

# **Лабораторная работа №1.**

**Простые модели компьютерной сети**

Александр Андреевич Шуплецов

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Выполнение работы</b>	<b>6</b>
<b>3</b>	<b>Выводы</b>	<b>16</b>
	<b>Список литературы</b>	<b>17</b>

## Список иллюстраций

2.1	создание директории и файла для шаблона . . . . .	6
2.2	содержимое шаблона . . . . .	7
2.3	текст простого примера описания топологии сети . . . . .	8
2.4	вывод простого примера описания топологии сети . . . . .	8
2.5	текст примера с усложненной топологией сети . . . . .	10
2.6	вывод примера с усложненной топологией сети . . . . .	11
2.7	текст примера с кольцевой топологией сети . . . . .	12
2.8	вывод примера с кольцевой топологией сети . . . . .	13
2.9	текст дополненного примера с кольцевой топологией сети . . . .	14
2.10	вывод дополненного примера с кольцевой топологией сети . . . .	15

## Список таблиц

# 1 Цель работы

Приобретение навыков моделирования сетей передачи данных с помощью средства имитационного моделирования NS-2, а также анализ полученных результатов моделирования.

## 2 Выполнение работы

1. Создадим директорию для работы, файл для шаблона.

```
openmodelica@openmodelica-VirtualBox:~/Desktop$ mkdir mip
openmodelica@openmodelica-VirtualBox:~/Desktop$ cd mip
openmodelica@openmodelica-VirtualBox:~/Desktop/mip$ mkdir lab-ns
openmodelica@openmodelica-VirtualBox:~/Desktop/mip$ cd lab-ns
openmodelica@openmodelica-VirtualBox:~/Desktop/mip/lab-ns$ touch shablon.tcl
```

Рис. 2.1: создание директории и файла для шаблона

2. Напишем содержимое шаблона.

```
Файл  Правка  Поиск  Вид  Документ  Справка
# 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
# 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
# процедура finish закрывает файлы трассировки
# и запускает визуализатор nam
proc finish {} {
    global ns f nf |
    $ns flush-trace
    close $f
    close $nf
    exec nam out.nam &
    exit 0
}
# at-событие для планировщика событий, которое запускает
# процедуру finish через 5 с после начала моделирования
$ns at 5.0 "finish"
# запуск модели
$ns run
```

Рис. 2.2: содержимое шаблона

3. Напишем простой пример описания топологии сети, состоящей из двух узлов и одного соединения.

```

# at-событие для планировщика событий, которое запускает
# процедуру finish через 5 с после начала моделирования
# создание 2-х узлов:
set N 2
for {set i 0} {$i < $N} {incr i} {
set n($i) [$ns node]
}
# соединение 2-х узлов дуплексным соединением
# с полосой пропускания 2 Мб/с и задержкой 10 мс,
# очередь с обслуживанием типа DropTail
$ns duplex-link $n(0) $n(1) 2Mb 10ms DropTail
# создание агента UDP и присоединение его к узлу n0
set udp0 [new Agent/UDP]
$ns attach-agent $n(0) $udp0
# создание источника трафика CBR (constant bit rate)
set cbr0 [new Application/Traffic/CBR]
# устанавливаем размер пакета в 500 байт
$cbr0 set packetSize_ 500
# задаем интервал между пакетами равным 0.005 секунды,
# т.е. 200 пакетов в секунду
$cbr0 set interval_ 0.005
# присоединение источника трафика CBR к агенту udp0
$cbr0 attach-agent $udp0
# Создание агента-приёмника и присоединение его к узлу n(1)
set null0 [new Agent/Null]
$ns attach-agent $n(1) $null0
# Соединение агентов между собой
$ns connect $udp0 $null0
# запуск приложения через 0,5 с
$ns at 0.5 "$cbr0 start"
# остановка приложения через 4,5 с
$ns at 4.5 "$cbr0 stop"
$ns at 5.0 "finish"
# запуск модели
$ns run

```

Рис. 2.3: текст простого примера описания топологии сети

4. Вывод простого примера описания топологии сети, состоящей из двух узлов и одного соединения.

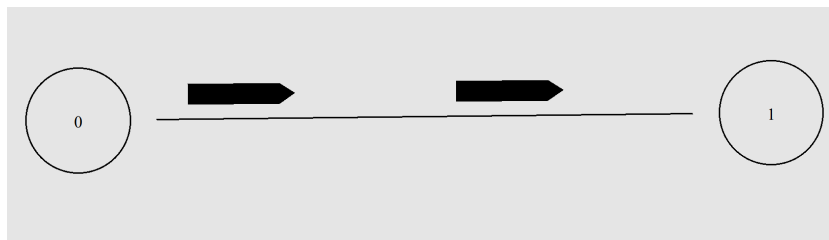


Рис. 2.4: вывод простого примера описания топологии сети



5. Напишем пример с усложненной топологией сети.

```

# at-событие для планировщика событий, которое запускает
# процедуру finish через 5 с после начала моделирования
set N 4
for {set i 0} {$i < $N} {incr i} {
set n($i) [$ns node]
}
$ns duplex-link $n(0) $n(2) 2Mb 10ms DropTail
$ns duplex-link $n(1) $n(2) 2Mb 10ms DropTail
$ns duplex-link $n(3) $n(2) 2Mb 10ms DropTail
$ns duplex-link-op $n(0) $n(2) orient right-down
$ns duplex-link-op $n(1) $n(2) orient right-up
$ns duplex-link-op $n(2) $n(3) orient right
# создание агента UDP и присоединение его к узлу n(0)
set udp0 [new Agent/UDP]
$ns attach-agent $n(0) $udp0
# создание источника CBR-трафика
# и присоединение его к агенту udp0
set cbr0 [new Application/Traffic/CBR]
$cbr0 set packetSize_ 500
$cbr0 set interval_ 0.005
$cbr0 attach-agent $udp0
# создание агента TCP и присоединение его к узлу n(1)
set tcp1 [new Agent/TCP]
$ns attach-agent $n(1) $tcp1
# создание приложения FTP
# и присоединение его к агенту tcp1
set ftp [new Application/FTP]
$ftp attach-agent $tcp1
# создание агента-получателя для udp0
set null0 [new Agent/Null]
$ns attach-agent $n(3) $null0
# создание агента-получателя для tcp1
set sink1 [new Agent/TCPSink]
$ns attach-agent $n(3) $sink1
$ns connect $udp0 $null0
$ns connect $tcp1 $sink1
$ns color 1 Blue
$ns color 2 Red
$udp0 set class_ 1
$tcp1 set class_ 2
$ns duplex-link-op $n(2) $n(3) queuePos 0.5
$ns queue-limit $n(2) $n(3) 20
$ns at 0.5 "$cbr0 start"
$ns at 1.0 "$ftp start"
$ns at 4.0 "$ftp stop"
$ns at 4.5 "$cbr0 stop"
$ns at 5.0 "finish"
# запуск модели
$ns run

```

Рис. 2.5: текст примера с усложненной топологией сети

6. Вывод примера с усложненной топологией сети.

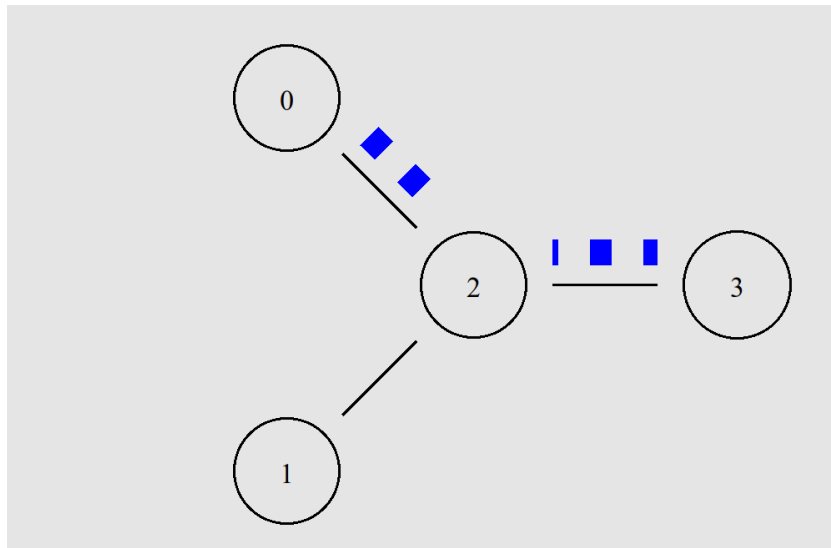


Рис. 2.6: вывод примера с усложненной топологией сети

7. Напишем пример с кольцевой топологией сети.

```

# создание объекта Simulator
set ns [new Simulator]
$ns rtproto DV
# 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
# 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
# процедура finish закрывает файлы трассировки
# и запускает визуализатор nam
proc finish {} {
    global ns f nf
    $ns flush-trace
    close $f
    close $nf
    exec nam out.nam &
    exit 0
}
# at-событие для планировщика событий, которое запускает
set N 7
for {set i 0} {$i < $N} {incr i} {
    set n($i) [$ns node]
}
for {set i 0} {$i < $N} {incr i} {
    $ns duplex-link $n($i) $n([expr ($i+1)%$N]) 1Mb 10ms DropTail
}
set udp0 [new Agent/UDP]
$ns attach-agent $n(0) $udp0
set cbr0 [new Agent/CBR]
$ns attach-agent $n(0) $cbr0
$cbr0 set packetSize 500
$cbr0 set interval_ 0.005
set null0 [new Agent/Null]
$ns attach-agent $n(3) $null0
$ns connect $cbr0 $null0
$ns at 0.5 "$cbr0 start"
$ns rtmodel-at 1.0 down $n(1) $n(2)
$ns rtmodel-at 2.0 up $n(1) $n(2)
$ns at 4.5 "$cbr0 stop"
$ns at 5.0 "finish"
# процедуру finish через 5 с после начала моделирования
$ns at 5.0 "finish"
# запуск модели
$ns run

```

Рис. 2.7: текст примера с кольцевой топологией сети

## 8. Вывод примера с кольцевой топологией сети.

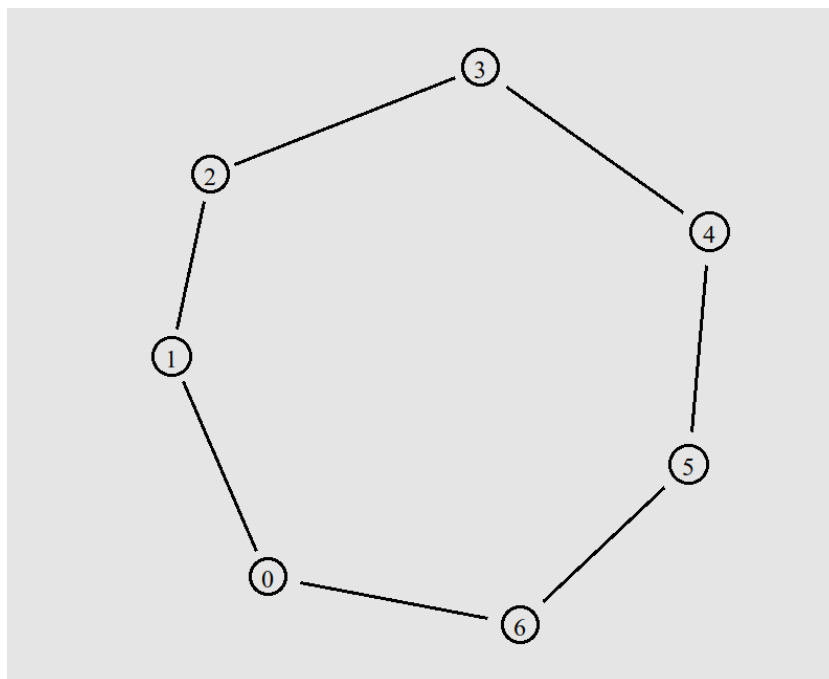


Рис. 2.8: вывод примера с кольцевой топологией сети

9. Дополним пример с кольцевой топологией сети.

```

# создание объекта Simulator
set ns [new Simulator]
$ns rtproto DV
# 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
# 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
# процедура finish закрывает файлы трассировки
# и запускает визуализатор nam
proc finish {} {
    global ns f nf
    $ns flush-trace
    close $f
    close $nf
    exec nam out.nam &
    exit 0
}
# at-событие для планировщика событий, которое запускает
set N 5
for {set i 0} {$i < $N} {incr i} {
    set n($i) [$ns node]
}
for {set i 0} {$i < $N} {incr i} {
    $ns duplex-link $n($i) $n([expr ($i+1)%$N]) 1Mb 10ms DropTail
}
set n5 [$ns node]

$ns duplex-link $n5 $n(1) 1Mb 10ms DropTail
set tcp1 [new Agent/TCP/Newreno]
$ns attach-agent $n(0) $tcp1
set ftp [new Application/FTP]
$ftp attach-agent $tcp1
set sink1 [new Agent/TCPSink/DelAck]
$ns attach-agent $n5 $sink1
$ns connect $tcp1 $sink1
$ns at 0.5 "$ftp start"
$ns rtmodel-at 1.0 down $n(0) $n(1)
$ns rtmodel-at 2.0 up $n(0) $n(1)
$ns at 4.5 "$ftp stop"
# процедуру finish через 5 с после начала моделирования
$ns at 5.0 "finish"
# запуск модели
$ns run

```

Рис. 2.9: текст дополненного примера с кольцевой топологией сети

10. Вывод дополненного примера с кольцевой топологией сети.

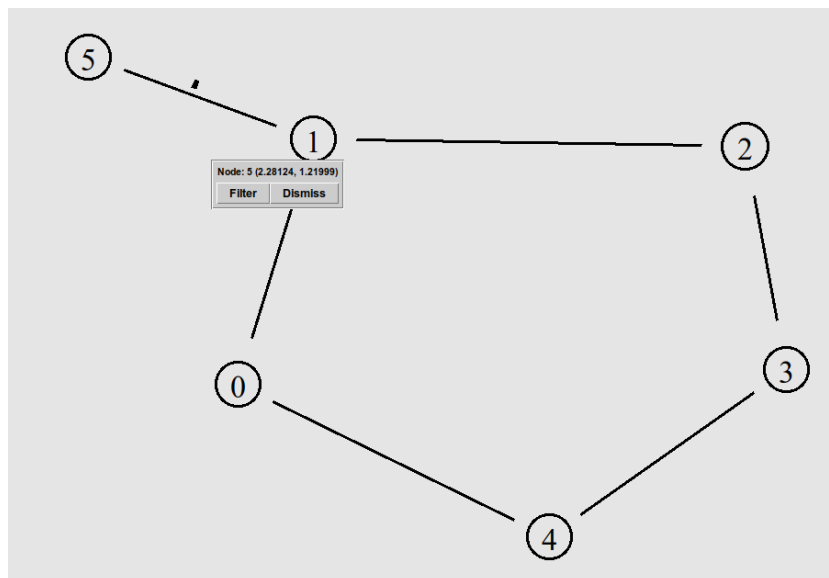


Рис. 2.10: вывод дополненного примера с кольцевой топологией сети

## **3 Выводы**

Я приобрел навыки моделирования сетей передачи данных с помощью средства имитационного моделирования NS-2, а также сделал анализ полученных результатов моделирования.



# Список литературы

Королькова А. В., Кулябов Д.С. “Материалы к лабораторным работам”