

УДК 681.3.06(075)

ИССЛЕДОВАНИЕ АЛГОРИТМОВ МАРШРУТИЗАЦИИ С ИСПОЛЬЗОВАНИЕМ СИСТЕМЫ МОДЕЛИРОВАНИЯ NS-2

*Моргайлов Д.Д., Ладыженский Ю.В.
Донецкий национальный технический университет*

Рассматриваются назначение и основные функции системы моделирования NS-2. Приведена структура каталогов NS-2. Описаны основные алгоритмы маршрутизации, доступные в NS-2. Выполнено моделирование алгоритмов маршрутизации Static, Session, DV. Исследована зависимость времени моделирования сети от числа узлов при групповом широковещании.

Введение

NS-2 (Network Simulator 2) – сетевой симулятор, наиболее приспособленный среди аналогичных продуктов для проведения научных исследований в области компьютерных сетей. Немалую долю пользователей симулятора составляют также те, кто использует его в учебных целях (в данном случае простота использования программного средства крайне важна) [1].

Целью написания данной работы является исследование алгоритмов маршрутизации с использованием системы моделирования NS-2. Необходимо решить следующие **задачи**: отметить назначение NS-2, выделить основные функции сетевого симулятора, показать структуру каталогов NS-2, привести модели с использованием алгоритмов маршрутизации и эксперименты с моделями.

Актуальность поставленных задач подтверждается значительным возрастанием требований к эффективности построения и использования компьютерных сетей, ее соответствию текущим требованиям и требованиям ближайшего времени, возможность и стоимость дальнейшего развития и перехода к новым технологиям, особенно в условиях недостаточного финансирования информационных технологий на предприятиях [1].

1 Назначение NS-2

NS (версия 2) – это объектно-ориентированная, управляемая событиями дискретная среда моделирования компьютерных сетей, разработанная в Калифорнийском университете в Беркли на языках C++ и OTcl и работающая в операционной системе Linux. Эта среда позволяет осуществлять имитационное моделирование сетей связи и обладающий целым рядом характеристик, среди которых высокая производительность, хорошая масштабируемость, визуализация результатов и гибкость [2].

NS-2 моделирует передачу сообщений в сетях на уровне приложений, транспортном, сетевом (который в NS-2 называют уровнем маршрутизации), канальном (LL и MAC) и физическом уровнях [3].

NS-2 способен моделировать стационарные и беспроводные сети (спутниковые и мобильные), независимо от их топологии, и поддерживает большое количество протоколов: уровня приложений, сетевого, транспортного, канального и физического уровней [3]. Для мобильных сетей NS-2 способен моделировать поведение пользователей (перемещение узлов сети с течением времени); затухание сигнала в соответствии с различными моделями распространения электромагнитных волн; энергопотребление узлов сети [3].

2 Структура каталогов NS-2

Структура каталогов полной версии NS-2 показана на рис. 1.

3 Основные функции NS-2

Основные функции системы моделирования NS-2 соответствуют объектам системы.

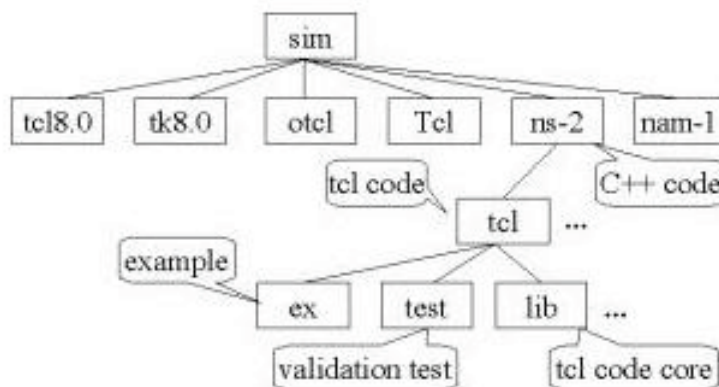


Рисунок 1. Структура каталогов полной версии NS-2

Объекты типа NODE предназначены для работы с узлом: получение списка соседних узлов, прикрепление и отсоединение агентов на заданный порт.

Объекты типа LINK предназначены для определения количества времени, требующегося пакетам для прохождения соединения, а также отслеживания динамики для данного соединения и записи информации в файл. Соединение характеризуется следующими параметрами: полоса пропускания, задержка, вероятность потери пакетов.

Объекты типа QUEUE – основной класс объектов управления пакетами при их движении по моделируемой топологии [4]. Данный объект может использовать один из следующих алгоритмов обработки очереди:

- DropTail (простейший алгоритм типа FIFO);
- FQ (Fair Queuing);
- SFQ (Stochastic Fair Queuing);
- DRR (Deficit Round-Robin scheduling);
- RED (Random Early-Detection gateways);
- CBQ (Class-Based Queuing);
- CBQ/WRR (взвешенное Round-Robin scheduling между классами одного приоритетного уровня).

Объекты типа QUEUEMONITOR используются для мониторинга получаемых, отправляемых и отбрасываемых пакетов и байтов [4]. Они также поддерживают вычисление статистики (средний размер очереди, интегральное значение размера очереди в байтах и пакетах).

Объекты - Агенты:

- rtProto (Static, Session, DV алгоритмы);
- Null (приемник трафика);
- LossMonitor (приемник трафика, поддерживает сбор статистики о полученных данных: число полученных байтов (пакетов), число потерянных байтов (пакетов));
- TCP (Reno, NewReno, Sack1, Fack1, FullTCP);
- TCPSink (приемники TCP пакетов: DelAck, Sack1, Sack1/DelAck);
- CBR (CONSTANT BIT-RATE – генерация пакетов данных с постоянной битовой скоростью);
- SOURCE (генерация данных для пересылки TCP);
- TRAFFIC (создание данных для пересылки по транспортному протоколу UDP в соответствии с законами экспоненциального распределения, распределения Парето, а также с возможностью загрузки трафика из trace файла).

В NS-2 возможно создание нового агента.

Network Animator (Сетевой Аниматор) поддерживает графическое отображение топологии моделируемой сети, анимацию пакетного уровня, а также различные средства инспекции данных [4]. Отображаемые данные могут быть получены в результате работы симулятора типа NS, либо в

результате исследований трафика реальной сети [4].

Xgraph способен читать данные из файлов или непосредственно со стандартного ввода для построения графиков в среде X-Windows [4].

4 Модели и эксперименты с моделями

Далее будет рассмотрено использование доступных в NS-2 алгоритмов маршрутизации (Static, Session, DV).

Ниже приведена топология, состоящая из семи узлов. Каждый узел, за исключением последнего, соединяется со следующим, последний соединяется с первым [4]. Данные передаются от узла 0 к узлу 3 (рис. 2а).

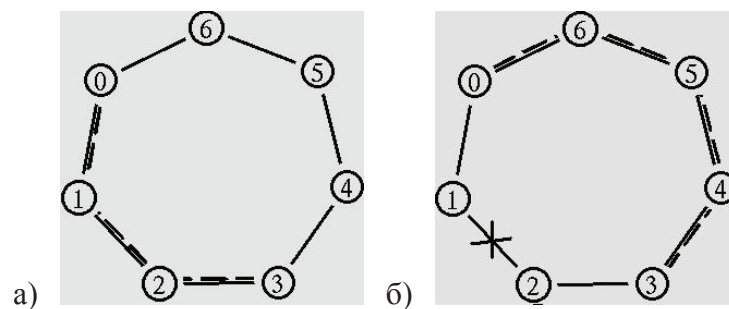


Рисунок 2. Кольцевая топология сети: а) Исходный маршрут пакетов; б) Новый маршрут пакетов

Соединение между узлами 1 и 2 на время между 1.0 и 2.0 секунды будет разорвано, а передаваемые в это время пакеты будут потеряны [4], так как по умолчанию используется алгоритм Static (базируется на алгоритме Dijkstra SPF, маршрут вычисляется один раз перед началом симуляции).

Для решения этой проблемы используется механизм DV маршрутизации, который предполагает вычисление маршрутов с использованием простого протокола маршрутизации с расчетом вектора расстояния [4] (децентрализованная маршрутизация – создается агент на каждом узле). Сразу после запуска симулятора в сети отправляется небольшое количество маленьких пакетов, которые используются для обмена информацией, необходимой для маршрутизации между узлами [4]. После разрыва соединения будет вычислен новый маршрут, и пакеты будут отсылааться через узлы 6, 5 и 4 (рис. 2б).

Однако в рассматриваемом примере предпочтительнее использовать механизм Session маршрутизации (централизованная маршрутизация, маршрут пересчитывается при каждом изменении топологии), поскольку, в отличие от вышеприведенных моделей, будут потеряны только те пакеты, которые в момент разрыва соединения проходили по данному соединению.

Все вышеприведенные алгоритмы маршрутизации поддерживают вычисление маршрута с учетом весов соединений.

В следующем примере задается топология с одиннадцатью узлами, один из которых осуществляет широковещательную (multicast) передачу данных десяти другим (рис. 3).

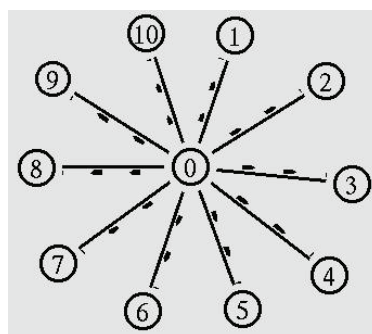


Рисунок 3. Топология с широковещательной передачей данных

Узлы 1, 2, 3, 4, 5 добавляются в группу широковещания до начала моделирования, узлы 6, 7, 8, 9, 10 добавляются в группу широковещания в момент времени 0.5 секунд и удаляются из группы в момент времени 1.5 секунд. Узел 0 начинает передачу данных в момент времени 0.25 секунд.

Для исследования времени моделирования сети с данной топологией необходимо добавлять в сеть новые узлы и включать их в группу узлов, которым адресована широковещательная передача данных (группа multicast). Зависимость времени моделирования сети от числа узлов приведена на рис. 4.

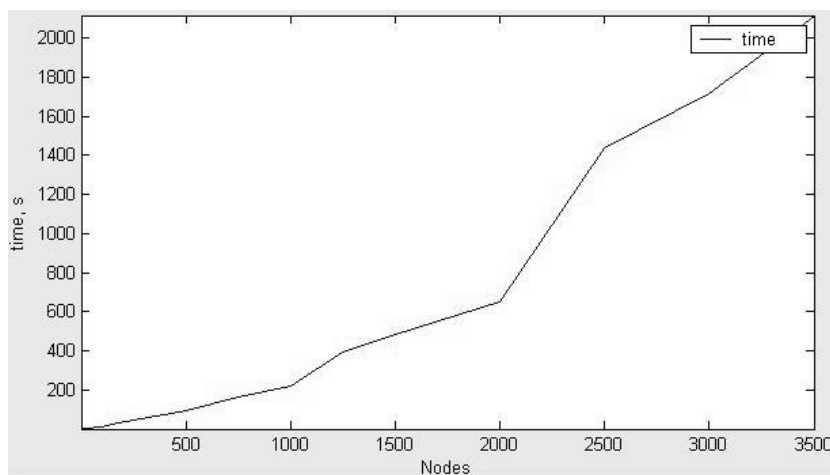


Рисунок 4. График зависимости времени моделирования сети от числа узлов

При числе узлов, равном 4000 и более, из-за недостатка памяти сеть смоделирована не будет. Например, может возникнуть ошибка о невозможности выделения памяти под следующий узел.

Возможности системы были исследованы на персональном компьютере с объёмом оперативной памяти 2 Гб.

Выводы

- Использование полученных в ходе исследования результатов должно помочь повысить эффективность и надежность передачи пакетов данных в реальных компьютерных сетях.
- Приведенные модели могут использоваться в качестве примеров при изучении студентами компьютерных сетей.

Литература

- [1] Преимущества пакета Network Simulator 2. Электронный ресурс. Режим доступа: http://www.networksimulation.ru/index.php?option=com_content&view=article&id=128:2009-12-04-15-35-58&catid=123:2010-04-29-09-34-29&Itemid=231
- [2] А.С. Соколов, А.А. Суконщиков. Применение программного комплекса Network Simulator-2 для моделирования устройств с поддержкой качества обслуживания. Электронный ресурс. Режим доступа: <http://regconf.vstu.edu.ru/comp/userdoc/doc/1621.doc>
- [3] Основы симулятора ns-2. Электронный ресурс. Режим доступа: http://osll.fruct.org/wiki/doku.php?id=papers:ns2_fundamentals
- [4] Network Simulator (Сетевой симулятор ns2). Электронный ресурс. Режим доступа: <http://telematika.stu.neva.ru/files/kafedra/students/posobie/ns-old-html/ns-old.htm>