

УДК 621.391

МЕТОДИКА ЭКСПЕРИМЕНТАЛЬНОГО ИССЛЕДОВАНИЯ МЕТОДОВ УПРАВЛЕНИЯ ТРАФИКОМ В СЕТИ



О.Ю. ВСЕЕВА

Харьковский национальный
университет радиоэлектроники

Викладено основні проблеми та методіку експериментального дослідження методів управління трафіком в телекомунікаційних мережах з використанням обладнання виробництва компанії Cisco Systems.

The basic problems and methodology of experimental study of traffic control methods in telecommunication network with using equipment manufactured by Cisco Systems are offered.

Описаны основные проблемы и методика экспериментального исследования методов управления трафиком в телекоммуникационных сетях с использованием оборудования производства компании Cisco Systems.

Введение

Одним из основных этапов синтеза новых, более эффективных методов управления трафиком телекоммуникационной сети (ТКС) является их экспериментальное исследование. При этом преследуются следующие основные цели:

- 1) проверка адекватности математических моделей, положенных основу предлагаемого метода управления трафиком;
- 2) оценка эффективности новой разработки путем ее сравнительного анализа с существующими практическими и теоретическими решениями.

В общем случае перечисленные задачи могут быть решены путем моделирования (аналитического или имитационного), а также путем проведения натурного или лабораторного эксперимента. При этом наибольшая достоверность с точки зрения последующего применения новой разработки на практике достигается именно в последнем варианте организации исследований.

В свою очередь проведение лабораторного эксперимента, в котором управление трафиком осуществляется в соответствии с некоторым новым, впервые предлагаемым методом, сопряжено с рядом организационных трудностей. Первая и основная проблема заключается в том, что в ходе эксперимента полученные с помощью предлагаемого метода результаты расчетов должны быть отображены на порядок использования маршрутов в реальной сети через настройки сетевых элементов. В перспективе предлагаемые модели и основанные на них методы решения задачи управления трафиком должны быть заложены в маршрутизаторы на уровне их операционных систем. Однако на этапе проведения лабораторного эксперимента исследователь вынужден использовать существующее, имеющееся в его распоряжении оборудование и программное обеспечение. При этом зачастую операционная система маршрутизаторов является закрытой, как например операционная система IOS (Internetwork

Operating System) маршрутизаторов *Cisco* [1], и реализует управление трафиком в соответствии с заложенными в нее производителем алгоритмами, которые нельзя заменить на произвольные (в нашем случае на исследуемый). Именно данный факт представляет наибольшую сложность при проведении лабораторного эксперимента и объясняет популярность различных средств имитационного моделирования, которым зачастую отдают предпочтение. Тем не менее, проведение лабораторного эксперимента по исследованию новых методов управления трафиком с использованием уже упомянутых маршрутизаторов *Cisco* является задачей реализуемой. Ниже подробно изложены основные проблемы и методика экспериментального исследования методов управления трафиком с задействованием сетевого оборудования производства компании *Cisco Systems*, которая была предложена и неоднократно апробирована сотрудниками лаборатории систем распределения информации кафедры телекоммуникационных систем ХНУРЭ [2, 3].

I. Структурная схема лабораторного эксперимента

Методика проведения эксперимента по исследованию вновь предлагаемых и существующих методов управления трафиком с использованием сетевого оборудования производства компании *Cisco Systems* предполагает реализацию следующих трех модулей (рис. 1):

- модуль 1 – конечная станция, работающая в режиме отправителя информационного трафика с заданными характеристиками;
- модуль 2 – фрагмент телекоммуникационной сети с управляемыми настройками сетевого оборудования;
- модуль 3 – конечная станция, которая работает в режиме получателя трафика и обеспечивает трассировку процесса получения пакетов. Этот же модуль выполняет функции обработки результатов с целью оценки показателей качества обслуживания (*QoS*-показателей), достигаемых в условиях исследуемого метода управления.

Техническая реализация каждого из модулей и их возможное программное обеспечение описаны в табл. 1. Для генерации, приема и обработка трафика конечными станциями в ходе лабораторного эксперимента предлагается использовать два программных пакета нагрузочного тестирования: *D-ITG* (*Distributed Internet Traffic Generation*), разработанный неапольским университетом им. Федерико II (Италия); и *IxChariot*, разработанный фирмой *Ixia* [4, 5]. Каждый в отдельности и тем более в совокупности данные пакеты предоставляют широкие возможности для исследования в реальных условиях новых механизмов и приложений, связанных с передачей трафика и управления им. Одновременное использование двух пакетов вместо одного обеспечит повышение достоверность получаемых в ходе эксперимента результатов и расширение перечня оцениваемых показателей качества обслуживания.

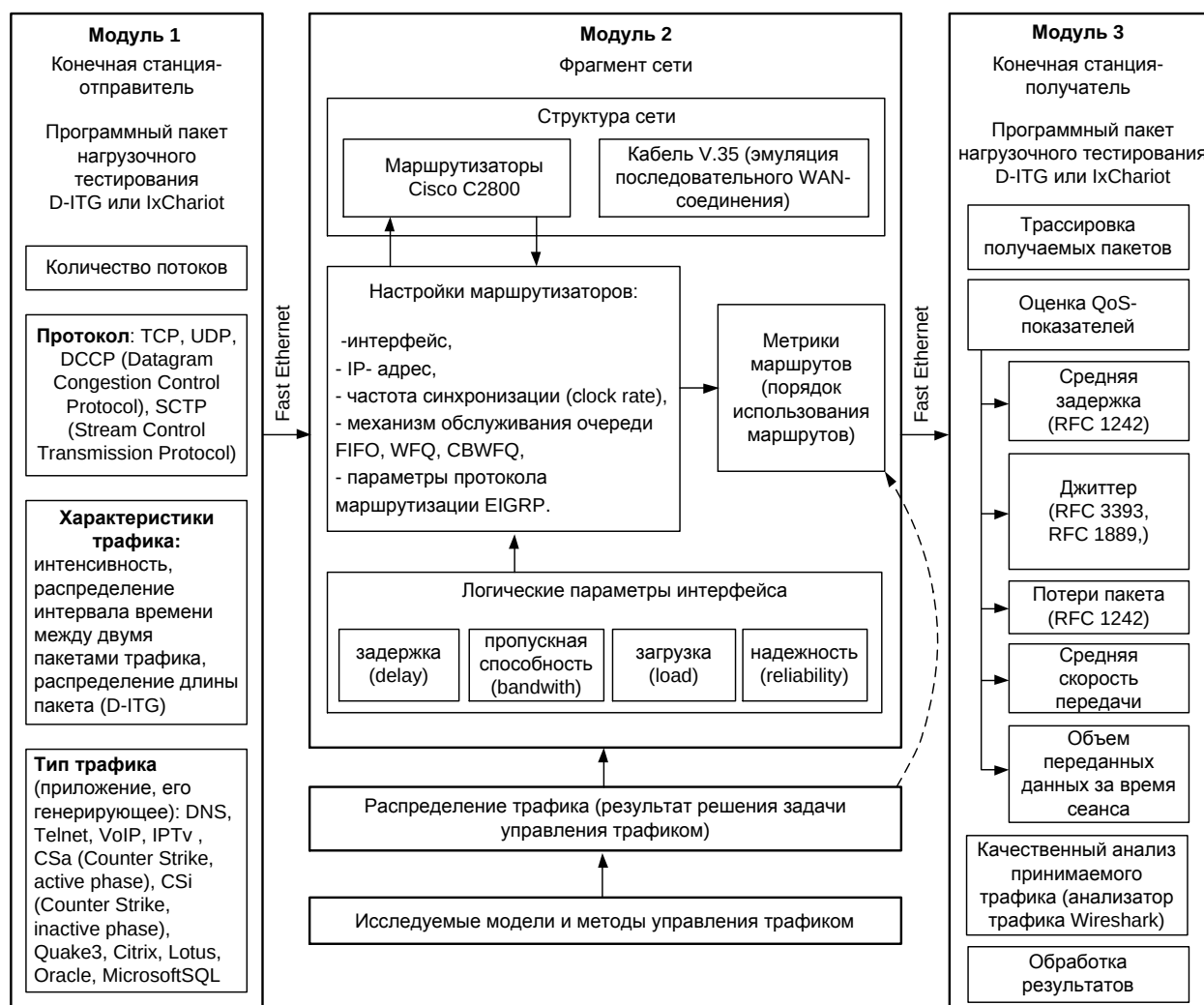


Рис. 1. Структурная схема проведения лабораторного эксперимента

Используемые в эксперименте пакеты нагрузочного тестирования *D-ITG* и *IxChariot* являются достаточно близкими по своим возможностям и, по утверждениям разработчиков [4, 5], позволяют получить оценки основных показателей качества обслуживания (средняя задержка передачи пакетов, вариация задержки (джиттер), коэффициент потерь пакетов, производительность) с высокой степенью достоверности. Пакет *IxChariot*, в отличие от *D-ITG*, обладает графическим интерфейсом и в целом является коммерческим продуктом, который предназначен для тестирования вводимых в эксплуатацию сетей. В свою очередь *D-ITG* в большей степени ориентирован на проведение научных исследований. Например, в рамках пакета *D-ITG* существует возможность указания в явном виде закона распределения интервала времени между соседними пакетами и закона распределения длины пакета.

Таблица 1. Реализация модулей лабораторного эксперимента

Модуль	Функция	Техническая реализация	Программное обеспечение	
			Вариант 1	Вариант 2
1. Конечная станция - отправитель	Генерация трафика с заданными характеристиками	Персональный компьютер с сетевым интерфейсом <i>Fast Ethernet</i>	<i>D-ITG</i> (<i>ITGSend.exe</i>)	<i>IxChariot</i>
2. Фрагмент сети	Передача трафика в соответствии с заданным порядком распределения по трактам передачи	Модульные маршрутизаторы <i>Cisco C2800</i> , кабели <i>V.35</i> для эмуляции последовательного (<i>serial</i>) <i>WAN</i> -соединения	Операционная система <i>IOS</i> , протокол маршрутизации <i>EIGRP</i>	
3. Конечная станция - получатель	Прием и обработка тестового трафика, качественный анализ принимаемого трафика, измерение его характеристик; обработка результатов измерений	Персональный компьютер с сетевым интерфейсом <i>Fast Ethernet</i>	Анализатор трафика <i>Wireshark</i>	
			<i>D-ITG</i> (<i>ITGRecv.exe</i> , <i>ITGDec.exe</i>) <i>Matlab</i>	<i>IxChariot</i>

Кроме того, использование двух пакетов, благодаря заложенным в них различным методикам оценки результирующей величины джиттера *IPDV* (*IP packet delay variation*), позволяет в зависимости от потребностей получить различные оценки данной величины: в рамках *D-ITG* согласно выражению

$$IPDV = \frac{1}{N_{отсчетов}} \sum_{i=1}^{N_{отсчетов}} IPDV_i ,$$

и в рамках *IxChariot* как [6] –

$$IPDV_{new} = IPDV_{old} - \frac{|IPDV_i| - IPDV_{old}}{16}, \quad i = \overline{1, N_{отсчетов}},$$

где $IPDV_i$ – джиттер i -го принятого пакета; $N_{отсчетов}$ – число отсчетов за время измерений; $IPDV_{old}$, $IPDV_{new}$ – предыдущее и последующие значения вариации $IPDV$.

Одним из принципиальных отличий пакета *IxChariot* является отсутствие в нем механизмов восстановления очередности пакетов трафика на приемной стороне. Как следствие, все пакеты, принятые с нарушением очередности (*out-of-order*), классифицируются как потерянные и учитываются при оценке коэффициента потерь пакетов. В отличие от *IxChariot* пакет *D-ITG* позволяет отследить количество непосредственно потерянных пакетов (*lost packet*), что послужило одной из причин выбора пакета *D-ITG* и его совместного с *IxChariot* использования при проведении лабораторного эксперимента.

Одной из серьезных проблем, которые возникли при организации лабораторного эксперимента, была проблема синхронизации конечных станций, что является

необходимым условием для получения адекватных оценок QoS-показателей. В рамках пакета нагрузочного тестирования *IxChariot* предусматривается три типа синхронизации: синхронизация по умолчанию (*Default clock synchronization*), обеспечиваемая встроенными алгоритмами синхронизации; аппаратная синхронизация, требующая подключения специальных, аппаратно реализованных модулей фирмы *Ixia* (*Ixia hardware clock synchronization*), и внешняя синхронизация (*External clock synchronization*).

В соответствии с документацией пакета *IxChariot* синхронизация по умолчанию является эффективной в случае симметричной маршрутизации, предполагающей, что пути для передачи трафика в прямом и обратном направлениях проходят через одни и те же маршрутизаторы. Однако зачастую подлежащие исследованию в ходе лабораторного эксперимента методы управления трафиком ТКС связаны с асимметричной маршрутизацией, что не позволяет использовать синхронизацию по умолчанию. Метод внешней синхронизации связан с использованием системного времени конечной станции, что предполагает синхронизацию самих конечных станций и что может быть осуществлено при помощи *NTP* (*Network Time Protocol*)-сервера. Как и в случае синхронизации по умолчанию, этот метод не рассчитан на асимметричную маршрутизацию. В отличие от *IxChariot* пакет *D-ITG* не имеет собственных средств синхронизации и для создания временных меток пакетов предусматривает использование времени операционной системы.

В сложившейся ситуации с целью синхронизации конечных станций, допускающей использование как пакета *IxChariot*, так и пакета *D-ITG*, в условиях асимметричной маршрутизации было принято следующее решение: использовать различными конечными станциями одно и то же системное время. Для этого две конечные станции физически были реализованы двумя сетевыми платами, подключенными к одному персональному компьютеру, где были одновременно запущены программы генерирования и передачи трафика, получения и обработки трафика, каждая из которых работала со своим сетевым интерфейсом. Передача тестового трафика между этими сетевыми интерфейсами через используемый в эксперименте фрагмент сети обеспечивалась соответствующими сетевыми настройками.

II. Особенности генерирования тестового трафика

Искусственное генерирование трафика, близкого к реальному, является одной из важнейших функциональных возможностей пакетов нагрузочного тестирования *D-ITG* и *IxChariot*, при этом в обоих пакетах предусматриваются широкие возможности по управлению параметрами тестового трафика. В целом, благодаря взаимодействию использованию двух пакетов, в зависимости от требований в ходе проведения лабораторного эксперимента можно варьировать:

- количеством потоков, передаваемых между конечными станциями;
- продолжительностью генерирования трафика;
- интенсивностью каждого отдельного потока (пак/с или бит/с);
- длиной пакетов трафика или их законом распределения;

- видом и параметрами закона распределения временного интервала между соседними пакетами (например, в *D-ITG* длина пакета и временной интервал между соседними пакетами могут быть распределены по равномерному, экспоненциальному, нормальному, Гамма-закону или по законам Парето, Коши, Пуассона);
- типом протокола транспортного уровня: *TCP, UDP, DCCP, SCTP*;
- типом трафика (имитация потока, создаваемого определенным протоколом прикладного уровня): *VoIP, IPTV, Telnet, DNS, Citrix, Lotus*, трафик баз данных (*Oracle, MicrosoftSQL*), игровой трафик (*Counter Strike, Quake3*).

Особенности интерфейсов различных пакетов отражаются на способах задания этих параметров. В рамках *D-ITG* управление экспериментом осуществляется при помощи командной строки, и необходимый набор параметров для генерирования трафика задается путем вызова программы *ITGSend.exe* с использованием ключей. Например, вызов программы *ITGSend.exe* для генерирования *TCP*-трафика (*-T TCP*) с адресом получателя 10.10.3.10 (*-a 10.10.3.10*) и интенсивностью трафика 80пак/с (*-C 80*), где длины пакетов распределены по равномерному закону и изменяются от 500байт до 1500 байт (*-u 500 1500*), продолжительностью генерации 180 с (*-t 180000*) выглядит следующим образом:

```
ITGSend.exe -a 10.10.3.10 -C 80 -u 500 1500 -T TCP -t 180000 -x 80pak.
```

При использовании пакета *IxChariot*, благодаря наличию графического интерфейса, задание параметров трафика сводится к выбору нужных опций, пример чего приведен на рис. 2.

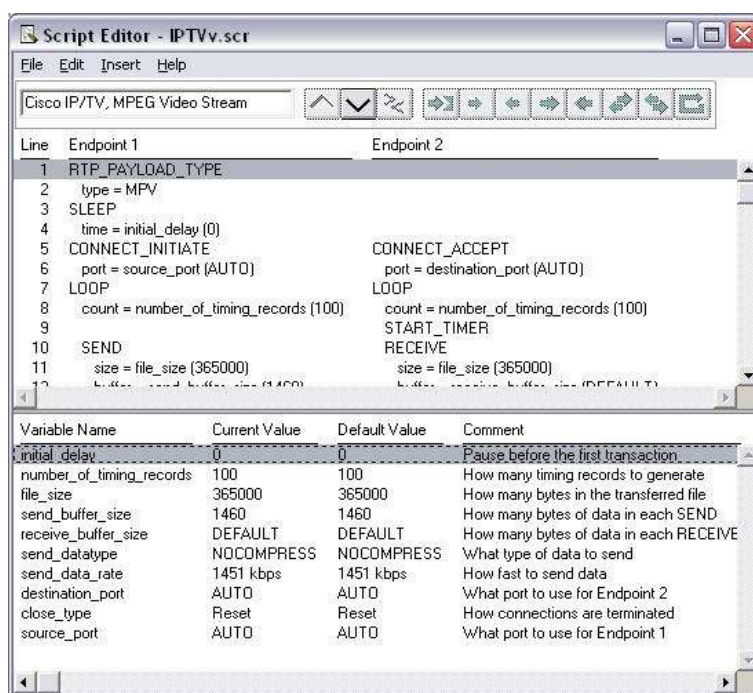


Рис. 2. Пример задания параметров *IPTV* трафика (протокол *RTP*) в пакете *IxChariot*

В целом пакеты *D-ITG* и *IxChariot* предоставляют широкие возможности по управлению параметрами тестового трафика, и его генерирование не вызывает дополнительных сложностей при проведении лабораторного эксперимента.

III. Порядок настройки сети под результаты аналитических расчетов

Как было отмечено выше, в ходе проведения лабораторного эксперимента результаты расчетов, полученные аналитическим путем с помощью некоторого исследуемого метода управления трафиком, должны быть реализованы в сети. При этом, учитывая закрытость используемой операционной системы, было предложено [2] реализовывать требуемый порядок использования маршрутов не через модернизацию *IOS*, а через настройки протокола маршрутизации *EIGRP*. Расширенный протокол внутреннего шлюза *EIGRP* является протоколом динамической маршрутизации, сочетающим в себе достоинства маршрутизации по состоянию канала и дистанционно-векторной. Среди других протоколов маршрутизации *EIGRP* выгодно выделяет его возможность гибкого манипулирования метриками интерфейсов, управляя которыми можно добиться требуемого порядка использования маршрутов в сети.

В целом параметры сети, подлежащие настройке в ходе лабораторного эксперимента, можно разделить на две части: независящие от исследуемого порядка использования маршрутов (*IP*-адреса, пропускные способности интерфейсов) и зависящие от него (настройки протокола *EIGRP*). На рис. 3 для примера приведен фрагмент сети с указанием настроек первой группы.

Известно [7, 8], что протокол *EIGRP* использует при выборе маршрута сложную метрику

$$M_{EIGRP} = \left[K_1 M_b + \frac{K_2 M_b}{256 - M_l} + K_3 M_d \right] \frac{K_5}{M_r + K_4}, \quad (1)$$

где K_1, K_2, K_3, K_4, K_5 – постоянные коэффициенты; M_b, M_d, M_r, M_l – отдельные метрики маршрута, причем M_b – величина обратной логической пропускной способности маршрута; M_d – кумулятивная логическая задержка маршрута, M_r и M_l – надежность и загрузка маршрута соответственно.

Весовые коэффициенты K_1, K_2, K_3, K_4, K_5 (1) являются административно настраиваемыми параметрами, по умолчанию их значения равны $K_1 = 1, K_2 = 0, K_3 = 1, K_4 = 0, K_5 = 0$. В результате формула (1) принимает вид:

$$M_{EIGRP} = K_1 M_b + K_3 M_d. \quad (2)$$

В рамках операционной системы маршрутизаторов *IOS* параметры M_b и M_d являются логическими, и их административное назначение не сказывается на значениях соответствующих физических параметров. Например, изменение значения логической задержки M_d^{int} на каком-либо интерфейсе никоим образом не повлияет на реальную задержку передачи пакетов трафика через этот физический интерфейс.

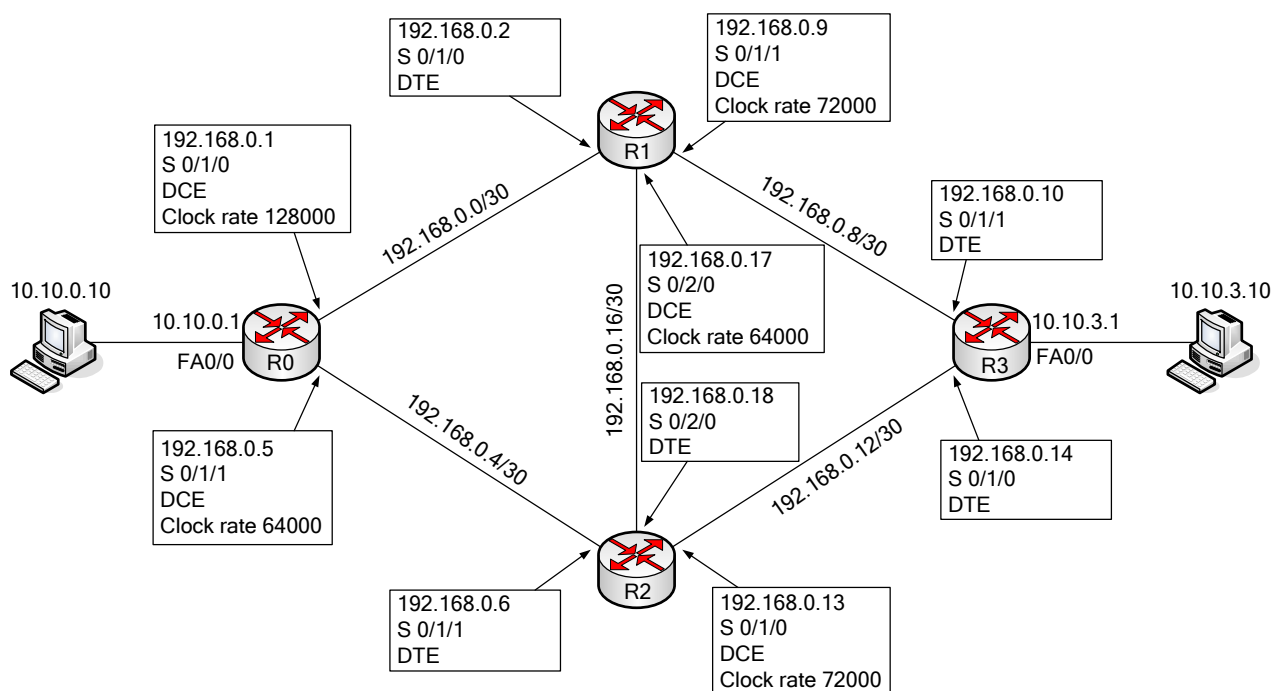


Рис. 3. Пример фрагмента сети с указанием основных настроек интерфейсов

Таким образом, путем административного назначения метрик интерфейсов M_b^{int} и M_d^{int} существует возможность управлять использованием различных маршрутов и тем самым получить требуемое распределение трафика. В связи с трудностями подстройки одновременно двух параметров коэффициент K_1 (2) приравнивался нулю, и метрика *EIGRP*-маршрута полностью определялась метриками M_d^{int} , указанными на интерфейсах ($M_{EIGRP} = K_3 M_d$).

В отличие от большинства протоколов маршрутизации протокол *EIGRP* обеспечивает балансировку нагрузки между маршрутами с неравной стоимостью. Для этого в маршрутной таблице протокола *EIGRP* хранится информация о маршрутах как с минимальной метрикой, так и с большими ее значениями. Маршруты, которые имеют большее значение метрики, но сохраняются в маршрутной таблице и используются затем для передачи трафика, называются маршрутами-кандидатами (*candidate route*). Для того чтобы маршрут стал маршрутом-кандидатом и использовался впоследствии для передачи трафика, необходимо выполнение двух условий [7, 8]. Первое условие требует, чтобы следующий маршрутизатор маршрута-кандидата имел свой маршрут к сети назначения с метрикой меньшей, чем метрика лучшего маршрута. Например, если маршрутизатор *A* имеет лучший маршрут к маршрутизатору *D* с метрикой M_A и рассматривает возможность использования в качестве маршрута-кандидата к маршрутизатору *D* маршрут через *B* с метрикой M_B , тогда необходимым условием его принятия будет наличие на маршрутизаторе *B* маршрута к *D* с метрикой $M'_B < M_A$ (рис. 4).

Второе условие определяет допустимое численное соотношение между метрикой лучшего маршрута M_A и метриками маршрутов-кандидатов M_B :

$$M_B \leq k_{\text{var}} M_A, \quad (3)$$

где k_{var} – административно задаваемый коэффициент (множитель, *variance*), который принимает целочисленное значение в интервале от 1 до 128.

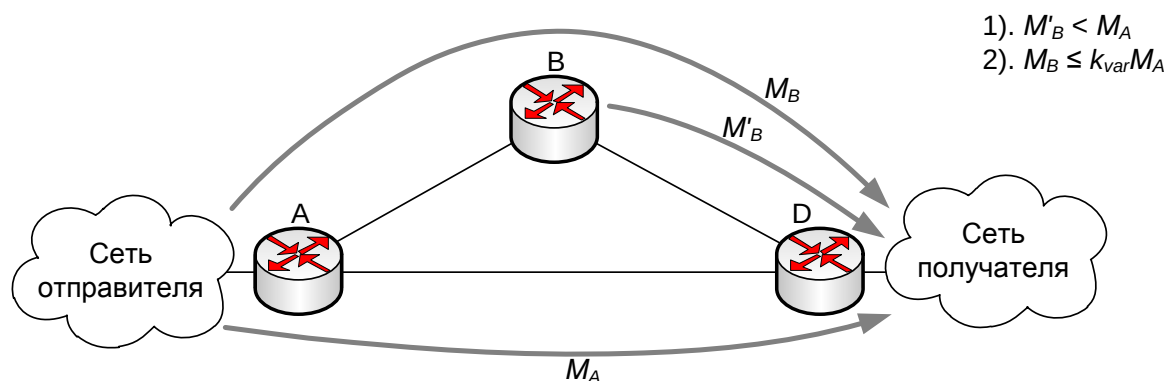


Рис. 4. Условия использования маршрутов с неравной стоимостью в протоколе EIGRP

Второе условие позволяет протоколу EIGRP динамически адаптироваться к изменениям в сети. Если произошедшие изменения влекут за собой ухудшение маршрута-кандидата, то нарушается условие (3) и данный маршрут исключается из таблицы. Напротив, если появляется новый маршрут, метрика которого удовлетворяет условию (3), то он становится маршрутом-кандидатом.

Если маршрутизатор EIGRP настроен на балансировку нагрузки между маршрутами с неравной стоимостью, то соотношение между метриками маршрутов M_A и M_B определяет частоту их использования: маршруты используются не равномерно, а приблизительно в том же соотношении, как и их метрики.

Как уже было отмечено, реализация требуемого распределения трафика по трактам передачи в ходе проведения лабораторного эксперимента обеспечивается подстройкой для каждого интерфейса одного параметра – его логической задержки $M_d^{\text{инт}}$. Основываясь на описанных выше условиях использования маршрутов с неравной стоимостью, выведем правила для расчета параметров $M_d^{\text{инт}}$ для каждого из интерфейсов фрагмента сети, например, представленного на рис. 5, которые в совокупности будут обеспечивать требуемое использование маршрутов Λ . При этом предполагается, что вектор Λ формируется предварительно аналитическим путем и указывает на интенсивность трафика λ^i , которая должна быть направлена в i -й тракт передачи в соответствии с исследуемым методом управления трафиком.

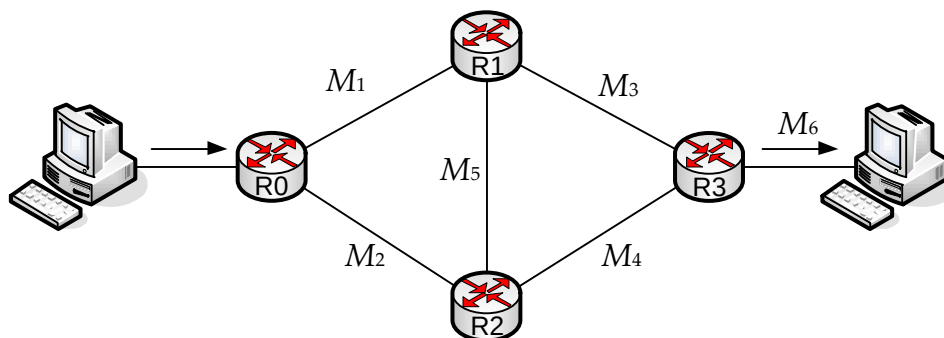


Рис. 5. Фрагмент используемой в лабораторном эксперименте сети
и метрики ее трактов передачи

Для данного фрагмента сети (рис. 5) в сегменте $R1, R2, R3$ возможны следующие ситуации:

- 1) $M_5 + M_4 + M_6 > M_3 + M_6$, где M_i – метрика i -го тракта передачи, которая определяется значением M_d^{int} , установленным на соответствующем интерфейсе. В этом случае для маршрутизатора $R1$ кратчайшим является маршрут через $R3$;
- 2) $M_3 + M_6 > M_5 + M_4 + M_6$, тогда для $R1$ кратчайшим является маршрут через $R2$;
- 3) $M_3 + M_6 = M_5 + M_4 + M_6$, тогда маршруты, проходящие из $R1$ через $R2$ и $R3$, являются маршрутами равной стоимости и используются равнозначно.

Допустим, выполняется первое условие, и кратчайший маршрут из $R1$ к сети назначения проходит через $R3$. Тогда, чтобы организовать дополнительный маршрут через $R2$, необходимо добиться выполнения следующих условий:

$$M_3 + M_6 > M_4 + M_6, \quad (4)$$

$$\frac{M_5 + M_4 + M_6}{M_3 + M_6} = \frac{\lambda^3}{\lambda^5}, \quad (5)$$

где λ^3, λ^5 – значения интенсивностей трафика, направляемого из маршрутизатора $R1$ в третий и пятый тракты передачи через интерфейсы $S0/1/1$ и $S0/2/0$ соответственно (рис. 3), являются целыми числами, так как определяют число пакетов, направляемых в тот или иной интерфейс.

Если для маршрутизатора $R1$ кратчайшим является маршрут через $R2$, то для дополнительного условия использования маршрута через $R3$ необходимыми являются следующие условия:

$$M_5 + M_4 + M_6 > M_6, \quad (6)$$

$$\frac{M_3 + M_6}{M_5 + M_4 + M_6} = \frac{\lambda^5}{\lambda^3}. \quad (7)$$

Для балансировки трафика между интерфейсами S0/1/0 и S0/1/1 маршрутизатора R0 (первый и второй тракты передачи) в требуемом соотношении λ^1/λ^2 должны выполняться условия:

- если для R0 кратчайшим является маршрут через маршрутизатор R1, то

$$M_1 + M_{R1} > M_{R2}, \quad \frac{M_2 + M_{R2}}{M_1 + M_{R1}} = \frac{\lambda^1}{\lambda^2}, \quad (8)$$

- если для R0 кратчайшим является маршрут через маршрутизатор R2, то

$$M_2 + M_{R2} > M_{R1}, \quad \frac{M_1 + M_{R1}}{M_2 + M_{R2}} = \frac{\lambda^2}{\lambda^1}, \quad (9)$$

где M_{R1} , M_{R2} – метрики кратчайших маршрутов до сети назначения, известные маршрутизаторам R1 и R2 соответственно, которые они объявляют маршрутизатору R0 (*Advertised Distance, AD*).

Таким образом, для реализации в данном фрагменте сети (рис. 5) между отправителем и получателем трех маршрутов:

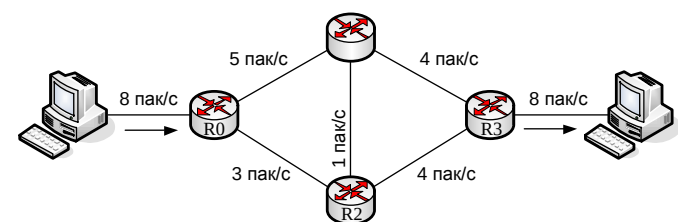
$$R0 - R1 - R3,$$

$$R0 - R1 - R2 - R3,$$

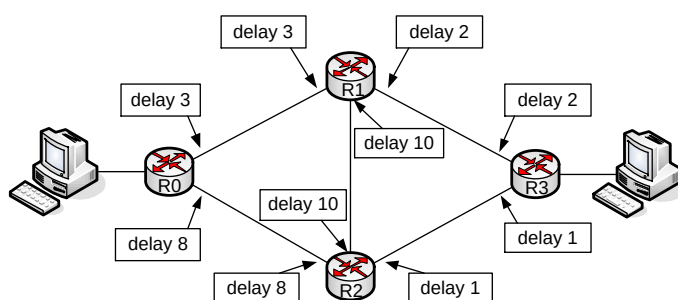
$$R0 - R2 - R3,$$

где частота использования каждого из маршрутов определяется заданным распределением трафика Λ , достаточно найти целые числа $M_1, \dots, M_6$, удовлетворяющие (4)–(9).

Рассмотрим пример применения описанного подхода. Для данного фрагмента сети (рис. 5) результатом решения задачи многопутевой маршрутизации с использованием тензорной модели [9] при интенсивности трафика 8 пак/с является вектор $\Lambda = [5,15 \ 2,85 \ 4,46 \ 3,54 \ 0,69]^T$. Поскольку в рамках протокола EIGRP осуществляется по пакетной балансировка трафика, полученные значения должны быть, в первую очередь, округлены до ближайшего целого числа $\Lambda = [5 \ 3 \ 4 \ 4 \ 1]^T$ (рис. 6, а), т.е. имеем $\frac{\lambda^1}{\lambda^2} = \frac{5}{3}$, $\frac{\lambda^3}{\lambda^5} = \frac{4}{1}$. Тогда из условий (4)–(9) находим $M_1 = 3$, $M_2 = 8$, $M_3 = 2$, $M_4 = 1$, $M_5 = 10$, $M_6 = 1$, которые представляют собой настраиваемые на интерфейсах значения задержки (*delay*) M_d^{unim} (рис. 6, б). Листинг конфигурационного файла маршрутизатора R0, где настраиваются метрики $M_1 = 3$ и $M_2 = 8$, приведен на рис. 6, в, а результирующие маршрутные таблицы на рис. 7.



а)



б)

```
R0#sh running-config
Building configuration...
Current configuration : 1690 bytes
!
version 12.4

no service password-encryption
!
hostname R0
!
enable secret 5 $1$8kSv$RixjKpOJ8huH7v2I8AGLQ.
!
no ip cef
!
interface FastEthernet0/0
 ip address 10.10.0.1 255.255.255.0
 no ip route-cache
 duplex auto
 speed auto
!
interface FastEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface Serial0/1/0
 bandwidth 128
 ip address 192.168.0.1 255.255.255.252
 no ip route-cache
 delay 3
 clock rate 128000
!
interface Serial0/1/1
 bandwidth 64
 ip address 192.168.0.5 255.255.255.252
 no ip route-cache
 delay 8
 clock rate 64000
!
router eigrp 1
 variance 4
 network 10.0.0.0
 network 192.168.0.0
 metric weights 0 0 0 1 0 0
 no auto-summary
!
end
```

в)

Рис. 6. Пример распределения трафика (а); метрики, его реализующие, (б)
и листинг конфигурационного файла маршрутизатора R0 (в)

В ходе проведения лабораторного эксперимента принятые к обслуживанию пакеты на каждом маршрутизаторе могут обрабатываться с помощью различных механизмов обслуживания очередей, например, *FIFO*, *WFQ*, *CBWFQ* или иных (рис. 1) [10]. Таким образом, существует возможность исследовать не только сам метод управления трафиком, но и его взаимодействие с различными механизмами обслуживания очередей с целью выбора наиболее эффективной их комбинации.

```
R0#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

 10.0.0.0/24 is subnetted, 2 subnets
C      10.10.0.0 is directly connected, FastEthernet0/0
D      10.10.3.0 [90/2560] via 192.168.0.6, 00:42:41, Serial0/1/1
        [90/1536] via 192.168.0.2, 00:42:41, Serial0/1/0
 192.168.0.0/30 is subnetted, 5 subnets
D      192.168.0.8 [90/2816] via 192.168.0.6, 00:42:41, Serial0/1/1
        [90/1280] via 192.168.0.2, 00:42:41, Serial0/1/0
D      192.168.0.12 [90/2304] via 192.168.0.6, 00:42:41, Serial0/1/1
        [90/1536] via 192.168.0.2, 00:42:41, Serial0/1/0
C      192.168.0.0 is directly connected, Serial0/1/0
C      192.168.0.4 is directly connected, Serial0/1/1
D      192.168.0.16 [90/4608] via 192.168.0.6, 00:40:37, Serial0/1/1
        [90/3328] via 192.168.0.2, 00:40:37, Serial0/1/0
R0#
```

```
R0#show ip route 10.10.3.0
Routing entry for 10.10.3.0/24
  Known via "eigrp 1", distance 90, metric 1536, type internal
  Redistributing via eigrp 1
  Last update from 192.168.0.6 on Serial0/1/1, 03:22:19 ago
  Routing Descriptor Blocks:
    192.168.0.6, from 192.168.0.6, 03:22:19 ago, via Serial0/1/1
      Route metric is 2560, traffic share count is 3
      Total delay is 100 microseconds, minimum bandwidth is 64 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 2
    * 192.168.0.2, from 192.168.0.2, 03:22:19 ago, via Serial0/1/0
      Route metric is 1536, traffic share count is 5
      Total delay is 60 microseconds, minimum bandwidth is 72 Kbit
      Reliability 255/255, minimum MTU 1500 bytes
      Loading 1/255, Hops 2
R0#
```

Рис. 7. Маршрутные таблицы маршрутизатора R0

IV. Методика сбора и обработки результатов лабораторного эксперимента

При проведении лабораторного эксперимента необходимо осуществлять как качественный, так и количественный анализ процесса обслуживания трафика. При этом с целью качественного анализа процесса передачи может быть использована возможность управления маркировкой пакетов. Например, все пакеты, передаваемые через интерфейс S0/1/1 маршрутизатора R0, т. е. пакеты, которые должны следовать вдоль маршрута R0 – R2 – R3, могут быть маркированы как принадлежащие к классу немедленной пересылки *EF* (*Expedited Forwarding*), а пакеты, передаваемые через интерфейс S0/1/0 этого же маршрутизатора и интерфейс S0/1/1 маршрутизатора R1, т.е. передаваемые по маршруту R0 – R1 – R3, не иметь маркировки (*DSCP*=0). Маркировка *DSCP* = *AF11* может быть связана с маршрутом R0 – R1 – R2 – R3. Это

позволит при помощи анализатора трафика, например, *Wireshark*, на приемной стороне по значениям *DSCP* принятых пакетов отследить порядок использования различных маршрутов и убедиться в соответствии наблюдаемого порядка использования маршрутов требуемому.

С целью количественного анализа функционирования сети в условиях исследуемого метода управления целесообразно воспользоваться возможностями пакетов нагрузочного тестирования *D-ITG* и *IxChariot*, которые обеспечивают не только расчет результирующих показателей *QoS* (среднего времени задержки, джиттера, коэффициента потерь пакетов и производительности), но и предоставляют возможность отследить их динамику в ходе проведения эксперимента. Например, на рис. 8 и 9 приведены зависимости, отображающие динамику межконцевой задержки и полученные для фрагмента сети (рис. 3) при помощи пакетов *IxChariot* (рис. 8) и *D-ITG* (рис. 9) в условиях трех способов маршрутизации: однопутевой в рамках протокола *EIGRP* с настройками по умолчанию (*EIGRP default*), многопутевой в рамках *EIGRP* с использованием путей неравной стоимости (*EIGRP variance*) и многопутевой в соответствии с тензорной $[NP, \pi\omega]$ -моделью (*tensorial*), описанной в [9].

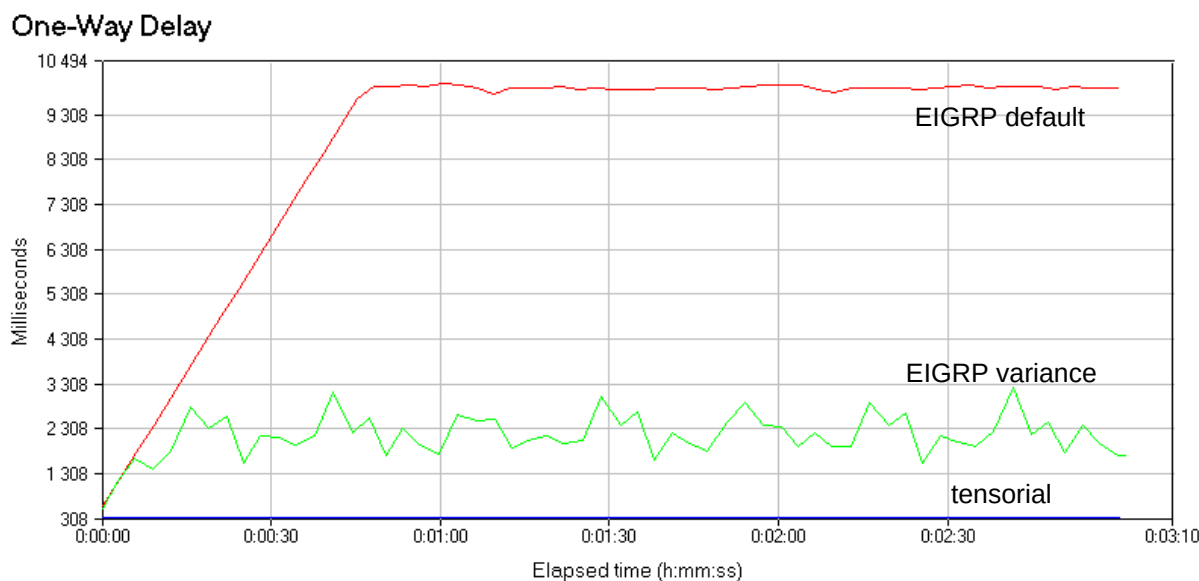


Рис. 8. Динамика межконцевой задержки, полученная при помощи пакета *IxChariot* для входного *IPTV*-трафика, который имел интенсивность 8 пак/с (85 кбит/с), длину пакета 1332 байта и передавался по протоколу *RTP*

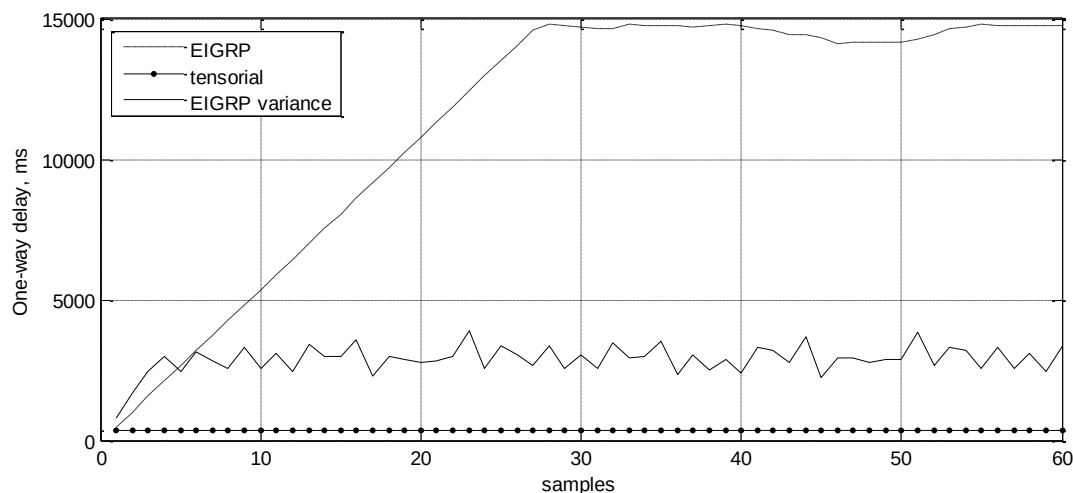


Рис. 9. Динамика межконцевой задержки, полученная при помощи пакета *D-ITG* для входного трафика, который имел интенсивность 8 пак/с (85 кбит/с), длину пакета 1332 байта и передавался по протоколу *UDP*

Таким образом, в итоге имеем возможность количественно оценить эффективность функционирования сети в условиях исследуемого метода управления трафиком и на основании сравнительного анализа с существующими решениями сделать определенные выводы. Например, из зависимостей, представленных на рис. 8 и 9, следует, что в случае однопутевой маршрутизации с ростом загруженности сети, начиная с некоторого момента времени (на рис. 8 это 45-я секунда эксперимента), сетевые устройства вдоль используемого маршрута входят в режим перегрузки, и средняя задержка достигает своего наибольшего возможного значения. Значительно лучшая ситуация складывается при многопутевой маршрутизации (*EIGRP variance* и *tensorial* на рис. 8 и 9). Наблюдаемые при этом колебания средней задержки связаны с переключениями маршрутов, что особо четко видно в случае *EIGRP* с балансировкой неравной стоимости (*EIGRP variance*). При реализации маршрутизации в соответствии с тензорной $[NP, \pi\omega]$ -моделью ТКС переключение маршрутов хоть и имеет место, однако не вызывает таких сильных скачков *QoS*-показателей. Более того, при этом наблюдается наименьшая величина средней задержки (рис. 8 и 9).

При этом результаты оценки величины задержки из конца в конец, полученные при помощи различных программных продуктов *D-ITG* и *IxChariot*, как видно из графиков (рис. 8 и 9), весьма близки. Так, для фрагмента сети, приведенного на рис. 3, в условиях входного *UDP*-трафика с интенсивностью 8 пак/с (85 кбит/с) при длине пакета 1332 байта и реализации маршрутизации в соответствии с предлагаемой тензорной моделью средняя задержка по результатам *IxChariot* составляет 308 мс и 331 мс по результатам *D-ITG*.

Выводы

Применение данной методики позволяет отслеживать динамику состояния сети в условиях реализуемого с помощью того или иного метода управления трафи-

ком при различных интенсивностях и типах сетевой нагрузки. В рамках методики заложена возможность оценить качество сетевого управления при использовании различных механизмов обслуживания очередей, профилирования трафика, отбрасывания пакетов и в условиях динамического изменения конфигурации сети.

Описанная в статье методика экспериментального исследования моделей и методов управления трафиком телекоммуникационной сети была неоднократно апробирована на лабораторных занятиях кафедры телекоммуникационных систем ХНУРЭ, а также при проведении научных, в т.ч. диссертационных исследований сотрудниками (аспирантами, докторантами, соискателями и др.) кафедры. Методика зарекомендовала себя как надежный инструмент для проверки адекватности математических моделей управления трафиком в сети и достоверности получаемых на их основе количественных результатов исследования – загруженности сети и основных показателей качества обслуживания. В целом же применение данной методики на финальных стадиях разработки новых методов управления трафиком позволит уточнить результаты аналитического и имитационного моделирования, связанные с проверкой принятых допущений и упрощений.

Список литературы:

1. Cisco IOS Configuration Fundamentals and Network Management Configuration Guide, Release 12.3. – Режим доступа: http://www.cisco.com/en/US/docs/ios/12_3/featlist/cfun_vcg.html
2. Симоненко Д.В., Андрушко Ю.В., Беленков А.Г. Методика проверки адекватности моделей многопутевой и иерархической маршрутизации с использованием сетевого оборудования компании Cisco Systems // Радиотехника: Всеукр. межвед. науч.-техн. сб. – 2007. – Вып. 151. – С. 37 – 43.
3. Симоненко Д.В., Саенко В.А., Добрышкин Ю.Н. Особенности проведения натурного эксперимента на сетевом оборудовании в ходе сравнительного анализа различных моделей и методов маршрутизации // Радиотехника: Всеукр. межвед. науч.-техн. сб. – 2009. – Вып. 159. – С. 234 – 238.
4. Avallone S., Guadagno S., Emma D. [and other] D-ITG distributed Internet traffic generator // First International Conference on the Quantitative Evaluation of Systems, QEST 2004, 27 – 30 Sept. 2004: proceedings of the conference. – P. 316 – 317.
5. IxChariot Performance Endpoints Release 7.10 913-0951 Rev. A, December 2009.
6. Schulzrinne H., Casner S., Frederick R., Jacobson V. RTP: A Transport Protocol for Real – Time Applications. RFC 1889. – 1996. – 75 p.
7. Руденко И. Tsunami Computing, Маршрутизаторы CISCO для IP-сетей. – М.: Кудиц-Образ, 2003. – 656 с.
8. Doyle J., Carroll J. CCIE Professional Development Routing TCP/IP. – Vol. I. – [2nd edition]. – Cisco Press, 2005. – 936 p.
9. Евсеева О.Ю. Тензорная модель гарантированного обеспечения нормированных показателей качества обслуживания в ТКС // Наукові записки УНДІЗ. – 2008. – № 3 (5). – С. 71 – 81.
10. Вегенша III. Качество обслуживания в сетях IP. – М.: Изд. дом «Вильямс», 2003. – 386 с.