan_
```

Порядок выполнения

```
for {set i 0} {$i < $N} {incr i} {  
    $ns at 0.0 "$ftp($i) start"  
    $ns at 0.0 "plotWindow $tcp($i) $windowVsTimeAll"  
}
```

```
$ns at 0.0 "plotWindow $tcp(1) $windowVsTimeOne"
```

```
# at-событие для планировщика событий, которое запускает  
# процедуру finish через 20s после начала моделирования
```

```
$ns at 20.0 "finish"
```

```
# запуск модели
```

```
$ns run
```

Порядок выполнения

Порядок выполнения

Запустив созданную программу на выполнение получим nam файл со схемой моделируемой сети (рис. (fig:001?)).

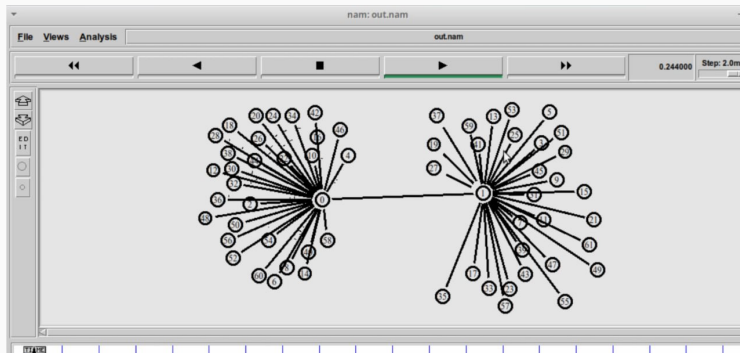


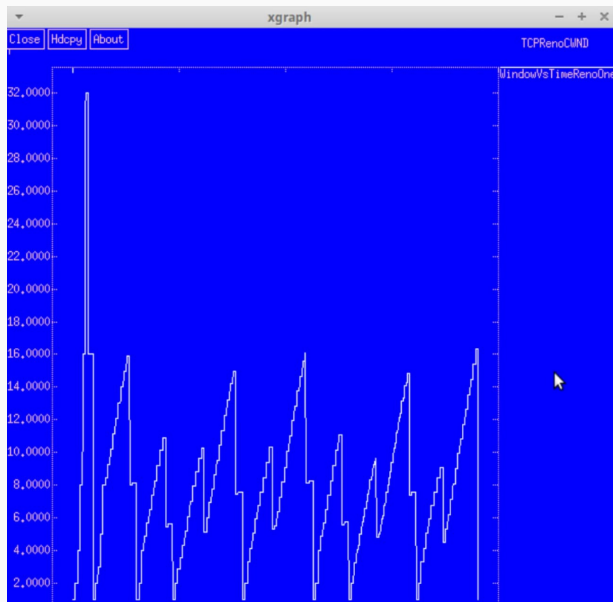
Рис. 1: Моделируемая сеть

Порядок выполнения

2. Получим графики изменения размера окна TCP на линке 1-го источника (рис. (fig:002?)) и на всех источниках (рис. (fig:003?)).

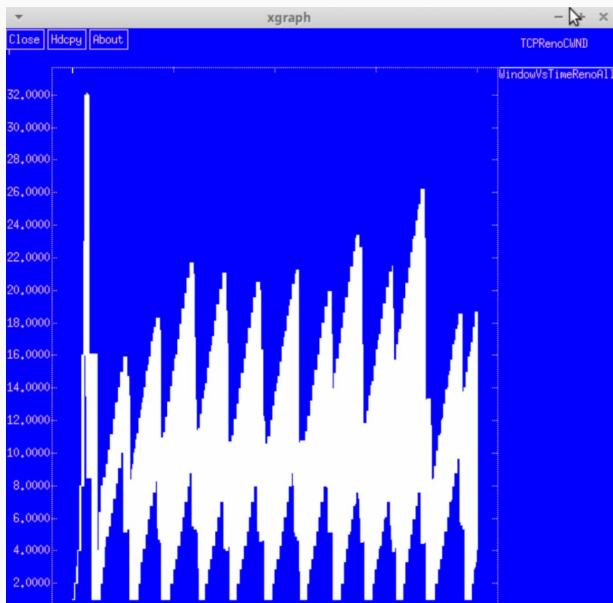
Порядок выполнения

Порядок выполнения



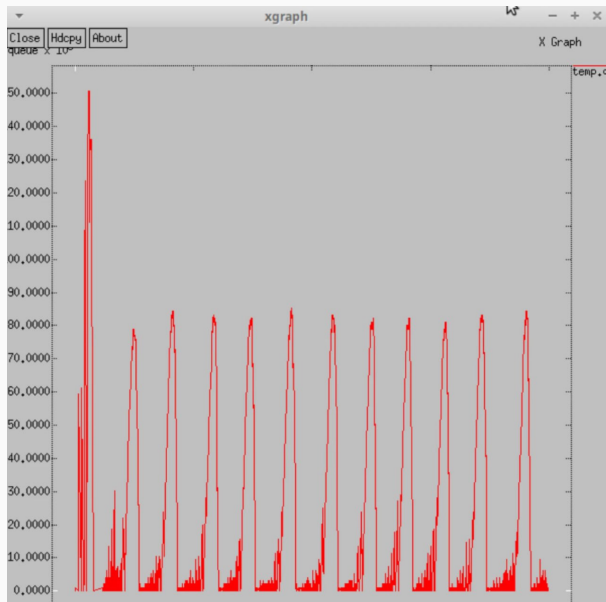
Порядок выполнения

Порядок выполнения



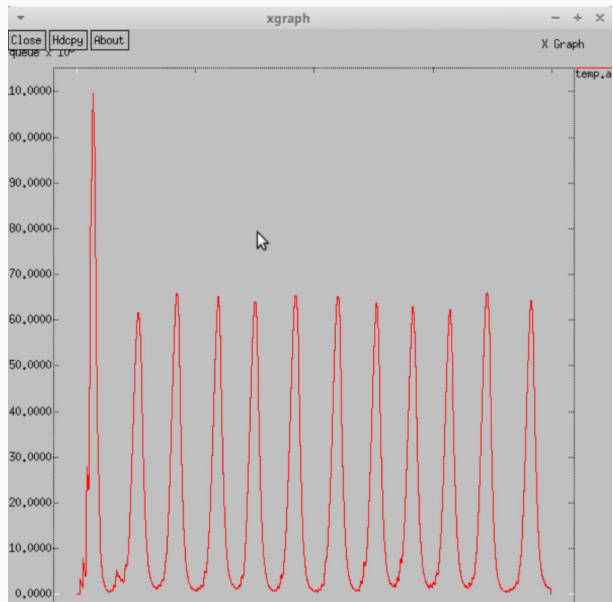
Порядок выполнения

Порядок выполнения



Порядок выполнения

Порядок выполнения



Порядок выполнения

4. Ранее мы строили графики с помощью `xgraph`, теперь реализуем это с помощью `GNUPlot`. Программа для построения графиков в `GNUPlot`.

Порядок выполнения

```
#!/usr/bin/gnuplot -persist  
# задаём текстовую кодировку,  
# тип терминала, тип и размер шрифта  
set encoding utf8  
set term pngcairo font "Helvetica,9"
```

Порядок выполнения

```
# задаём выходной файл графика
set out 'window_1.png'
# задаём название графика
set title "Изменение размера окна TCP на линке 1-го источника при N=30"
# подписи осей графика
set xlabel "t[s]" font "Helvetica, 10"
set ylabel "CWND [pkt]" font "Helvetica, 10"
# построение графика, используя значения
# 1-го и 2-го столбцов файла WindowVsTimeRenoOne
plot "WindowVsTimeRenoOne" using ($1):($2) with lines title "Размер окна TCP"
```

Порядок выполнения

```
# задаём выходной файл графика
set out 'window_2.png'
# задаём название графика
set title "Изменение размера окна TCP на всех N источниках при N=30"
# построение графика, используя значения
# 1-го и 2-го столбцов файла WindowVsTimeRenoAll
plot "WindowVsTimeRenoAll" using ($1):($2) with lines title "Размер окна TCP"
```

Порядок выполнения

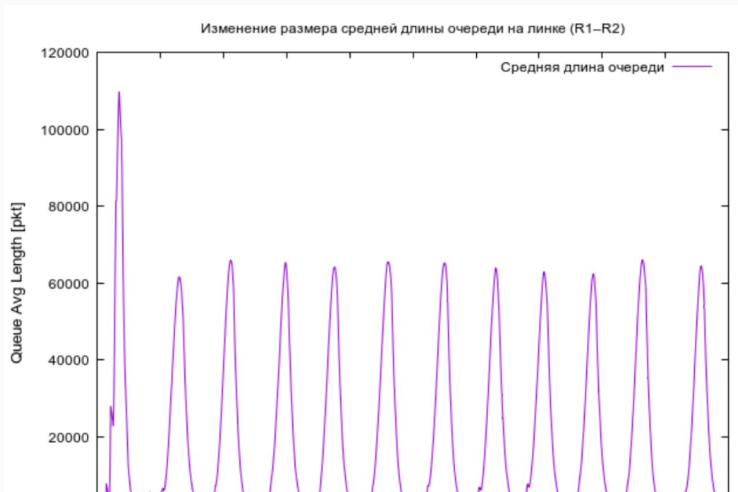
```
# задаём выходной файл графика
set out 'queue.png'
# задаём название графика
set title "Изменение размера длины очереди на линке (R1-R2)"
# подписи осей графика
set xlabel "t[s]" font "Helvetica, 10"
set ylabel "Queue Length [pkt]" font "Helvetica, 10"
# построение графика, используя значения
# 1-го и 2-го столбцов файла temp.q
plot "temp.q" using ($1):($2) with lines title "Текущая длина очереди"
```

Порядок выполнения

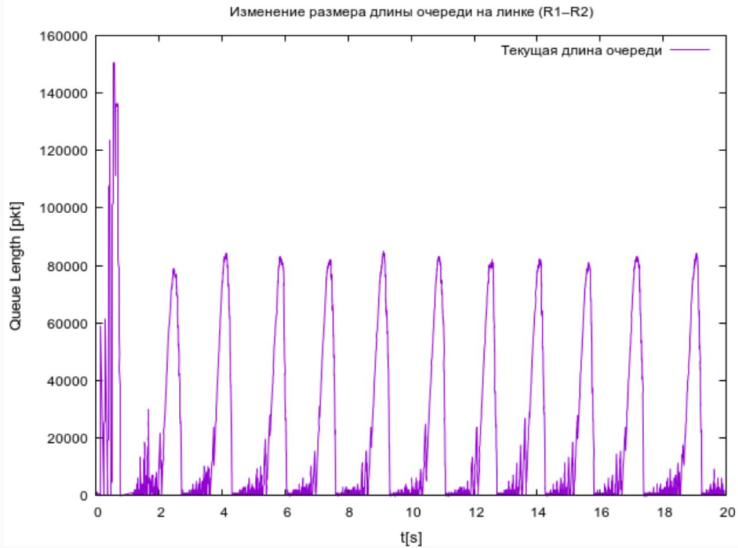
```
# задаём выходной файл графика
set out 'av_queue.png'
# задаём название графика
set title "Изменение размера средней длины очереди на линке (R1-R2)"
# подписи осей графика
set xlabel "t[s]" font "Helvetica, 10"
set ylabel "Queue Avg Length [pkt]" font "Helvetica, 10"
# построение графика, используя значения
# 1-го и 2-го столбцов файла temp.a
plot "temp.a" using ($1):($2) with lines title "Средняя длина очереди"
```

Порядок выполнения

Получим следующие графики: графики изменения размера средней длины очереди (рис. (fig:006?)) и длины очереди (рис. (fig:007?)).



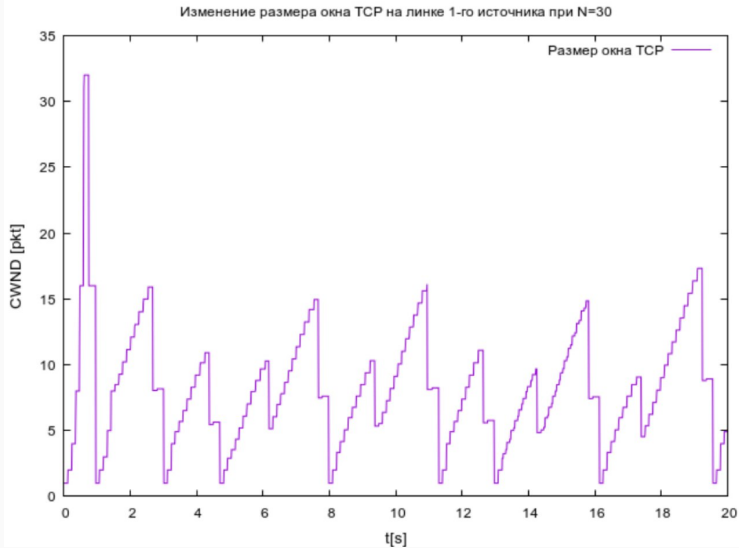
Порядок выполнения



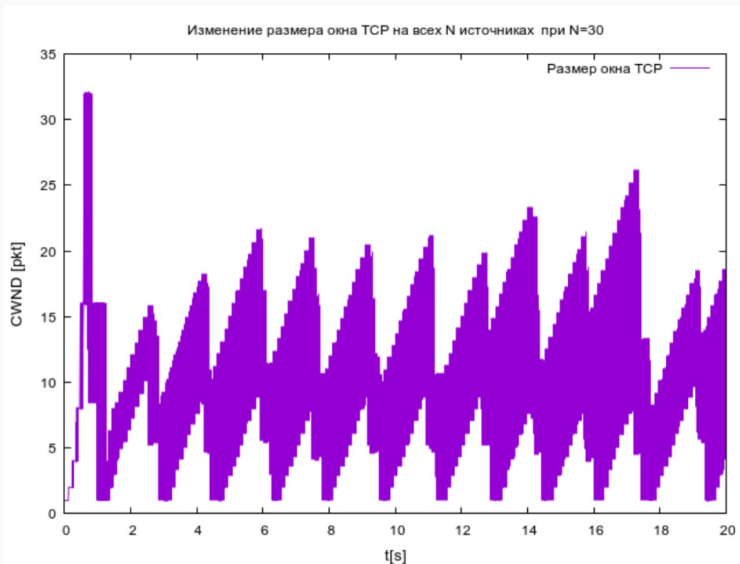
Порядок выполнения

5. Графики изменения размера окна TCP на линке 1-го источника (рис. (fig:008?)) и всех источников (рис. (fig:009?)).

Порядок выполнения



Порядок выполнения



Выводы

В результате выполнения данной лабораторной работы была разработана имитационная модель в пакете NS-2, построены графики изменения размера окна TCP, изменения длины очереди и средней длины очереди.