

УДК 021.8 + 025.1

## ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ ТРАФИКА В ВЫЧИСЛИТЕЛЬНЫХ СЕТЯХ НА ОСНОВЕ АППАРАТА СЕТЕЙ ПЕТРИ

*M. V. Семехина*

### SIMULATION MODELING OF THE TRAFFIC TRANSFER PROCESSES IN COMPUTING NETWORKS BASED ON PETRI NETWORKS

*M. V. Semekhina*

В работе представлен комплексный подход к моделированию локальных вычислительных сетей (ВС) для их качественного анализа. В частности, для имитационного моделирования предлагается использование раскрашенных временных сетей Петри (СП) с очередями. Вводится понятие ролевых функционалов и операций над ними. Приводится пример построения СП для фрагмента локальной сети, предложен общий алгоритм и результаты работы имитационной модели.

We consider the way of simulation modeling of traffic in computing networks. Such approach uses colored Petri networks with time mechanism and queues. We also offer the definition of role functional and operation on them and presents the algorithm and results of

**Ключевые слова:** моделирование вычислительных сетей, теория массового обслуживания, имитационное моделирование, сети Петри.

**Keywords:** modeling of the computing networks, queuing theory, simulation modeling, Petri network

Высокие требования к качеству работы сети обеспечивают интерес к мониторингу и моделированию вычислительных сетей. Средства мониторинга доступны сегодня в широком ассортименте, начиная от системных консольных утилит, заканчивая программными комплексами с возможностью управления и с применением экспертных систем (HP OpenView NNM [11], NetView (Tivoli) от IBM, Spectrum от Cabletron). В области моделирования ВС дело обстоит иначе, поскольку здесь необходимы не только технические знания, но и понимание теоретических основ процессов передачи и обработки данных, происходящих в вычислительных сетях. В таблице 1 представлены результаты сравнения распространенных средств анализа и моделирования ВС.

Высокая стоимость, сложность внедрения и интеграции, ограниченные возможности некоторых существующих программных продуктов, в совокупности с существующим спросом, заставляет исследователей искать новые решения для анализа и моделирования ВС [2, 3, 5, 8, 10, 12]. Фрактальные свойства сетевого трафика исследуются в работах С. В. Ильницкого [3] и коллектива американских исследователей (M. S. Taqqu, D. V. Wilson, W. E. Leland) [12]. В Пермском государственном университете разрабатывается распределенная имитационная система Triad.Net [5], которая использует трехуровневое описание имитационной модели в терминах структур, процедур, описывающих алгоритм взаимодействия структур, и сообщений, которыми обмениваются структуры. Исследователи из Пензенского государственного университета (Н. П. Вашкевич, В. Н. Дубинин, С. А. Зинкин) [2] формализуют ВС с помощью языка описания сетевых моделей (ЯОСМ) и далее используют статистические методы обработки полученной модели. Методология гибридного моделирования ВС на основе аналитиче-

ских методов и дискретных систем подробно рассматривается в работе А. Ф. Ярославцева [10].

Интерес к данной теме со стороны ведущих ИТ-компаний и независимых ученых определяет актуальность исследований в этой области.

В настоящей статье рассматриваются аспекты имитационного моделированию ВС на базе сетей Петри. Предлагаемый подход используется авторами в комплексе с аналитическими методами, что позволяет всесторонне провести анализ работы существующей или проектируемой сети.

На основе описанного подхода и комплексного использования методов аналитического и имитационного моделирования ВС, дополняющих друг друга, создан прототип информационной системы, **основными задачами** которой являются:

1. Создание адекватных моделей ВС на основе данных трех типов: а) данные о топологии сети; б) данные о маршрутах и характеристиках потоков трафика; в) данные о вычислительных мощностях сетевых устройств.
2. Аналитический расчет сетевых параметров для представленных моделей.
3. Проведение имитационного эксперимента по передаче трафика в рамках построенной модели сети.
4. Анализ полученных результатов с целью выявления «критичных» участков сети с низкой или нестабильной скоростью обработки/передачи сообщений.

Система позволит оценить ВС на этапе ее проектирования или модернизации, а также будет полезна для оценки эффективности уже функционирующих сетей и выработки рекомендаций для их оптимизации.

Таблица 1  
Сравнительные характеристики автоматизированных средств анализа и моделирования ВС

<i>Продукт</i>	<i>COMNET</i>	<i>NetMaker</i>	<i>OPNet</i>	<i>OMNeT++</i>	<i>SES/Strategizer</i>	<i>NS-2</i>	<i>NetCracker</i>
Производитель	CACI Products	Make Systems	OPNET		Scientific and Engineering Software		Net Cracker technology
Наличие библиотеки устройств	+	+	+	+	+	+	+
Настройка библиотеки устройств	+	+	-	+	+	+	-
Пошаговая трассировка	+	-	-	+	+	+	+
Генерация отчетов	+	+	-	+	+	-	+
Прочие особенности	Возможность ввода данных о трафике в режиме реального времени, гипотез о трафике. Возможность тонкой подстройки параметров сети. Наличие дополнительных модулей для различных задач	Высокая производительность; наличие библиотек устройств разных производителей	Удобство проектирования объектов сети, гибкость полученных моделей	Симулятор дискретных событий, обладает высоким уровнем абстракции. Для моделирования используется язык описания структуры модели NED и C++	Простота применения. Ясность схем сети	Использование двухуровневого программирования (язык C++ и OTcl) и объектно-ориентированного подхода обеспечивает гибкую настройку моделей и сценариев их работы	Простота построения схем сети и настройка параметров сетевых компонент и потоков трафика; анимация
Стоимость, долл	от 20 000 за базовую конфигурацию	от 40 000	от 17 000	бесплатно	от 10 000	бесплатно	бесплатно
Недостатки	Сложность внедрения. Трудность восприятия схем сети. Несоответствие некоторых отчетов, высокая стоимость	Высокая стоимость, потребность в дополнительном ПО. Требуется дополнительное обучение	Высокая стоимость, слабые возможности импорта и настройки компонент продукта	Проблемы с внедрением, сложность обучения пользователей	Потребность в дополнительном ПО. Слабая визуализация	Слабая визуализация; сложность моделей и анализа результатов	Слабые возможности анализа информации; невозможность импорта данных из реальной сети

### Математическая модель ВС на основе СП

Качество работы ВС чаще всего, оценивают по следующим основным параметрам [9]: пропускная способность (емкость); скорость; надежность.

Для моделирования используются иерархические раскрашенные СП с временным механизмом [4, 6], где цвет (тип меток) соответствует типу трафика. Время в модели принимается дискретным и измеряется тактами. В течение одного такта метка может совершить не более одного перемещения через разрешенный переход. Каждый сетевой объект (рабочая станция, канал передачи, коммуникационное устройство) моделируется как подсеть Петри.

*Основные принципы построения модели:*

1. Теоретико-множественное определение СП можно представить в следующем виде [4,6]:

$$W=\{P, T, I, O, M\},$$

где  $P=\{p_1, \dots, p_n\}$  – конечное множество позиций;  $T=\{t_1, \dots, t_m\}$  – конечное множество переходов сети;  $I:T \rightarrow P^*$  – входная функция инцидентности, сопоставляющая переходу множество его входных позиций ( $I(t, p)=1$  если  $p$  – входная позиция для  $t$ ,  $I(t, p)=0$  в противном случае);  $I^+:T \rightarrow P^*$  – выходная функция инцидентности, сопоставляющая переходу множество его выходных позиций ( $I^+(t, p)=1$  если  $p$  – выходная позиция для  $t$ ,  $I^+(t, p)=0$  в противном случае);  $M=(M(p_1), \dots, M(p_n))$  – вектор распределения меток по позициям (маркировка).

Однако для целей моделирования ВС в нашем случае удобнее ввести следующее определение:

$$W = \{P, T, D, TR\},$$

где  $P$  – множество позиций;  $T$  – множество переходов;  $D$  – множество дуг;  $TR$  – множество типов трафика;  $TR(p)$  – множество типов меток, которые могут находиться в позиции  $p$ ;  $m(tr, p)$  – метка типа  $tr$ , находящаяся в позиции  $p$ .

2.  $\forall (t, r) t \in T, r \in TR \exists q \in R$  – временная задержка на каждом переходе определяется для каждого типа трафика.

3. Введем понятие очереди для типов меток для описания ситуации, когда в момент обработки метки во входную позицию перехода могут поступать новые метки разных типов. Для однородных СП, когда все метки однотипны, понятие очереди аннулируется, ибо с точки зрения СП метки одного типа принципиально не отличаются друг от друга и для перемещения выбирается любая из них. Если же через позицию проходит смешанный трафик, выбор типа метки для перехода должен учитывать очередность их поступления:

$$\begin{aligned} \forall p \in P, \forall tr_1..tr_N \in Tr: M(p) = \{tr_1..tr_N\} \exists \\ \{\{k_i; tr_i\}\}, i=1..N: k_i \in Z = t(m(tr, p)). \end{aligned}$$

Тогда для любой позиции  $p$  и ее разрешенных типов трафика определен набор пар  $\{k_i; tr_i\}$ , где  $k_i$  – момент времени поступления метки типа  $tr_i$ . Тип следующей метки для обработки выбирается выбором  $tr_i$  при минимальном  $k_i$ .

4. Переход является разрешенным, если во всех входных позициях находятся метки, и переход не находится в состоянии «занят». Это можно выразить следующим образом:

$$R(t)=1 \Leftrightarrow (S(t)=1) \& (\prod M(p_i)>0),$$

где  $p_1..p_k \in P$  – входные позиции для перехода  $t$ ,  $R(t)$  – функция «разрешенности» перехода  $t$ , а  $S(t)$  – функция «незанятости» перехода  $t$ .

### Представление основных устройств сети

Функционально каждое сетевое устройство может выступать в нескольких «ролях»: передатчика, приемника или генератора трафика. Причем, одно и то же устройство может сочетать в себе несколько ролей, при этом набор ролей носит статический характер, то есть определяется в момент моделирования устройства и не меняется в течение работы модели. Сеть Петри, выполняющую одну из перечисленных функций, будем называть ролевым функционалом. Это понятие вводится для удобства моделирования сетевых устройств, каждое из которых может выступать в одной или нескольких из перечисленных ролей. При таком подходе модель любого сетевого устройства строится из таких ролевых функционалов с помощью операций, приведенных ниже.

Формально определение ролевого функционала для нераскрашенной СП можно представить следующим образом:

$$F: \{m(p^{in}_1),..m(p^{in}_n)\} \rightarrow \{m'(p^{in}_1),..m'(p^{in}_n); m''(p^{out}_1),..m''(p^{out}_s)\},$$

где  $p^{in}_1..p^{in}_n$  – входные позиции  $F$ ;  $p^{out}_1..p^{out}_s$  – выходные позиции  $F$ ;  $m(p)$  – количество меток, входящих в позицию  $p$  извне;  $m'(p)$  – количество меток, которые остаются в позиции  $p$  после одного срабатывания всех разрешенных переходов в  $F$ ;  $m''(p)$  – количество меток, выходящих из позиции  $p$  за пределы  $F$  после одного срабатывания всех разрешенных переходов в  $F$ .

Определим основные ролевые функционалы и представим их в виде простых СП.

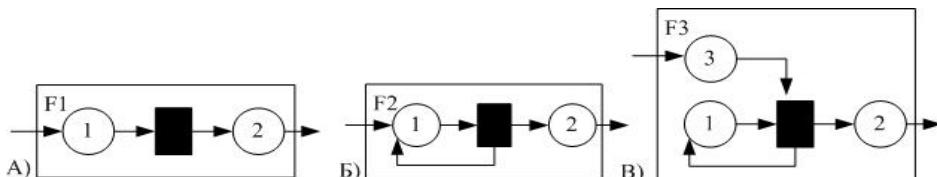


Рис. 1. Ролевые функционалы: а – приемник/передатчик трафика, б – независимый генератор трафика, в – зависимый генератор трафика

**Функционал – приемник/передатчик трафика** (рис. 1а): элементарная сеть Петри, на основе которой строятся все остальные функционалы. Состоит из двух позиций 1 и 2, перехода  $t$  и двух дуг (от позиции 1 к переходу  $t$ , и от перехода  $t$  к позиции 2). Трафик будет поступать в позицию 1 и передаваться в позицию 2, т. е. при наличии метки в позиции 1, метка попадает через переход в позицию 2. С переходом связана задержка обслуживания.

$F(n, 0) = (0, n)$ , т. е. при входе в позицию 1  $n$  меток и после срабатывания  $t$ , во входной позиции останется 0 сообщений, а в выходной позиции 2 появится  $n$  меток.

**Функционал – генератор трафика** независимый вариант (рис. 1б): момент поступления новых

меток в позицию 2 зависит только от временной задержки перехода.

$F(n, 0) = (n, n)$ ; т. е. при входе в позицию  $p_1$   $n$  меток и после срабатывания  $t_1$ , во входной позиции опять будет  $n$  сообщений, а в выходной позиции появятся  $n$  меток.

**Функционал – генератор трафика** зависимый вариант (рис. 1в): момент поступления новых меток в позицию 2 зависит от временной задержки перехода и от наличия в позиции 3 метки.

$F(n, 0, m) = (n, 0, m+n)$ , однако если для позиций 3 и 2 является разрешенным трафик только типа А, для позиции 1 соответственно – только трафик типа Б, то, с учетом этих наложенных ограничений,  $F(n, 0, m) = (n, 0, m)$ .

### Операции с сетями Петри

Для целей моделирования ВС определим (по аналогии с алгеброй сетей Петри, порождающей класс регулярных сетей [4, 6]), следующие операции над элементами сетей  $A_i$ : позициями  $p_j$ , переходами  $t_k$ , дугами  $d_l$  ( $i, j, k, l \in Z$ ): 1. Отождествление позиций «~». 2. Добавление дуги « $\rightarrow$ ». 3. Исключение дуги « $!-\rightarrow$ ». 4. Переименование позиций «/». 5. Объединение СП «&». 6. Расщепление СП «%».

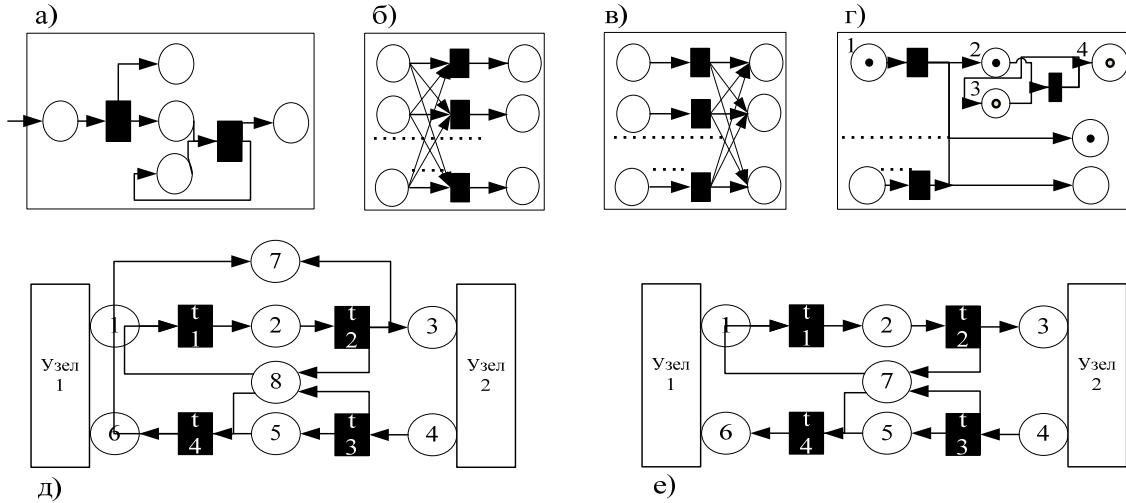


Рис. 2. Модели сетевых устройств: а – рабочая станция; б – концентратор; в – коммутатор-мост; г – маршрутизатор, д – полудуплексный канал связи, е – duplexный канал связи

Аналогичным образом строятся модели других сетевых устройств (рис. 2).

Формально построение устройства с использованием ролевых функционалов и операций между ними выглядит следующим образом, на примере полудуплексного канала передачи данных в ВС (рис. 2д):

Прямой алгоритм построения такой сети следующий:

1.  $V_1 = B(t/t_1^0)$ ;
2.  $V_2 = A(p_1/p_2, p_2/p_3, t/t_2)$ ;
3.  $V_3 = B(p_1/p_8, t/t_1^0)$
4.  $V_4 = B(p_1/p_2, p_2/p_8, t/t_2)$
5.  $V_5 = B(p_1/p_2, p_2/p_7, t/t_2)$
6.  $V' = (p_2(V_1) \sim p_2(V_2))$
7.  $V'' = (t_1^0(V') \sim t_1^0(V_3); p_2(V') \sim p_2(V_3))$
8.  $V''' = (t_2(V'') \sim t_2(V_4); p_2(V'') \sim p_2(V_4); p_8(V'') \sim p_8(V_4))$
9.  $V = (p_2(V''') \sim p_2(V_5); t_2(V''') \sim t_2(V_5))$
10.  $W_1 = B(p_1/p_4, p_2/p_5, t/t_4^0)$ ;
11.  $W_2 = A(p_1/p_5, p_2/p_6, t/t_2)$ ;
12.  $W_3 = B(p_1/p_8, t/t_4^0)$
13.  $W_4 = B(p_1/p_5, p_2/p_8, t/t_3)$
14.  $W_5 = B(p_1/p_5, p_2/p_7, t/t_3)$
15.  $W' = p_5(W_1) \sim p_5(W_2)$
16.  $W'' = (t_4^0(W') \sim t_4^0(W_3); p_5(W') \sim p_5(W_3))$
17.  $W''' = (t_3(W'') \sim t_3(W_4); (p_5(W'') \sim p_5(W_4); p_8(W'') \sim p_8(W_4))$

$$\begin{aligned} 18. W &= (p_5(W''') \sim p_5(W_5); \\ &t_3(W''') \sim t_3(W_5)) \\ 19. VW &= (p_8(V) \sim p_8(W); p_7(V) \sim p_7(W) \end{aligned}$$

Аналогичным образом можно показать конструирование и других типов сетевых устройств.

Приведенные алгоритмы наглядно демонстрируют простоту реализации и имеют линейную сложность для всех представленных типов устройств.

### Апробация модели

Приведем результаты практического использования описанного в статье имитационного подхода, который реализуется в программном комплексе СиМАКС (Система моделирования и анализа компьютерных сетей) в Кемеровском государственном университете.

На основе собранных данных сетевой статистики, аналитически (метод «анализа средних» [1]) рассчитывалась средняя очередь сообщений в каждом узле  $L_i(N)$ , и среднее время собственно обработки сообщений (без учета ожидания в очереди)  $T_i(N)$ . В примере оцениваются параметры рабочих станций и коммуникационного устройства типа мост, для однородного трафика. **Результаты имитационного расчета**

Для получения результатов модель запускалась на выполнение для разных вариантов продолжи-

тельности выполнения:  $T=100$ ,  $T=1000$  и  $T=5000$  тактов. Для каждого значения модель выполнялась по три раза, после чего рассчитывались средние значения (табл. 4).

Данные таблицы 5 показывают относительную погрешность и ее максимальные значения во всех вариантах работы модели ( $T=100$ ,  $T=1000$ ,  $T=5000$ ). С увеличением количества тактов погрешность уменьшается, и результаты имитационного моделирования сходятся к результатам аналитического моделирования.

**Выходы.** В работе описан подход к имитационному моделированию архитектуры и процессов пе-

редачи данных в ВС, на основе модифицированных сетей Петри. Предложен механизм построения сетей Петри для основных объектов ВС, формализованы функции сетевых объектов в виде ролевых функционалов. Для демонстрации построенной модели приводится пример построения СП для локальной сети, проведен анализ полученных результатов.

Рассмотренный подход реализуется в программном комплексе СиМАКС, который используется для анализа топологии и связностей корпоративной сети.

Таблица 2

### Исходные данные замеров

Сетевой узел	1	2	3	4	5	6
Средняя длительность обслуживания с ожиданием	4,0981	1,3162	2,7455	7,7523	7,4632	0,9392

Маршрутная матрица  $P_{ij}$  потоков задается следующим образом:

$$\begin{pmatrix} P_{11} & P_{12} & \dots & P_{16} \\ P_{21} & P_{22} & \dots & P_{26} \\ P_{31} & P_{32} & \dots & P_{36} \\ \dots & \dots & \dots & \dots \\ P_{61} & P_{62} & \dots & P_{66} \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0.02 & 0.32 & 0 & 0.03 & 0.49 & 0.14 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

В таблице 3 приведены результаты аналитических расчетов методом анализа средних значений [1].

Таблица 3

### Результаты расчетов

Номер сетевого узла	1	2	3	4	5	6
Средний размер очереди сообщений в узле	0,0134	1,8201	7,2642	0,0106	0,2167	0,6748
Среднее время обработки сообщения в узле	0,2473	2,0936	2,6738	0,1304	0,162	1,7743

Таблица 4

### Результаты имитационного расчета

Номер сетевого узла		1	2	3	4	5	6
Средний размер очереди сообщений в узле	T=100	0,0224	2,171	6,0184	0,0723	0,3601	0,8984
	T=1000	0,012	2,0284	6,849	0,0682	0,2001	0,723
	T=5000	0,0129	1,984	7,191	0,0382	0,221	0,665
Среднее время обработки сообщения в узле	T=100	0,3657	1,8791	2,6733	0,0969	0,1004	2,1321
	T=1000	0,2041	1,9789	2,5309	0,1144	0,1702	1,8802
	T=5000	0,2157	2,089	2,69	0,1312	0,167	1,8247

Таблица 5

### Расчет погрешности

Номер узла	Очередь						Время обработки					
	Относительная погрешность (E)						Относительная погрешность (E)					
	T=100 (%)		T=1000 (%)		T=5000 (%)		T=100 (%)		T=1000 (%)		T=5000 (%)	
1	0,671641791	67	- 0,1044776	10	-0,037313	4	0,4787707	48	-0,1746866	17	-0,12778	13
2	0,192791605	19	0,1144443	11	0,09005	9	- 0,1024551	10	-0,054786	5	0,0021972	0

3	-0,17149858	17	-0,057157	6	-0,010077	1	-0,000187	0	-0,0534445	5	0,0060588	1
4	0,716981132	72	0,1509434	15	0,0754717	8	- 0,2569018	26	-0,1226994	12	0,006135	1
5	0,661744347	66	- 0,0766036	8	0,0198431	2	- 0,3802469	38	0,0506173	5	0,0308642	3
6	0,331357439	33	0,0714286	7	-0,014523	1	0,201657	20	0,0596855	6	0,0284056	3
Max. E(%)		72		15		9		48		17		13

### Литература

1. Вишневский, В. М. Теоретические основы проектирования компьютерных сетей / В. М. Вишневский. – М.: Техносфера, 2003. – С. 114.
2. Дубинин, В. Н. Организация и проектирование интеллектуальных распределенных вычислительных систем с групповыми взаимодействиями / В. Н. Дубинин // Вычислительная техника в автоматизированных системах контроля и управления: межвуз. сб. науч. тр. – Пенза: ПГУ, 1999. – Вып. 26. – С. 31 – 38.
3. Ильницкий, С. В. Работа сетевого сервера при самоподобной нагрузке // Электронный ресурс <http://314159.ru/ilnickis/ilnickis1.pdf> (дата обращения 02.05.2010).
4. Котов, В. Е. Сети Петри / В. Е. Котов. – М.: Наука, 1984.
5. Миков, А. И. Система имитации с удаленным доступом / А. И. Миков, Е. Б. Замятина // Материалы третьей междисциплинарной конференции с международным участием (НБИТТ-21), 21 – 23 июня 2004 г. – Петрозаводск. – С. 73.
6. Питерсон, Дж. Теория сетей Петри и моделирование систем / Дж. Питерсон. – М.: Мир, 1984.
7. Среда описания ресурса (RDF): Понятия и абстрактный синтаксис // Электронный ресурс. URL: [http://www.w3.org/2007/03/rdf\\_concepts\\_ru/Overview.html](http://www.w3.org/2007/03/rdf_concepts_ru/Overview.html) (дата обращения 02.04.2009).
8. Тарасов, В. Н. Вероятностное компьютерное моделирование сложных систем для анализа их производительности / В. Н. Тарасов: дис. ... д-ра техн. наук. – Оренбург, 2002.
9. Олифер, Н. А. Средства анализа и оптимизации локальных сетей / Н. А. Олифер, В. Г. Олифер // URL: <http://www.citforum.ru/nets/optimize/index.shtml> (дата обращения 28.04.09).
10. Ярославцев, А. Ф. Методы и программные средства гибридного моделирования мультисервисных сетей большой размерности / А. Ф. Ярославцев: дис. ... д-ра техн. наук. – Новосибирск, 2006.
11. HP OpenView Network Node Manager // URL: [http://www.openview.ru/nnm\\_2.htm](http://www.openview.ru/nnm_2.htm) (дата обращения 24.02.2009).
12. Leland, W. E., Murad S. Taqqu, W. Willinger, R. Sherman // Proof of a Fundamental Result in Self-Similar Traffic Modeling / ACM SIGCOMM'95. – San-Fransisco, 1995.