РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

**Факультет физико-математических и естественных наук**

**Кафедра прикладной информатики и теории вероятностей**

ОТЧЕТ

по лабораторной работе № 1

дисциплина: Моделирование информационных процессов

Студент: Худицкий Василий Олегович

Группа: НКНбд-01-19

**МОСКВА**

2022 г.

# Постановка задачи

1. Шаблон сценария для NS-2

Создать шаблон сценария для NS-2, который можно использовать в дальнейшем в большинстве разрабатываемых скриптов, добавляя в него до строки описание объектов и действий моделируемой системы.

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

Требуется смоделировать сеть передачи данных, состоящую из двух узлов, соединённых дуплексной линией связи с полосой пропускания 2 Мб/с и задержкой 10 мс, очередью с обслуживанием типа DropTail. От одного узла к другому по протоколу UDP осуществляется передача пакетов, размером 500 байт,

с постоянной скоростью 200 пакетов в секунду.

1. Пример с усложнённой топологией сети

Описание моделируемой сети:

– сеть состоит из 4 узлов (n0, n1, n2, n3);

– между узлами n0 и n2, n1 и n2 установлено дуплексное соединение с пропускной способностью 2 Мбит/с и задержкой 10 мс;

– между узлами n2 и n3 установлено дуплексное соединение с пропускной способностью 1,7 Мбит/с и задержкой 20 мс;

– каждый узел использует очередь с дисциплиной DropTail для накопления пакетов, максимальный размер которой составляет 10;

– TCP-источник на узле n0 подключается к TCP-приёмнику на узле n3 (по-умолчанию, максимальный размер пакета, который TCP-агент может генерировать, равняется 1KByte)

– TCP-приёмник генерирует и отправляет ACK пакеты отправителю и откидывает полученные пакеты;

– UDP-агент, который подсоединён к узлу n1, подключён к null-агенту на узле n3 (null-агент просто откидывает пакеты);

– генераторы трафика ftp и cbr прикреплены к TCP и UDP агентам соответственно;

– генератор cbr генерирует пакеты размером 1 Кбайт со скоростью 1 Мбит/с;

– работа cbr начинается в 0,1 секунду и прекращается в 4,5 секунды, а ftp начинает работать в 1,0 секунду и прекращает в 4,0 секунды.

1. Пример с кольцевой топологией сети

Требуется построить модель передачи данных по сети с кольцевой топологией и динамической маршрутизацией пакетов:

– сеть состоит из 7 узлов, соединённых в кольцо;

– данные передаются от узла n(0) к узлу n(3) по кратчайшему пути;

– с 1 по 2 секунду модельного времени происходит разрыв соединения между

узлами n(1) и n(2);

– при разрыве соединения маршрут передачи данных должен измениться на резервный.

1. Упражнение

Требуется внести следующие изменения в реализацию примера с кольцевой

топологией сети:

– топология сети должна соответствовать представленной на рис. 1;

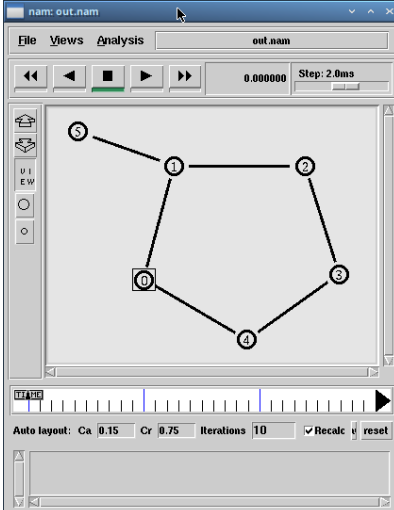


Рисунок Топология сети из упражнения

– передача данных должна осуществляться от узла n(0) до узла n(5) по кратчайшему пути в течение 5 секунд модельного времени;

– передача данных должна идти по протоколу TCP (тип Newreno), на принимающей стороне используется TCPSink-объект типа DelAck; поверх TCP работает протокол FTP с 0,5 до 4,5 секунд модельного времени;

– с 1 по 2 секунду модельного времени происходит разрыв соединения между

узлами n(0) и n(1);

– при разрыве соединения маршрут передачи данных должен измениться на резервный, после восстановления соединения пакеты снова должны пойти по

кратчайшему пути.

# Выполнение работы

1. Шаблон сценария для NS-2

Создал директорию mip, в которой будут выполняться лабораторные работы. Внутри mip создал директорию lab-ns, а в ней файл shablon.tcl

Изображение выглядит как текст

Автоматически созданное описание

Рисунок Создание необходимых директорий и файла шаблона

Написав шаблон, листинг которого представлен ниже, запустить симулятор командой ns shablon.tcl и получил результат, представленный на рис. 3: Изображение выглядит как текст

Автоматически созданное описание

Рисунок Результат создания шаблона

Листинг:

# 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 {} {

# описание глобальных переменных

global ns f nf

# прекращение трассировки

$ns flush-trace

# закрытие файлов трассировки

close $f

# закрытие файлов трассировки nam

close $nf

# 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

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

Скопировал содержимое созданного шаблона в новый файл, изменил его согласно постановке задачи, запустил симулятор(рис. 4):



Рисунок Команды в терминале для примера 1

В результате получил в nam следующую визуализацию(рис. 5):

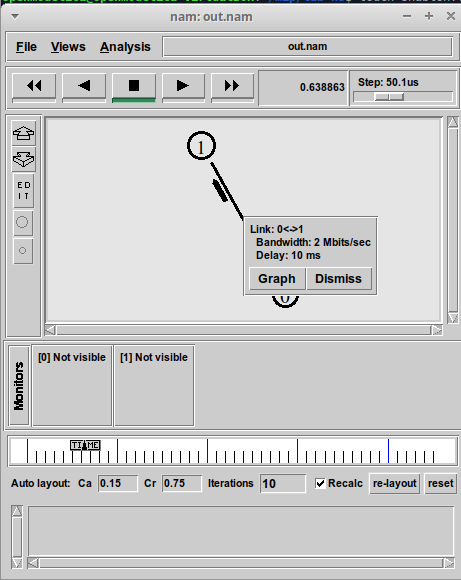


Рисунок Результат для примера 1

Листинг:

# 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 {} {

# описание глобальных переменных

global ns f nf

# прекращение трассировки

$ns flush-trace

# закрытие файлов трассировки

close $f

# закрытие файлов трассировки nam

close $nf

# запуск nam в фоновом режиме

exec nam out.nam &

exit 0

}

# 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,

# очередью 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

1. Пример с усложнённой топологией сети

Скопировал содержимое созданного шаблона в новый файл, изменил его согласно постановке задачи, запустил симулятор(рис. 6):

Изображение выглядит как текст

Автоматически созданное описание

Рисунок Команды в терминале для примера 2

В результате получил в nam следующую визуализацию(рис. 7):

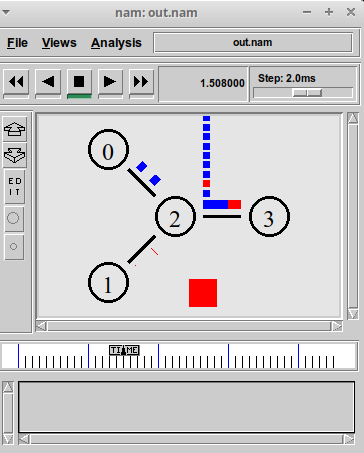


Рисунок Результат для примера 2

Листинг:

# 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 {} {

# описание глобальных переменных

global ns f nf

# прекращение трассировки

$ns flush-trace

# закрытие файлов трассировки

close $f

# закрытие файлов трассировки nam

close $nf

# запуск nam в фоновом режиме

exec nam out.nam &

exit 0

}

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"

# at-событие для планировщика событий, которое запускает

# процедуру finish через 5 с после начала моделирования

$ns at 5.0 "finish"

# запуск модели

$ns run

1. Пример с кольцевой топологией сети

Скопировал содержимое созданного шаблона в новый файл, изменил его согласно постановке задачи, запустил симулятор(рис. 8):

Изображение выглядит как текст

Автоматически созданное описание

Рисунок Команды в терминале для примера 3

В результате получил в nam следующую визуализацию(рис. 9-10):

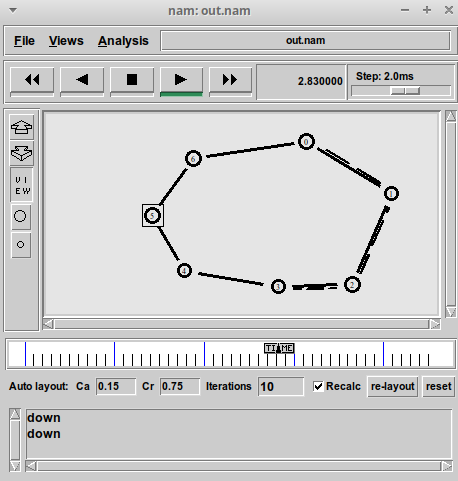


Рисунок Передача данных по кратчайшему пути (пример 3)

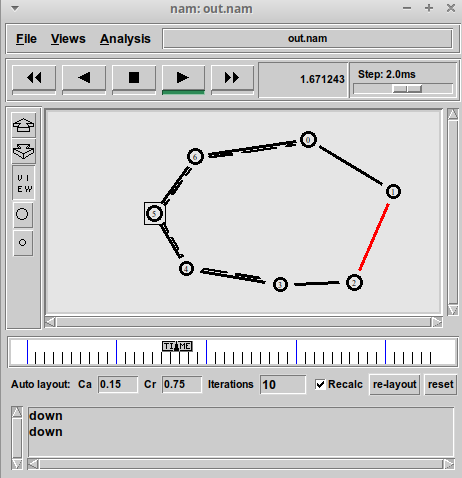


Рисунок Передача данных в случае разрыва соединения (пример 3)

Листинг:

# 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 {} {

# описание глобальных переменных

global ns f nf

# прекращение трассировки

$ns flush-trace

# закрытие файлов трассировки

close $f

# закрытие файлов трассировки nam

close $nf

# запуск nam в фоновом режиме

exec nam out.nam &

exit 0

}

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"

# at-событие для планировщика событий, которое запускает

# процедуру finish через 5 с после начала моделирования

$ns at 5.0 "finish"

# запуск модели

$ns run

1. Упражнение

Скопировал содержимое примера 3 в новый файл, изменил его согласно постановке задачи, запустил симулятор(рис. 11):



Рисунок Команды в терминале для упражнения

В результате получил в nam следующую визуализацию (рис. 12-14):

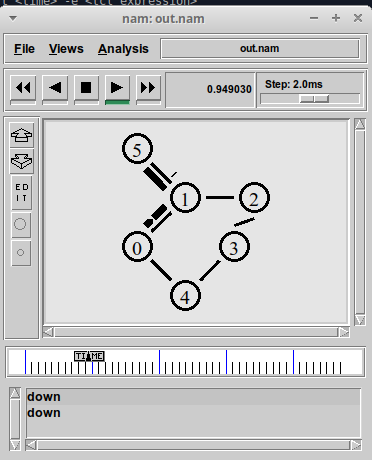


Рисунок Передача данных по кратчайшему пути в начале

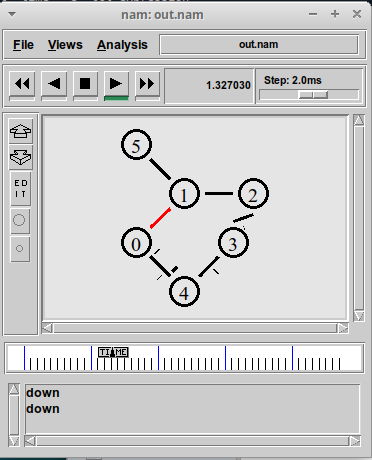


Рисунок Передача данных в случае разрыва соединения

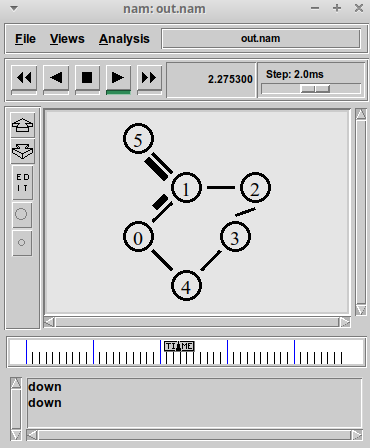


Рисунок Передача данных по кратчайшему пути после второй секунды симуляции

Листинг:

# 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 {} {

# описание глобальных переменных

global ns f nf

# прекращение трассировки

$ns flush-trace

# закрытие файлов трассировки

close $f

# закрытие файлов трассировки nam

close $nf

# запуск nam в фоновом режиме

exec nam out.nam &

exit 0

}

set N 6

for {set i 0} {$i < $N} {incr i} {

set n($i) [$ns node]

}

$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(5) $n(1) 1Mb 10ms DropTail

$ns duplex-link-op $n(0) $n(1) orient right-up

$ns duplex-link-op $n(1) $n(2) orient right

$ns duplex-link-op $n(2) $n(3) orient down

$ns duplex-link-op $n(3) $n(4) orient left-down

$ns duplex-link-op $n(4) $n(0) orient left-up

$ns duplex-link-op $n(5) $n(1) orient right-down

set tcp [new Agent/TCP/Newreno]

$ns attach-agent $n(0) $tcp

set sink [new Agent/TCPSink/DelAck]

$ns attach-agent $n(5) $sink

$sink set interval\_ 100ms

$ns connect $tcp $sink

set ftp [new Application/FTP]

$ftp attach-agent $tcp

$ftp set type\_ FTP

$ns at 0.5 "$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"

# at-событие для планировщика событий, которое запускает

# процедуру finish через 5 с после начала моделирования

$ns at 5.0 "finish"

# запуск модели

$ns run

# Заключение

В ходе выполнения лабораторной работы я приобрёл навыки моделирования сетей передачи данных с помощью средства имитационного моделирования NS-2, создав шаблон сценария для NS-2, рассмотрев примеры моделирования трёх различных сетей, а также внеся изменения в реализацию примера с кольцевой топологией сети.