

Лабораторная работа №4

Задание для самостоятельного выполнения

Абу Сувейлим Мухаммед Мунирачи

04 мая 2024

Российский университет дружбы народов, Москва, Россия

Информация

- Абу Сувейлим Мухаммед Мунифович
- студент, НКНбд-01-21
- Российский университет дружбы народов
- 1032215135@pfur.ru

Вводная часть

- Цели: Приобретение навыков моделирования ситей на NS-2. [1]

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.

- Королькова, А. В. Моделирование информационных процессов : учебное пособие / А. В. Королькова, Д. С. Кулябов. - М. : РУДН, 2014. – 191 с. : ил. [1]
- Korolkova A., Kulyabov D., Черноиванов А. К вопросу о классификации алгоритмов RED // Вестник РУДН. Серия «Математика. Информатика. Физика». 2009. С. 34–46. [2]
- Алленов О. Алгоритм RED: красный свет для лишних пакетов [Электронный ресурс]. 1998. URL: <https://www.osp.ru/nets/1998/09/143680>. [3]

Теоретическое введение

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;
- 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: $q_{min} = 75$, $q_{max} = 150$, $q_w = 0.002$, $p_{max} = 0.1$;

Выполнение работы

1. Во-первых, создадим новый файл example.tcl:

```
touch example.tcl
```

2. и откроем example.tcl на редактирование. Создадим новый объект Simulator:

```
set ns [new Simulator]
```

3. Открываем на запись файл out.tr для регистрации событий:

```
set tf [open out.tr w]  
$ns trace-all $tf
```

4. Открытие на запись файла out.nam для визуализатора nam. Все результаты моделирования будут записаны в переменную nf:

```
set nf [open out.nam w]  
$ns namtrace-all $nf
```

5. Максимальный размер TCP-окна 32; размер передаваемого пакета 500 байт:

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

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

6. Маршрутизаторы:

```
set node_(r1) [$ns node]
```

```
set node_(r2) [$ns node]
```


7. Узлы сети:

```
set N 20
for {set i 0} {$i < $N} {incr i} {
    set node_(s$i) [$ns node]
    $ns duplex-link $node_(s$i) $node_(r1) 100Mb 20ms DropTail

    set node_(n$i) [$ns node]
    $ns duplex-link $node_(n$i) $node_(r2) 100Mb 20ms DropTail

    set tcp_($i) [$ns create-connection TCP/Reno $node_(s$i) TCPSink $node_
    set ftp_($i) [$tcp_($i) attach-source FTP]
}
```

8. Мониторинг очереди и размер окна:

```
set windowVsTime1A [open Window1A w]
for {set i 0} {$i < $N} {incr i} {
    $ns at 0.0 "$ftp_($i) start"
    if {$i == 1} {
        set windowVsTime2B [open Window2B w]
    }
    set qmon [$ns monitor-queue $node_(s$i) $node_(r1) [open qm_($i).out w]
    [$ns link $node_(s$i) $node_(r1)] queue-sample-timeout;
    $ns at 0.0 "plotWindow $tcp_($i) $windowVsTime1A"
}
$ns at 0.0 "plotWindow $tcp_(1) $windowVsTime2B"
```

9. 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:

```
$ns simplex-link $node_(r1) $node_(r2) 20Mb 15ms RED
```

```
$ns queue-limit $node_(r1) $node_(r2) 300
```

```
$ns simplex-link $node_(r2) $node_(r1) 15Mb 20ms DropTail
```

10. Параметры алгоритма RED: $q_{min} = 75$, $q_{max} = 150$, $q_w = 0,002$, $p_{max} = 0.1$:

```
set red [[$ns link $node_(r1) $node_(r2)] queue]  
$red set thresh_ 75  
$red set maxthresh_ 150  
$red set q_weight_ 0.002  
$red set linterm_ 0.1
```

11. Формирование файла с данными о размере окна TCP. Здесь cwnd_ — текущее значение окна перегрузки:

```
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"  
}
```

2. Мониторинг очереди:

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

3. Процедура finish:

```
proc finish {} {
    global ns nf tf tchan_ N
    $ns flush-trace
    close $nf
    # подключение кода AWK:
    set awkCode {
    }
```

```
exec rm -f Current-Queue Avrage-Queue
exec touch Avrage-Queue Current-Queue
exec awk $awkCode all.q
puts $tf \"queue
exec cat Current-Queue >@ $tf
puts $tf \\n\"ave-queue\"
exec cat Avrage-Queue >@ $tf
close $tf
```

Запуск xgraph с графиками окна TCP и очереди:

```
exec xgraph -bb -tk -x time -t CWND "windowVsTime2B" Window2B &
exec xgraph -bb -tk -x time -t CWND "windowVsTime1A" Window1A &
exec xgraph -bb -tk -x time -y "CurrentQueue" CurrentQueue &
```

14. Планировщик событий и запуск модели:

```
$ns at 20 "finish"
```

```
$ns run
```


Результаты

15. График изменения размера окна TCP на линке 1-го источника при N=20:

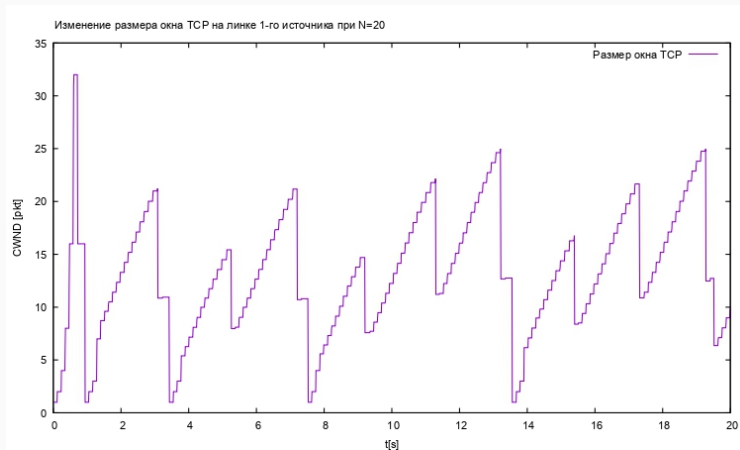


Figure 1: Изменение размера окна TCP на линке 1-го источника при N=20

16. График изменения размера окна TCP на всех источниках при N=20:

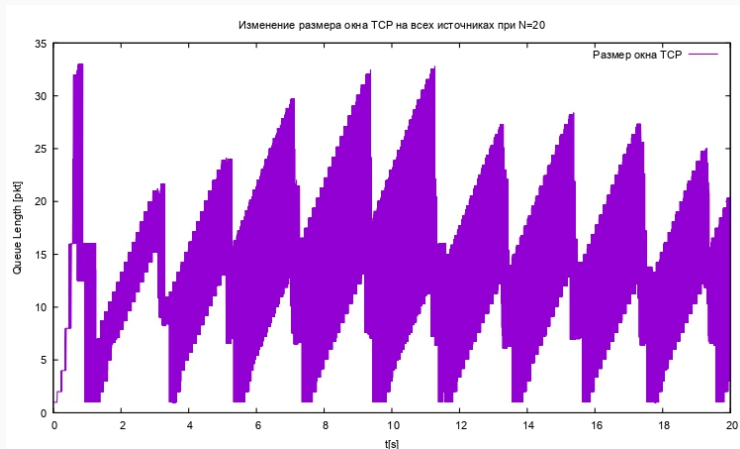


Figure 2: Изменение размера окна TCP на линке 1-го источника при N=20

График изменения размера длины очереди на линке (R1-R2) при $N=20$, $q_{\min} = 75$, $q_{\max} = 150$

17. График изменения размера длины очереди на линке (R1-R2) при $N=20$, $q_{\min} = 75$, $q_{\max} = 150$:

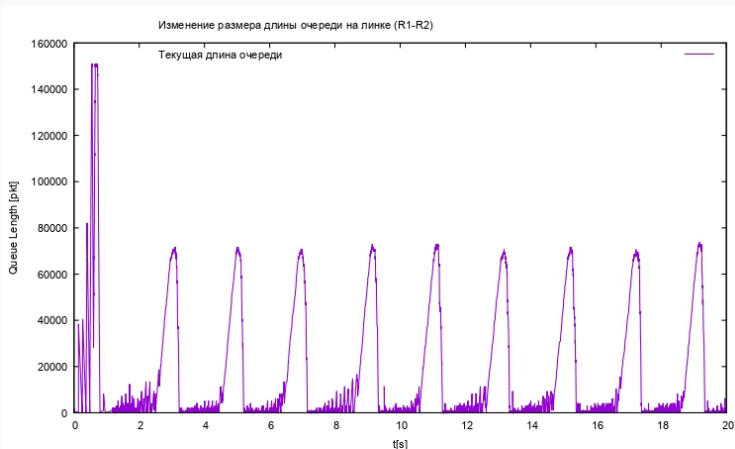
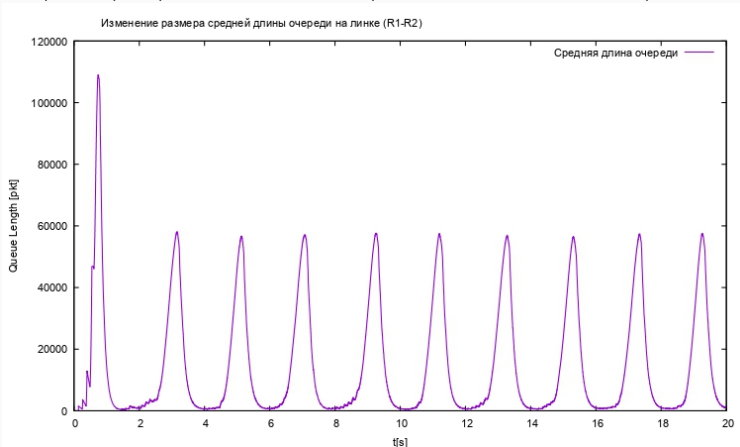


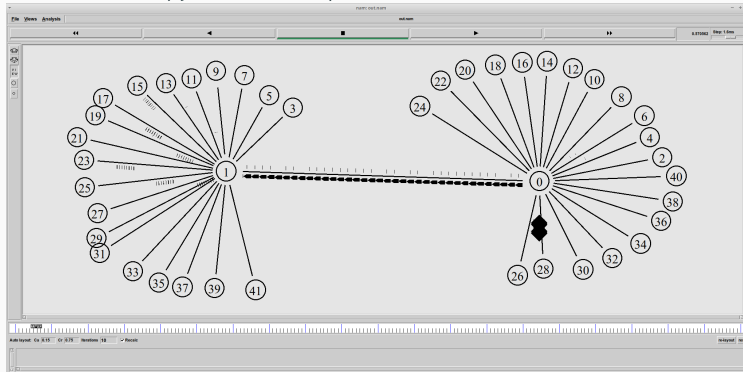
График изменения размера средней длины очереди на линке (R1-R2) при $N=20$, $q_{\min} = 75$, $q_{\max} = 150$

18. График изменения размера средней длины очереди на линке (R1-R2) при $N=20$, q_{\min}



= 75, $q_{\max} = 150$:

19. Схема моделируемой сети при N=20



- Изучали как работает алгоритм RED. [1]