

МНОГОУРОВНЕВОЕ МОДЕЛИРОВАНИЕ СЕТЕЙ**НА БАЗЕ СТЕКА ПРОТОКОЛОВ TCP/IP****В СРЕДЕ MATLAB/SIMULINK****Анопrienко А.Я., Джон С.Н., Нури А. Ж.**

Кафедра ЭВМ ДонНТУ

anoprien@cs.dgtu.donetsk.ua

Abstract

Anoprienko A.J., John S.N., Nouri A.J. Multilevel simulation of networks on the basis of TCP/IP stack in Matlab/Simulink environment. Multilevel simulation of a network infrastructure is necessary at creation of the effective distributed computing systems, for example, for modeling environments. Efficiency of network interaction at TCP/IP level has in this case one of the major values. The transport control protocol (TCP) is intended for use as the reliable protocol of dialogue between hosts - computers in communication computer networks with packet switching, and also in systems uniting such networks. As an overall objective of the protocol TCP it to ensure reliable service for the communications between processes in multi-network system and to be the common protocol for the communications between hosts-computers in set of networks, that's why it is necessary to analyze optimize some areas of this protocol, for example self-synchronization of the congestion avoidance process. One more problem appears when, for example, a large number of border routers attach to a single route server at an NAP (Network Access Point), significant operational coordination is necessary to maintain routing consistency. An alternative arrangement, in which each border router attach with multi-route server topology, provides other significant benefits. For this purposes it is possible to use Matlab/Simulink environment. Some received results are described in this article.

Введение

При разработке распределенных моделирующих сред [1] на базе современной инфраструктуры Интернет особое значение приобретает повышение эффективности сетевого взаимодействия на всех уровнях стека TCP/IP: начиная от скорости распространения сигналов на физическом уровне и заканчивая быстродействием приложений на прикладном уровне [2]. Но особое значение при этом имеют весьма нетривиальные процессы, происходящие на межсетевом (IP) и транспортном (TCP) уровнях. Построение достаточно адекватных и полных аналитических моделей в этом случае является практически невозможным, поэтому основным методом исследования сетевой инфраструктуры в этом случае является моделирование, как с использованием специализированных средств [3], так и на базе достаточно универсальных систем моделирования.

В данной статье описывается использование программного обеспечения MATLAB (модули SIMULINK и STATEFLOW) для модельного исследования протокола TCP и процессов взаимодействия серверов маршрутизации (уровень IP). При этом анализируются некоторые алгоритмы управления перегрузкой в сети, разработанные в новых версиях протокола TCP, и с помощью разработанной модели проверяется эффективность этих алгоритмов. Исследуются также эффективность различных

вариантов организации взаимодействия маршрутизаторов в точках доступа к сети (NAP - Network Access Point) при отслеживании текущих изменений в сетевой инфраструктуре.

Исследование алгоритмов управления перегрузкой протокола TCP

Для моделирования протокола TCP использовалась наиболее детальная диаграмма состояний, представленная на рис. 1. При этом рассматривались следующие состояния: CLOSED является фиктивным состоянием, поскольку оно представляет состояние, когда не существует блока TCP, а потому и нет соединения.

LISTEN - Ожидание запроса на соединение со стороны чужих портов и программ TCP.

SYN-SENT - Ожидание парного запроса на установление соединения. С нашей стороны запрос уже сделан.

SYN-RECEIVED - Ожидание подтверждения после того, как запрос соединения уже принят и отправлен.

ESTABLISHED - Состояние открытого соединения, принимаемые данные можно представить пользователю. Это нормальное состояние соединения в фазе передачи данных.

FIN-WAIT-1 - Ожидание запроса от чужой программы TCP, или подтверждения ранее отправленного запроса на закрытие соединения.

FIN-WAIT-2 - Ожидание запроса на закрытие соединения со стороны чужой программы TCP.

CLOSE-WAIT - Ожидание запроса на закрытие соединения со стороны своего клиента.

CLOSING - Ожидание подтверждения со стороны чужой программы TCP запроса о закрытии соединения.

LAST-ACK - Ожидание запроса на закрытие соединения, ранее отправленного чужой программой TCP (запрос включал также подтверждение получения чужого запроса на закрытие соединения).

TIME-WAIT - Ожидание когда истечет достаточное количество времени и можно быть уверенным, что чужая программа TCP получила подтверждение своего запроса на закрытие соединения.

CLOSED - Состояние полного отсутствия соединения.

Соединение TCP переходит с одного состояния на другое в ответ на события. Событие - это запросы клиента (открытие, посылка, получение, закрытие, отказ, получение состояния соединения), приход сегментов, и особенно тех, которые содержат флаги SYN, ACK, RST и FIN, а также истечение выделенного времени. Диаграмма состояний на рисунке 2 является сводной и иллюстрирует лишь смену состояний, а также вызвавшие это события, производимые действия, но не адреса, условия ошибок или действия, не связанные прямо с изменением состояния.

Для моделирования использовался пакет Stateflow - интерактивный инструмент проектирования для моделирования и имитации сложных систем. Тесно интегрированный с Simulink® и MATLAB®, Stateflow дает возможность элегантно проектировать встроенные системы, эффективно объединяя сложную логику управления и наблюдения внутри Simulink моделей. С помощью Stateflow можно быстро разрабатывать графические модели управляемых событиями систем, используя теорию конечного состояния машин, формализм диаграмм состояния и нотацию потоковых диаграмм. Комбинация Stateflow, Simulink и MATLAB создает уникальную интегрированную среду моделирования, в которой можно моделировать, имитировать и анализировать сложные динамические системы.

Основой моделирования при этом является аппарат сетей Петри, реализующий условно-событийный подход, для чего компоненты системы и их действия представляются абстрактными **событиями**. Событие может произойти (реализоваться) один раз, повториться многократно или не произойти ни разу.

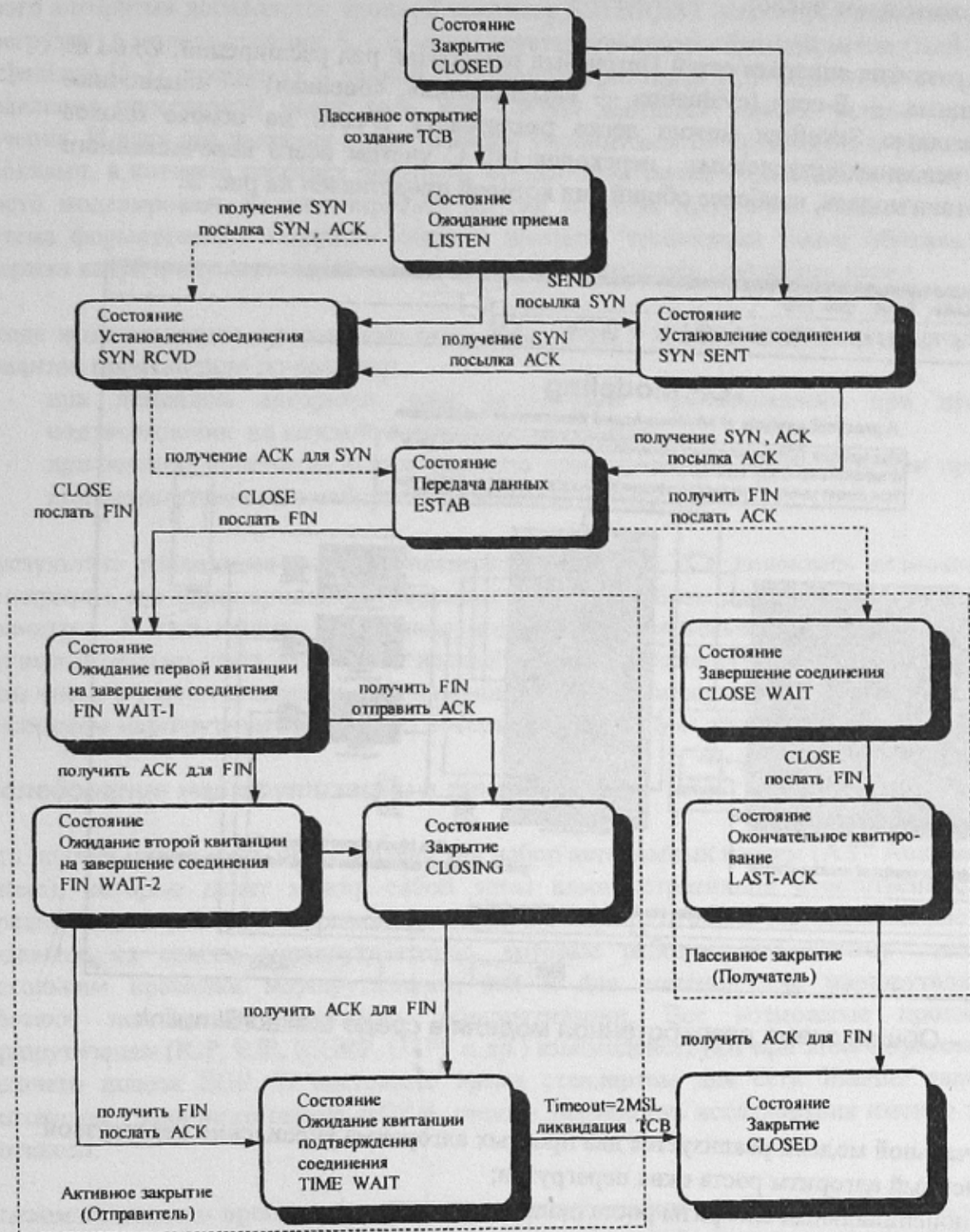


Рис. 1 - Диаграмма состояний протокола TCP

Совокупность действий, возникающих как реализация событий при функционировании системы, образует **процесс**, порождаемый этой системой. В общем случае одна и та же система может функционировать в одних и тех же условиях по-разному, порождая некоторое множество процессов. Для того, чтобы событие произошло, необходимо появление ситуации, в которой это событие может быть реализовано. При этом ситуация определяется как совокупность некоторых **условий** возникновения события.

То есть событие реализуется, если выполнены условия его реализации. Условие может иметь емкость: условие не выполнено (емкость равна нулю), условие выполнено (емкость равна единице), условие выполнено с n -кратным запасом (емкость равна n , где n - целое положительное число).

В результате развития аппарата сетей Петри был разработан ряд расширений. Одно из наиболее мощных — Е-сети (evaluation — «вычисления», «оценка») — «оценочные сети». С помощью Stateflow можно легко реализовать Е-сети на основе блоков состояния и условных/безусловных переходов [4]. С учетом всего перечисленного была разработана модель, наиболее общий вид которой представлен на рис. 2.

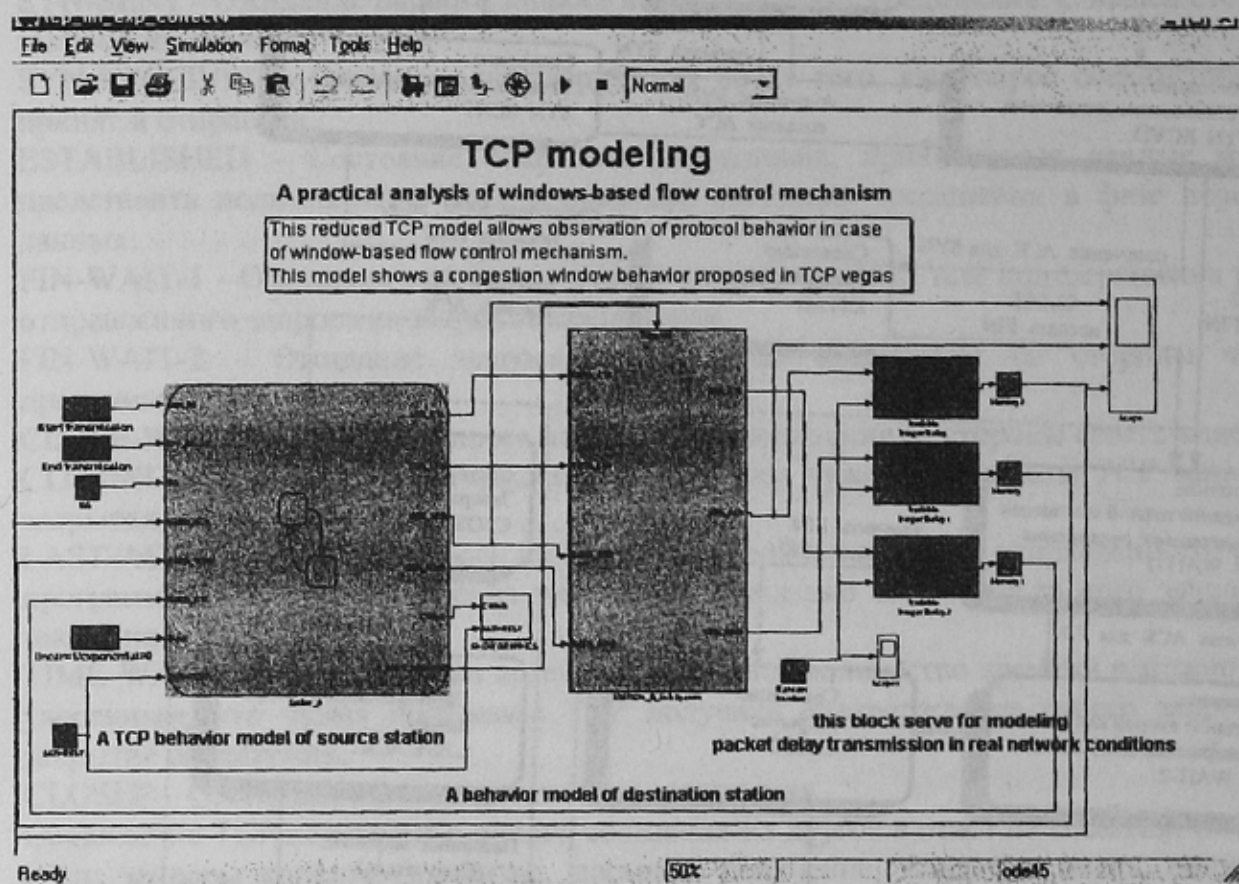


Рис. 2 – Общая схема разработанной модели в среде Matlab/Simulink.

В первоначальной модели реализуется два простых алгоритма управления перегрузкой:

- линейный алгоритм роста окна перегрузки;
- экспоненциальный алгоритм роста окна перегрузки.

В первом алгоритме размер окна перегрузки CWND инкрементируется каждый раз когда станция А принимает подтверждение какого-нибудь сегмента до достижения порогового значения перегрузки которое предлагает сеть в зависимости от количества станций отправителей подключенных к приемной станции или к промежуточному маршрутизатору.

Второй алгоритм имеет экспоненциальный характер поскольку CWND инкрементируется или по экспоненциальному закону (увеличиваясь в два раза), т.е. $CWND(k+1) = 2 * CWND(k)$.

Основная проблема состоит в том, что протокол TCP при своей работе не взаимодействует с сетью для определения оптимальных параметров передачи или по крайней мере, для определения оптимальной скорости своего трафика. Для реализации нового алгоритма добавляется входной параметр Ssthresh (пороговое значение окна перегрузки) в модель станции A, т.е. моделируется механизм обратной петли (feed-back mechanism) [5-7], поскольку в данной реализации протокола TCP выполняет алгоритм управления перегрузкой после того, как CWND достигает заранее установленного значения. И хотя это является существенным упрощением по сравнению с реальными условиями, в которых работает протокол, но зато появляется возможность достаточно просто моделировать и анализировать другие аспекты протокола. Изменена также система формирования задержки кадра в процессе трансляции таким образом, что задержка кадра постоянно изменяется с помощью генератора случайных чисел.

В ходе моделирования приращение окна перегрузки в каждом из 2-х исследованных вариантов происходило по-разному:

- при линейном алгоритме окно перегрузки увеличивалось при приеме подтверждения на каждый отправленный сегмент;
- при экспоненциальном алгоритме окно перегрузки увеличивалось при приеме подтверждении всего набора отправленных сегментов.

В результате реализации модели поведения протокола TCP появилась возможность имитировать его характеристики в реальных условиях работы при различных исходных параметрах. В дальнейшем полученная модель может использоваться в комплексе с другими моделями стека TCP/IP для исследования процессов сетевого взаимодействия, в том числе с учетом анализа и оптимизации работы промежуточных устройства в сети (в основном маршрутизаторов), через которые проходит весь трафик TCP.

Исследование маршрутизации в точках NAP

Сеть Internet может быть представлена как набор автономных систем (AS - Autonomous System), которые делят между собой зоны административной ответственности и определяют для различных организаций правила маршрутизации. Автономные системы создаются на основе маршрутизаторов, которые работают по одному или по нескольким правилам маршрутизации, как и для доменов, где маршрутизаторы работают по одному протоколу маршрутизации. Все возможные протоколы маршрутизации (IGP, RIP, EIGRP, OSPF и др.) взаимодействуют при этом с протоколом внешнего шлюза EGP. В настоящее время стандартом для сети Internet является протоколом граничного шлюза BGP-4, отсюда значимость исследования именно этого протокола.

В целом сложность проблем организации маршрутизации в крупномасштабных сетях TCP/IP тесно связана с топологией сети и принятыми в ней правилами маршрутизации. В последних спецификациях дистанционно-векторных протоколов EIGRP, RIP-2 и, в частности, BGP-4, были устранены почти все серьезные проблемы, связанные с этой категорией протоколов, кроме явления конвергенции (convergence), определяемой интервалом времени, за который обновляются все маршруты в сети, то есть устанавливается факт существования, отсутствия или изменения того или иного маршрута. Процесс удержания и исключения из таблицы маршрутов в больших сетях может длиться до нескольких (иногда до 15-ти и более) минут, пока не произойдет полная конвергенция, то есть, пока всем узлам сети не будет сообщена полная информация об изменении маршрута.

В процессе конвергенции резко увеличивается занятость процессоров на маршрутизаторах, так как они вынуждены заниматься одновременно текущим регулированием трафика и обновлением маршрутной информации. Причем, обновление может занимать большую часть пропускной способности каналов связи, поскольку обмен довольно объемной маршрутной информацией и доставка пакетов проводятся по одному каналу. Взаимодействие серверов маршрутизации при этом может быть организовано как в рамках полносвязной топологии, так и в рамках более простых и эффективных топологических вариантов взаимосвязи.

В рамках данной работы, в частности, были исследованы три топологии:

- топология “звезда” с разным количеством спикеров (рис. 3);
- топология 4-1-4-1 с двумя серверами (рис. 4);
- топология “дерево” с каскадной структурой серверов.

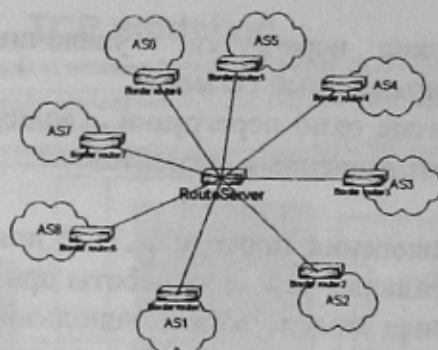


Рис. 3 - Топология “звезда”.

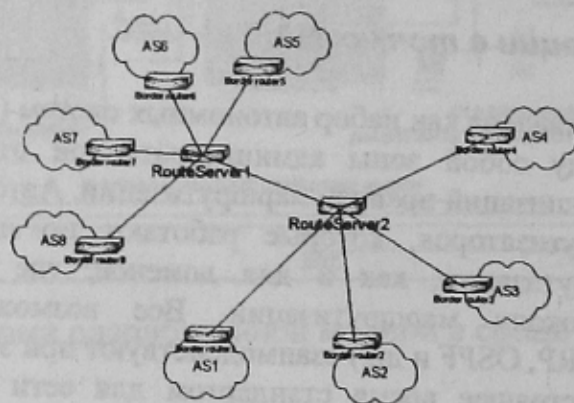


Рис. 4 - Топология 4-1-4-1.

Каждый сервер должен инициализировать соединение отдельно с каждым так называемым «спикером» на нижнем уровне, с граничным сервером своего кластера на высшем уровне и также синхронизацию с соседними серверами. В среде моделирования Simulink и Stateflow можно с помощью компонента “подсистема” реализовать отдельные блоки для каждого сервера и для каждого соединения, которые могут обмениваться сигналами (данными) через порты. Пример одной из таких подсистем представлен на рис. 5.

После реализации всех моделей в среде Simulink/Stateflow (топология “звезда” с разным количеством спикеров, топология 4-1-4-1 с двумя серверами и топология

“дерево” с каскадной структурой серверов) анализировались минимум три параметра для сравнения эффективности той или иной топологии:

- входящий и выходящий трафик, который проходит через сервер маршрутизации;
- занятость серверов при процессе обновления;
- количество связей в каждой топологии.

Некоторые из полученных в результате моделирования зависимостей представлены на рисунках 6-8.

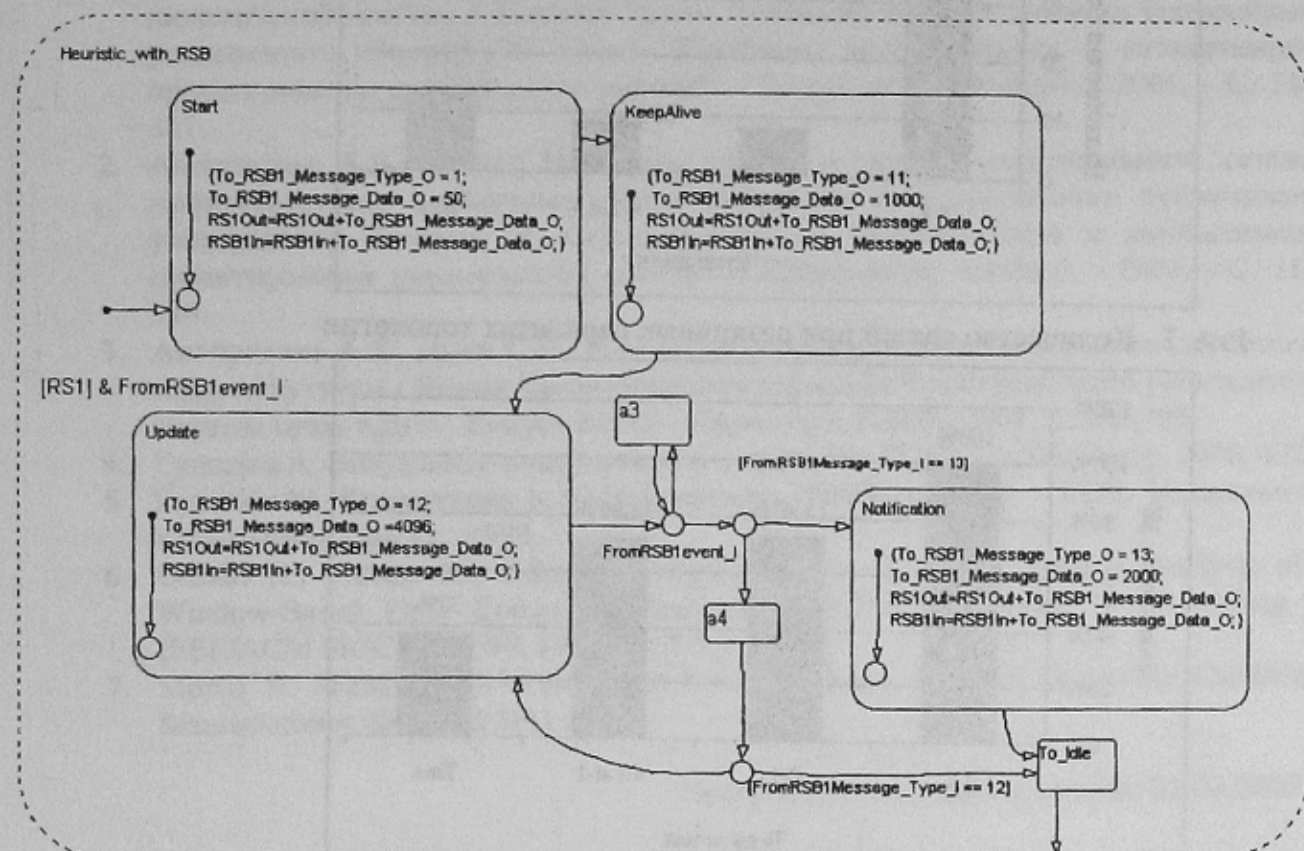


Рис. 5 – Пример моделирования алгоритма работы сервера при инициализации соединения с другим сервером одного кластера

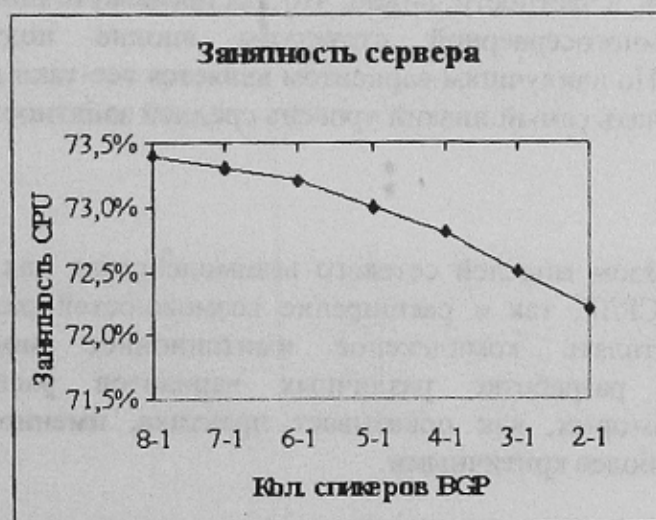


Рис. 6 - Занятость сервера при различных вариантах топологии «звезда»

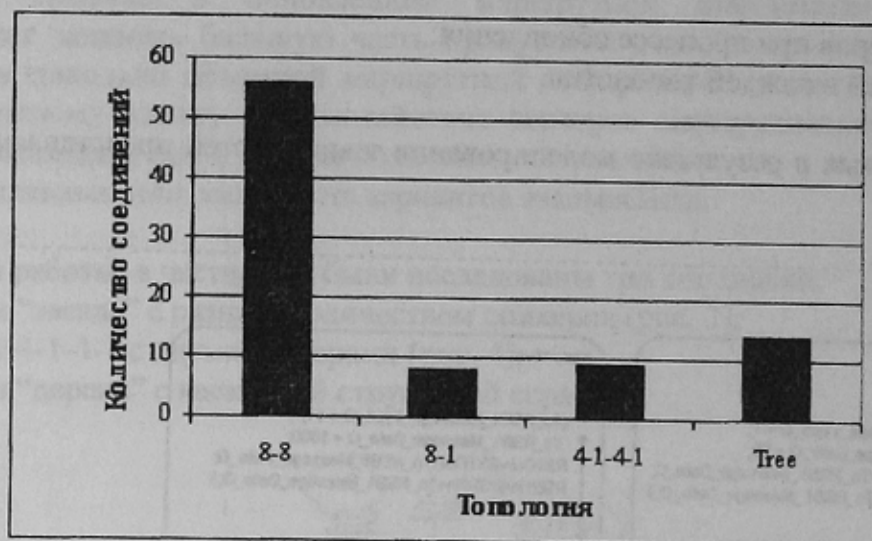


Рис. 7 - Количество связей при различных вариантах топологии

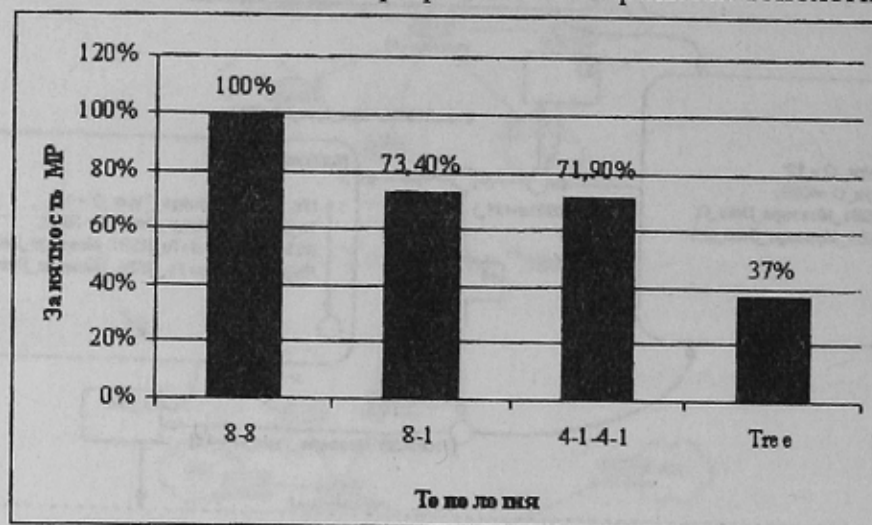


Рис. 8 - Занятость серверов при различных вариантах топологии

Из представленных результатов, в частности, видно, что для промежуточной топологии 4-1-4-1 эффективность многосерверной структуры вполне подтверждается результатами моделирования. Но наилучшим вариантом является все-таки древовидная топология, позволяющая получить самый низкий уровень средней занятости серверов.

Заклучение

Реализация аналогичным образом моделей сетевого взаимодействия как для других уровней стека протоколов TCP/IP, так и расширение возможностей разработанных моделей позволит осуществлять комплексное имитационное моделирование инфраструктуры сети при разработке различных вариантов распределенных моделирующих сред, для которых, как показывает практика, именно параметры сетевого обмена являются наиболее критичными.

В дальнейшем, в частности, в модель могут быть включены улучшенные механизмы для масштабирования правил маршрутизации и алгоритмы аутентификации, которые

позволяют обеспечить целостность адресной и другой информации между операторами сети. Важным моментом является также моделирование топологии точек доступа к сети для сетей более низкого уровня, то есть для доменов и соответственно для протоколов IGP как OSPF и RIP.

Список литературы

1. Аноприенко А.Я., Святный В.А. Высокопроизводительные информационно-моделирующие среды для исследования, разработки и сопровождения сложных динамических систем // Научные труды Донецкого государственного технического университета. Выпуск 29. Серия "Проблемы моделирования и автоматизации проектирования динамических систем" - Севастополь: «Вебер». - 2001. - С. 346-367.
2. Аноприенко А.Я., Джон С.Н. Задачи, методы и средства моделирования сетевой инфраструктуры // Научные труды Донецкого государственного технического университета. Выпуск 29. Серия "Проблемы моделирования и автоматизации проектирования динамических систем" - Севастополь: «Вебер». - 2001. - С. 312-319.
3. Аноприенко А.Я., Джон С.Н., Рычка С.В. Моделирование университетской сетевой инфраструктуры / Вісник Кременчуцького державного політехнічного університету. Наукові праці КДПУ. Випуск 2 (11). - Кременчук, КДПУ, 2001. С. 306-308.
4. Гультьев А. Визуальное моделирование в среде MATLAB. – СПб: Питер, 2000, 432с.
5. Кульгин М. Коммутация и маршрутизация IP/IPX-трафика. – СПб: Издательство «Компьютер Пресс», 1998.
6. Ohsaki H., Murata M., Ushio T., Miyahara H. A Control Teoretical Analysis of a Window-Based Flow Control Mechanism in TCP/IP Networks. In Proceeding of IEEE/ACM SIGCOMM'99, 1999, pp. 1-13.
7. Morris R. Scalable TCP Congestion Control. Thesis, Harvard University Cambridge Massachusetts, January 1999.

Поступила в редакційну колегію 01.02.2002 р.