

Лабораторная работа №1

Простые модели компьютерной сети

Лихтенштейн А.А.

Российский университет дружбы народов, Москва, Россия

Информация

- Лихтенштейн Алина Алексеевна
- студентка
- Российский университет дружбы народов
- 1132229533@pfur.ru
- <https://aaliechtenstein.github.io/ru/>

Приобрести навыки моделирования сетей передачи данных с помощью средства имитационного моделирования NS-2, а также проанализировать полученные результаты моделирования.

1. Создать шаблон сценария для NS-2;
2. Выполнить простой пример описания топологии сети, состоящей из двух узлов и одного соединения;
3. Выполнить пример с усложнённой топологией сети;
4. Выполнить пример с кольцевой топологией сети;
5. Выполнить упражнение.

Выполнение лабораторной работы

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

Рис. 1: Создание директорий и файла

Шаблон сценария для NS-2

```
*/home/openmodelica/mip/lab-ns/shablon.tcl - Mousepad
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
    # описание глобальных переменных
    $ns flush-trace
    close $f
    close $nf
    # прекращение трассировки
    # закрытие файлов трассировки
    # закрытие файлов трассировки nam
    # запуск nam в фоновом режиме
    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: Редактирование файла shablon.tcl

Шаблон сценария для NS-2

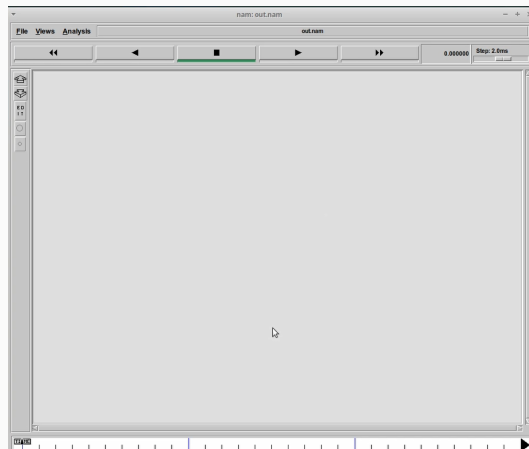


Рис. 3: Запуск шаблона сценария для NS-2

Простой пример описания топологии сети, состоящей из двух узлов и одного соединения

```
# 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"

# 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. 4: Пример описания топологии сети, состоящей из двух узлов и одного соединения

Простой пример описания топологии сети, состоящей из двух узлов и одного соединения

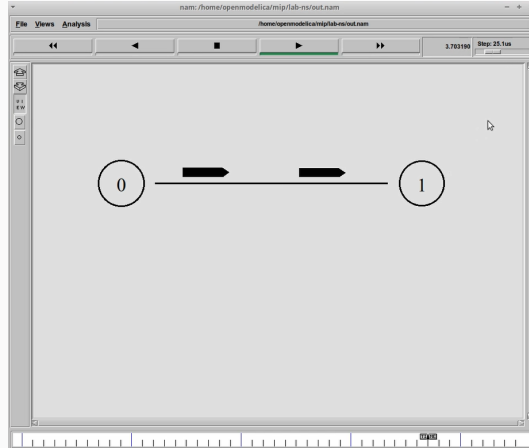


Рис. 5: Визуализация простой модели сети с помощью nam

Пример с усложнённой топологией сети

```
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

# 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. 6: Описание простой модели сети с помощью nam

Пример с усложнённой топологией сети

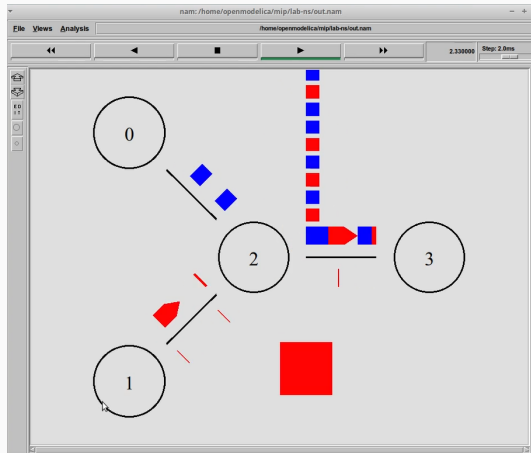


Рис. 7: Визуализация усложненной топологии сети

Пример с кольцевой топологией сети

```
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"

# at-событие для планировщика событий, которое запускает
# процедуру finish через 5 с после начала моделирования
$ns at 5.0 "finish"
# запуск модели
```

Рис. 8: Описание кольцевой топологии сети и динамической маршрутизацией пакетов

Пример с кольцевой топологией сети

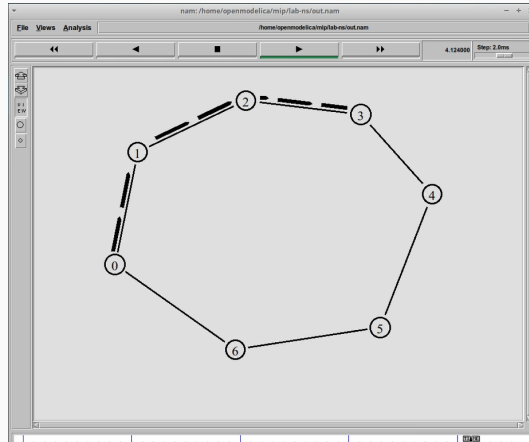


Рис. 9: Передача данных по кратчайшему пути сети с кольцевой топологией

Пример с кольцевой топологией сети

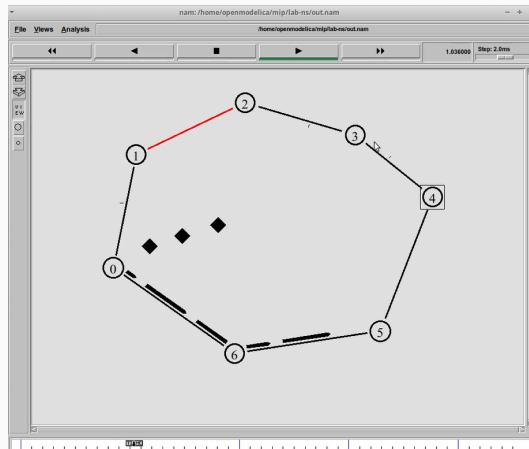


Рис. 10: Передача данных по сети с кольцевой топологией в случае разрыва соединения

Упражнение

```
# Количество узлов
set N 6
for {set i 0} {$i < $N} {incr i} {
    set n($i) {$ns node}
}

# Создание топологии, соответствующей рисунку 1.7
$ns duplex-link $n(0) $n(1) 1Mb 10ms DropTail
$ns duplex-link $n(1) $n(2) 1Mb 10ms DropTail
$ns duplex-link $n(2) $n(3) 1Mb 10ms DropTail
$ns duplex-link $n(3) $n(4) 1Mb 10ms DropTail
$ns duplex-link $n(4) $n(0) 1Mb 10ms DropTail
$ns duplex-link $n(1) $n(5) 1Mb 10ms DropTail

# Создание TCP-агента (NewReno) на n(0)
set tcp0 [new Agent/TCP/Newreno]
$tcp0 set window 8000
$ns attach-agent $n(0) $tcp0

# Создание TCPSink-агента с DeLack на n(5)
set sink [new Agent/TCPSink/DeLack]
$ns attach-agent $n(5) $sink

# Подключение TCP-агентов
$ns connect $tcp0 $sink
$tcp0 set fid_ 1

# Настройка FTP-трафика поверх TCP
set ftp [new Application/FTP]
$ftp attach-agent $tcp0

# Запуск и остановка передачи данных
$ns at 0.5 "$ftp start"
$ns at 4.5 "$ftp stop"

# Разрыв соединения между n(0) и n(1) с 1.0 по 2.0 секунду
$ns rtmodel-at 1.0 down $n(0) $n(1)
$ns rtmodel-at 2.0 up $n(0) $n(1)

# Завершение моделирования
$ns at 5.0 "finish"

# Запуск симуляции
$ns run
```

Рис. 11: Программа для упражнения по построению топологии сети

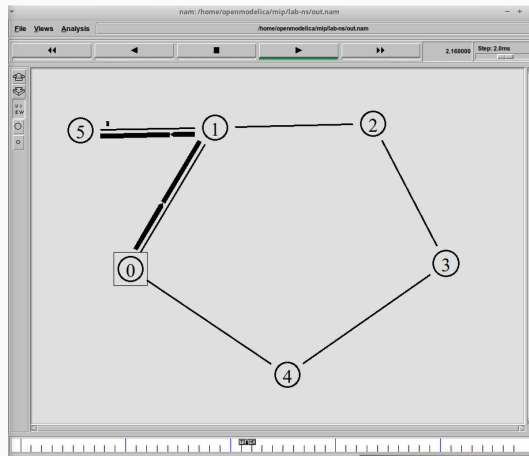


Рис. 12: Передача данных по изменённой кольцевой топологии сети

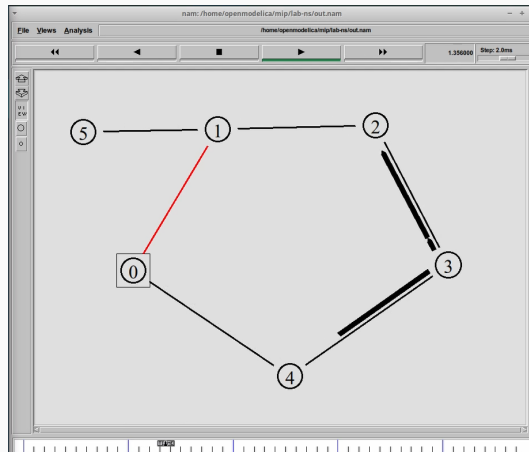


Рис. 13: Передача данных по сети в случае разрыва соединения

В процессе выполнения данной лабораторной работы я были приобретены навыки моделирования сетей передачи данных с помощью средства имитационного моделирования NS-2, а также проанализированны полученные результаты моделирования.

Список литературы

Королькова А.В., Кулябов Д.С. Моделирование информационных процессов