

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ

Российская академия наук

Московский физико-технический институт
(государственный университет)

Российский фонд фундаментальных исследований

Федеральная целевая программа

«Научные и научно-педагогические кадры инновационной России»
на 2009–2013 годы

Фонд содействия развитию малых форм предприятий
в научно-технической сфере

ТРУДЫ 52-й НАУЧНОЙ КОНФЕРЕНЦИИ МФТИ

Современные проблемы фундаментальных и прикладных наук

Часть I Радиотехника и кибернетика

Том 1



Москва–Долгопрудный, 2009

УДК 004:51:621.3:537.8

ББК 32.97

T78

T78 Труды 52-й научной конференции МФТИ «Современные проблемы фундаментальных и прикладных наук». Часть I. Радиотехника и кибернетика. Том 1. — М.: МФТИ, 2009. — 182 с.
ISBN 978-5-7417-0323-6

В сборник включены результаты фундаментальных и прикладных исследований студентов, аспирантов, преподавателей и научных сотрудников МФТИ, а также ряда научных и учебных организаций. Они представляют интерес для специалистов, работающих в области вычислительных и инфокоммуникационных технологий, обработки и защиты информации, систем спутниковой связи.

В октябре 2009 года Московский физико-технический институт (государственный университет) стал победителем конкурсного отбора программ развития университетов России, в отношении которых устанавливается категория «национальный исследовательский университет».

УДК 004:51:621.3:537.8

ББК 32.97

ISBN 978-5-7417-0323-6

© ГОУ ВПО «Московский физико-технический институт (государственный университет)», 2009

Программный комитет

Кудрявцев Н.Н., чл.-корр. РАН, ректор института — **председатель**
Кондранин Т.В., профессор, первый проректор — **зам. председателя**
Стрыгин Л.В., доцент — **учёный секретарь конференции**

Алфимов М.В., академик, директор Центра фотохимии РАН
Андреев А.Ф., академик РАН, директор ИФП РАН
Беляев С.Т., академик РАН, зав. кафедрой МФТИ
Велихов Е.П., академик РАН, президент РИЦ «Курчатовский институт»
Гуляев Ю.В., академик РАН, директор ИРЭ РАН
Дмитриев В.Г., чл.-корр. РАН, зав. кафедрой МФТИ
Иванников В.П., академик РАН, директор ИСП РАН
Коротеев А.С., академик РАН, директор Центра им. М.В. Келдыша
Кузнецов Н.А., академик РАН, зав. кафедрой МФТИ
Макаров В.Л., академик-секретарь Отделения ОН РАН, дир. ЦЭМИ РАН
Петров А.А., академик РАН, заведующий отделом ВЦ РАН
Фортвов В.Е., академик-секретарь Отделения ЭММПУ РАН
Патон Б.Е., академик, президент НАН Украины
Шпак А.П., академик, первый вице-президент НАН Украины
Черепин В.Т., чл.-корр. НАН Украины, директор ФТЦ НАНУ
Жданок С.А., академик-секретарь Отделения ФТН НАН Беларуси

Гаричев С.Н., д.т.н., декан ФРТК
Трунин М.Р., д.ф.-м.н., декан ФОПФ
Негодяев С.С., к.т.н., декан ФАКИ
Грознов И.Н., доцент, декан ФМБФ
Тодуа П.А., профессор, декан ФФКЭ
Вышинский В.В., профессор, декан ФАЛТ
Шананин А.А., профессор, декан ФУПМ
Леонов А.Г., профессор, декан ФПФЭ
Кривцов В.Е., доцент, декан ФИВТ
Ковальчук М.В., чл.-корр. РАН, декан ФНБИК
Деревнина А.Ю., д.т.н., декан ФИВС
Кобзев А.И., профессор, декан ФГН
Кваченко А.В., к.т.н., зав. кафедрой
Алехин А.П., профессор, зав. кафедрой
Белоусов Ю.М., профессор, зав. кафедрой
Бугаев А.С., академик РАН, зав. кафедрой
Габидуллин Э.М., профессор, зав. кафедрой
Гладун А.Д., профессор, зав. кафедрой
Иванов А.П., профессор, зав. кафедрой
Лужкин Д.С., профессор, зав. кафедрой
Петров И.Б., профессор, зав. кафедрой
Половинкин Е.С., профессор, зав. кафедрой
Сон Э.Е., чл.-корр. РАН, зав. кафедрой
Тельнова А.А., доцент, зав. кафедрой
Трухан Э.М., профессор, зав. кафедрой
Холодов А.С., чл.-корр. РАН, зав. кафедрой
Энтово Р.М., академик, зав. кафедрой

Секция радиотехники и защиты информации

УДК 519.688

С.М. Владимиров

vlsergey@gmail.com

Московский физико-технический институт
 (государственный университет)

Использование итеративного декодирования в сетевом кодировании

Передача сообщений в сетевом кодировании при использовании обычных блоковых кодов, рассчитанных на передачу кодовых слов (а не подпространств), с одной стороны, даёт возможность использовать весь опыт и наработки по блоковым кодам. Однако как было показано в работе Jinguo Kang и др. [1], линейные комбинации сообщений и последующее восстановление негативно сказываются на вероятности восстановления сообщений: если на вход декодера поступают сообщения A и $A \oplus B$, то при одинаковой вероятности ошибки в блоке для поступивших сообщений после декодирования вероятность ошибки для сообщения B будет большей. В качестве решения представленной задачи авторы предложили, во-первых, осуществлять одновременное декодирование сообщений итеративным способом, обмениваясь при этом информацией об апостериорных вероятностях декодирования битов, во-вторых, разбить сообщения на два блока, каждый из которых пойдёт по своему пути. Тем самым авторы добились уравнивания вероятностей декодирования сообщений A и B за счёт модификации декодера и модификации протокола.

В работе предлагается использование способа передачи информации, когда вместо одновременного декодирования двух и более сообщений, каждое из которых состоит из нескольких пакетов, используется единственное сообщение из нескольких частей. При этом предлагается способ передачи данных, который легко обобщается как на

произвольное количество частей сообщения (путей в сети), так и на случайное сетевое кодирование, когда о прохождении пакета в сети неизвестно.

Рассмотрим модель сети бабочка. Под \vec{a} и \vec{b} будем понимать две части одного кодового слова $\vec{m}LDPC$ -кода. По стандартному способу декодирования исходного сообщения в приёмниках восстанавливается каждая из его частей, после чего декодируется код и исправляются ошибки передачи. Обозначим принятые сообщения через \vec{m}_1 и \vec{m}_2 . Тогда без ограничения общности для левого приёмника:

$$\begin{aligned}\vec{a}' &= \vec{m}_1, \\ \vec{b}' &= \vec{m}_1 \oplus \vec{m}_2.\end{aligned}$$

После этого сообщение \vec{m}' восстанавливается простой конкатенацией $\vec{m}' = \vec{a}' || \vec{b}'$.

Предлагается новый способ декодирования, когда операция восстановления исходных частей сообщения включается в алгоритм декодирования \vec{m}' из принятых сообщений \vec{m}_1 и \vec{m}_2 с помощью итеративного декодирования. Пусть H — проверочная матрица выбранного LDPC-кода.

Построим сообщение \vec{x} как конкатенацию нулей и принятых сообщений:

$$\vec{x} = \vec{0} || \vec{m}_1 || \vec{m}_2$$

Первая часть вектора (нули) по размеру совпадает с размером исходного сообщения \vec{m} . Таким образом, размерность вектора \vec{x} будет равна удвоенной размерности исходного вектора \vec{m} . Будем использовать следующую проверочную матрицу:

$$H' = \begin{pmatrix} H & 0 & 0 \\ I_l & 0 & I_l & 0 \\ I_l & I_l & 0 & I_l \end{pmatrix}.$$

Здесь l — длина принятых сообщений \vec{m}_1 и \vec{m}_2 , I_l — единичная матрица размера $l = n/2$, где n — длина исходного кодового слова \vec{m} , а также ширина исходной матрицы H ; 0 — нулевые матрицы. Данную проверочную матрицу используем для итеративного «декодирования» сообщения \vec{x} , а именно для восстановления первых $2l$ бит. Итеративный алгоритм используем в том виде, как он описан в книге Маккея [2], причём будем считать, что первые $2l$ бит сообщения \vec{x} имеют априорные вероятности 50% (то есть стёрты). После окончания декодирования первые $2l$ бит сообщения будем использовать как восстановленное

сообщение. Можно показать, что при отсутствии ошибок передачи расширенная матрица P и вектор \vec{x} дают нулевой синдром ошибки.

Для моделирования использовалась матрица регулярного (6,3) LDPC-кода размером 96 на 48. Считалось, что работают два приёмника, первых из которых принимает сообщения a и b , второй — a и $a \oplus b$. При этом каждое сообщение закодировано с использованием амплитудной модуляции с $a = 1$, на канал передачи действует аддитивный белый гауссов шум с дисперсией σ . Оба приёмника пытаются исправить ошибки в кодовом слове \vec{m}' как первым, так и вторым способом. Измеряется вероятность ошибки в блоке в зависимости от величины энергии на информационный бит:

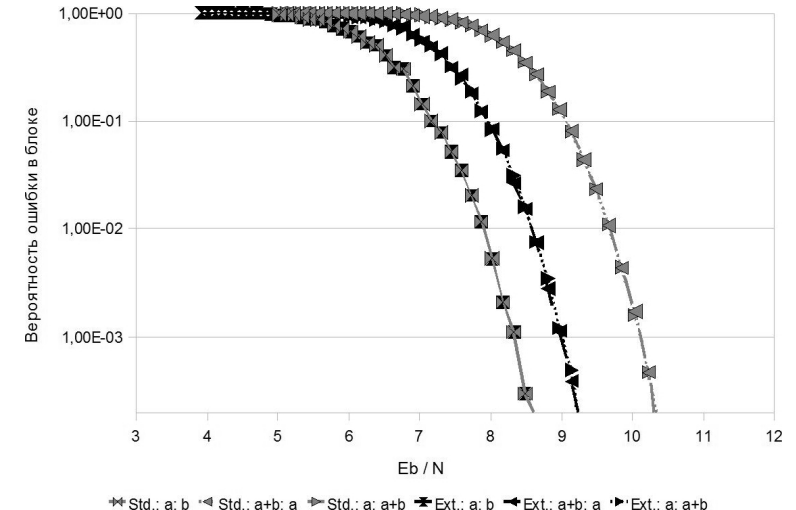


Рис. 1. Вероятность ошибки в блоке в зависимости от величины энергии на информационный бит для стандартного и улучшенного алгоритма

На диаграмме показано 6 графиков, 3 из которых совпадают. Красным отмечены вероятность ошибки в блоке для обычного алгоритма декодирования, жёлтым — для улучшенного алгоритма с расширенной проверочной матрицей. В качестве легенды указываются сообщения, которые пришли на вход декодера. Графики для $a, a \oplus b$ и $a \oplus b, b$ совпадают для обоих алгоритмов, что косвенно свидетельствует о точности моделирования. Совпадение графиков для a, b показывает, что для случая, когда на вход декодера пришли сами сообщения,

а не их линейные комбинации, использование нового алгоритма не даёт преимуществ, однако он и не хуже обычного алгоритма.

Данный способ удобно расширить на случай случайного сетевого кодирования, а также случай, когда количество блоков больше двух (рис. 1).

Литература

1. *King Jyngh, Zhou Bo, Ding Zhi, Lin Shu*. LDPC coding schemes for error control in a multicast network // In Proceedings of IEEE International Symposium on Information Theory, July 6-11. — Toronto: 2008. — P. 822–826.

2. *MacKayd David J.C.* Information theory, inference and learning algorithms. — Cambridge: Cambridge University Press, 2003. — 628 p. — ISBN 0-521-64298-1.

УДК 004.9

С.В. Лапшин

sv.lapshin@gmail.com

Московский физико-технический институт
(государственный университет)

Особое конструкторское бюро систем автоматизированного
проектирования

Особенности реализации средств защиты информации от несанкционированного доступа для инфраструктуры виртуализации

Для многих организаций сейчас очень важно сократить издержки, повысить эффективность и управляемость ИТ-инфраструктуры. Виртуализация платформ является хорошим для этого решением и за последние годы получила широкое распространение. Продуктами виртуализации стало намного проще пользоваться, они стали более надежными и функциональными. Примерами являются VMware Infrastructure, Citrix XenServer, KVM и т. д.

Любую ИТ-инфраструктуру необходимо защищать, в первую очередь от несанкционированного доступа. Традиционные средства защиты информации от НСД в системах виртуализации не эффективны, поскольку в структуре программной среды появляются два дополнительных уровня — гипервизор и виртуальные устройства. Такие изменения кардинально меняют подход к построению системы защиты. Например, если злоумышленник получит доступ к среде виртуализации, то любая виртуальная машина, а следовательно, и вся обрабатываемая ею информация, может быть полностью скомпрометирована. Традиционные средства защиты, даже если они начинают работать сразу после включения виртуальной машины, не смогут защитить обрабатываемую в гостевой ОС информацию, поскольку к ней есть доступ с более низкого уровня. Поэтому система защиты в среде виртуализации должна функционировать на всех уровнях защищаемой среды.

С точки зрения гипервизора и управляющей ОС виртуальная машина, на которой может быть запущена гостевая ОС, представляет собой набор файлов. Пользователь виртуальной инфраструктуры может иметь право на чтение и модификацию этих файлов. Осуществить эти права можно из ОС управления виртуальной инфраструктурой. Поэтому, для обеспечения защиты от НСД разграничения доступа только в гостевых ОС не достаточно. Подсистема разграничения доступа должна также функционировать в ОС управления.

Кроме того, необходимо обеспечивать целостность среды и самих средств защиты. Некоторые решения позволяют контролировать целостность, функционируя исключительно внутри контролируемой среды. Проблема в том, что компонент безопасности таких решений никак не защищен. Виртуальную машину в выключенном состоянии можно модифицировать, следовательно, можно изменить и компонент безопасности. А ведь именно он хранит в себе эталонные параметры среды и сравнивает их с полученными в результате исследования. Следовательно, его модификация приведёт к дискредитации системы контроля целостности, поэтому компонент безопасности должен находиться вне контролируемой программной среды.

В то же время он должен иметь доступ к контролируемой среде. Поскольку вычислительная среда меняется во времени, контроль целостности программной среды «снаружи» затруднен, и компоненту безопасности необходимо иметь возможность «влезть» в эту среду, исследовать ее, и сравнивать полученные параметры среды с эталонными. Под параметрами среды в данном случае подразумеваются не

только параметры гостевой операционной системы, но также настройки самой виртуальной машины, например, набор её оборудования.

Исходя из этих рассуждений, в системах виртуализации компонент безопасности должен находиться на уровне между гипервизором и виртуальными машинами. Также необходимо обеспечивать целостность самого компонента безопасности. Это может быть реализовано только двумя способами — из гипервизора и аппаратно. Поскольку вмешиваться в работу гипервизора некорректно и это может привести к существенному падению производительности системы, то контроль целостности компонента безопасности необходимо осуществлять аппаратно.

Аппаратная реализация технологических операций, процедур и протоколов изначально обладает более высоким потенциалом защиты, чем программная. В виртуальной инфраструктуре необходимо также обеспечивать целостность самого гипервизора, что возможно только с аппаратного уровня. Поэтому, проверкой целостности самого компонента безопасности и гипервизора должен заниматься аппаратный модуль доверенной загрузки.

Таким образом, защищенная система контроля целостности виртуальной программной среды должна состоять из компонента безопасности и аппаратного модуля доверенной загрузки. Компонент безопасности должен иметь доступ к защищаемой программной среде, но находиться на уровне между гипервизором и гостевой ОС. Защиту самого компонента безопасности и гипервизора должен обеспечивать аппаратный модуль доверенной загрузки.

Литература

1. *Конявский В.А.* Управление защитой информации на базе СЗИ НСД «Аккорд». — М.: Радио и связь, 1999. — 325 с.
2. *Yanick de Jong.* Xen hypervisor security in VM isolation. — Amsterdam: Universiteit van Amsterdam, 2009. — 30 с.
3. *Лапшин С.В., Счастливый Д.Ю.* Виртуализация: в поисках точки опоры // Материалы XIV международной конференции «Комплексная защита информации». — 2009. — С. 147.

УДК 004.056.57

В.А. Обернихин

obernikhin@mail.ru

Московский физико-технический институт
(государственный университет)

Восстановление работоспособности сетевых кластеров виртуальных серверов после вирусных атак

Центр обработки данных. В настоящее время появляется все большее количество центров обработки данных, в том числе и в России. Это связано с возрастающей активностью использования сети Интернет как компаниями, так и конечными пользователями. Крупные компании часто создают собственные центры обработки данных. Небольшим компаниям, как правило, выгоднее арендовать сервер в центре обработки данных сервис-провайдера. Крупные центры обработки данных могут состоять из более чем 20 тыс. физических серверов.

Виртуализация. Вычислительные мощности физических серверов часто используются не полностью. Также можно встретить ситуацию, когда, например, время CPU используется лишь на несколько процентов. Этот факт давно был замечен различными исследователями, в частности, уже в течение нескольких лет программное обеспечение проекта Берклиевского университета Seti@home [1] начинает использовать время CPU только когда оно используется не полностью другими пользовательскими программами.

Специализированное программное обеспечение даёт возможность запускать множество (десятки и даже сотни) виртуальных серверов на базе одного физического. С точки зрения пользователя каждый виртуальный сервер представляет из себя полнофункциональный физический сервер.

На одном физическом сервере могут быть запущены десятки и даже сотни виртуальных серверов.

Безопасность виртуальных серверов. С точки зрения информационной безопасности возникают дополнительные вопросы обеспечения безопасности между виртуальными серверами, а также без-

опасность самого физического сервера. Изоляция виртуальных серверов друг от друга и от основного физического сервера обеспечивается самим ПО виртуализации, однако механизмы интеграции гостевых систем с хостом а также уязвимости ПО делают возможным нарушение данной изоляции ([2]).

При получении несанкционированного доступа к основному физическому серверу злоумышленник получает неограниченные полномочия для доступа ко всем виртуальным серверам на данной машине, поэтому обеспечение безопасности основного физического сервера является приоритетным направлением исследований информационной безопасности.

Для любой компьютерной системы необходимым механизмом построения изолированной программной среды [3, с. 30] является обеспечение доверенной загрузки операционной системы [3, с.57]. В свою очередь для доверенной загрузки операционной системы необходимы средства контроля целостности загружаемых исполняемых файлов.

Наиболее распространенным аппаратным средством контроля целостности загружаемых модулей является Trusted Platform Module [4], интегрированный во множество выпускаемых на сегодняшний день материнских плат компьютеров и переносных устройств. Дополненный Trusted Execution Technology [5] данный аппаратный комплекс позволяет обеспечить доверенную загрузку основного «физического» сервера и виртуальных серверов [6].

Стоит отметить, что на сегодняшний день известны атаки, позволяющие сразу же после доверенной загрузки получить полный доступ к системе [7].

Восстановление сетевых кластеров виртуальных серверов после вирусных атак. Применение совокупностей технологий аппаратной поддержки виртуализации (VT-x), TPM и TXT позволяет не только контролировать целостность исполняемых модулей системы в момент загрузки, но и использовать тот же механизм на уже зараженной системе. В качестве обычного приложения, контролируемого TPM и TXT, может быть запущено приложение — «аудитор», для обследования текущего состояния системы. Поскольку TPM и TXT обеспечивают дополнительную изоляцию страниц памяти для приложений, приложение — «аудитор» может осуществлять деятельность по сбору информации и проверке текущего состояния системы даже в уже зараженной среде. При несанкционированном доступе к страницам памяти приложения-аудитора будут срабатывать обработ-

чки, позволяющие обеспечивать адекватную реакцию на попытки заражения приложения-аудитора.

Известные способы использования совокупности аппаратных модулей TPM, TXT и VT-x ограничивались обеспечением доверенной загрузки основного физического сервера и виртуальных серверов.

Предлагаемый способ запуска приложения-аудитора на уже зараженной системе позволяет расширить область применения данных аппаратных модулей для решения практически важной задачи восстановления работоспособности сетевых кластеров виртуальных серверов после вирусных атак.

Литература

1. Исследовательский проект Берклиевского университета Seti@home URL: [http:// setiathome.berkeley.edu/](http://setiathome.berkeley.edu/).
2. Бюллетень по информационной безопасности VMware Security Advisory VMSA-2009-0006 URL: <http://www.vmware.com/security/advisories/VMSA-2009-0006.html>.
3. Щербakov А.Ю. Методы и модели проектирования средств обеспечения безопасности в распределённых компьютерных системах на основе создания изолированной программной среды. Диссертация на соискание ученой степени доктора технических наук. — М.: 1997.
4. Trusted Computing Group URL: <http://www.trustedcomputinggroup.org/developers/>.
5. Trusted Execution Technology URL: <http://www.intel.com/technology/security/>.
6. Bernhard Jansen, HariGovind V.Ramasamy, Matthias Schunter. Flexible Integrity Protection and Verification Architecture for Virtual Machine Monitors» (IBM Research) // 2nd Workshop on Advances in Trusted Computing. — 2006. URL: <http://www.trl.ibm.com/projects/watc/XenSecurityServicesPaper.pdf>.
7. Wojtczuk R., Rutkowska. J. Attacking Intel®Trusted Execution Technology // Black Hat Briefings. — USA, DC: 2009. URL: <http://invisiblethingslab.com/press/itl-press-2009-01.pdf>.

УДК 519.72

А.А. Сенюта, А.С. Кшевецкий

anton.seny@gmail.com

Московский физико-технический институт
(государственный университет)

Оптимизация параметров интегрированной криптосистемы

Введение. Как известно, интегрированная криптосистема обеспечивает защиту от НСД и исправление случайных ошибок в канале связи. Варьируя параметры криптосистемы, мы можем улучшать одну характеристику, но при этом будут ухудшаться другие. Например, повышая криптостойкость, ухудшается помехозащищенность. Таким образом, важной является задача определения параметров криптосистемы для обеспечения оптимальных характеристик.

Целью работы является выбор оптимальных параметров интегрированной криптосистемы на основе ранговых кодов для использования в беспроводном канале связи с широкополосными и узкополосными шумами.

Был определён следующий набор параметров:

- вид криптосистемы;
- длины открытого и шифро-блоков;
- скорость шифрования;
- криптостойкость;
- модуляция, виды перемежения;
- гауссов шум;
- широкополосный и узкополосный шумы.

Проводилось исследование влияния каждого параметра на характеристики криптосистемы.

Для получения характеристики помехозащищенности, было выполнено моделирование криптосистемы. Результатом моделирования является зависимость вероятности ошибки на бит от величины гауссова шума.

Результаты. В выполненной работе выделено 3 набора параметров интегрированной криптосистемы (рис. 1). Из 3-х представленных наборов первый является наиболее оптимальным.

Наборы параметров 1, 2 обеспечивают большую криптостойкость, но меньшую помехозащищенность по сравнению с набором 3. Помехозащищенность рассматривается как требуемый уровень сигнал/шум, необходимый для обеспечения значения ошибки на бит, равной 10^{-2} . Разница в помехозащищенности между 1-м и 3-м наборами составляет 2 дБ. Это является малым значением для канала связи с высоким уровнем сигнал/шум (> 20 дБ). Значение криптостойкости набора 3 является критичным на сегодняшний день.

Медленное расшифрование (для всех 3-х наборов) обусловлено медленной операцией вычисления обратного элемента в поле. Выбрав базис поля специального вида можно существенно, оценочно в 10 и более раз, повысить скорость шифрования и расшифрования.

Для выбора вида криптосистемы было рассмотрено несколько их модификаций:

- криптосистема ГПТ [2];
- добавление столбцевого скремблера [3];
- на приводимых ранговых кодах [4];
- выбор матрицы столбцевого скремблера специального вида над расширенным полем.

В качестве оптимальной была взята криптосистема с матрицей столбцевого скремблера, выбранной над расширенным полем. Такая криптосистема обладает наибольшей криптостойкостью и одновременно возможностью неплохо исправлять ошибки в канале связи.

Выбрана модуляция QAM-16, характерная для радиоканала с ортогональным частотным уплотнением. Передаваемый сигнал представляется в виде матрицы из QAM-символов на частотно-временной диаграмме. Строки соответствуют подчастотам, столбцы — дискретам времени. Тогда широкополосный шум — стирание столбцов, а узкополосный шум — строк. Вероятность появления помехи задается как стирание одной строки или одного столбца. Для модели канала была выбрана вероятность появления помехи, равной 10^{-3} .

Перемежение применяется для исправления пакета ошибок. Причём в зависимости от вероятности появления помех (широкополосной, узкополосной) нужно применять свой вариант перемежения. В работе [1] произведён сравнительный анализ разных видов перемежений для ранговых кодов. Показано, что при равной вероятности помех двух видов, нужно применять перемежение, при котором каждый QAM-символ на частотно-временной диаграмме содержит биты разных кодовых слов рангового кода. Такой вариант перемежения

взяты в качестве оптимальных для интегрированной криптосистемы в условиях рассматриваемых помех.

3. Заключение. В работе предложены оптимальные параметры для интегрированной криптосистемы с защитой от несанкционированного доступа и помех в канале исходя из требований на криптостойкость. Проведено сравнение помехозащищенности для нескольких уровней криптостойкости.

Параметр	Набор параметров		
	1	2	3
Поле	$GF(2^{64})$		$GF(2^{32})$
Длина кодового слова (шифрблока)	64		32
Длина информационного слова (открытого блока)	32		16
Криптостойкость	$O(2^{160})$	$O(2^{120})$	$O(2^{70})$
Скорость шифрования, Кбит/с	800–920		1800–2100
Скорость расшифрования, Кбит/с	20–50		90–200
Отношение сигнал/шум (для вероятности ошибки на бит 10^{-2}), дБ	22,5	22	20,5

Рис. 1. Наборы оптимальных параметров интегрированной криптосистемы

Литература

1. *Kshevetskiy A., Senyuta A.* Performance Evaluation of Rank Codes in Channels with AWGN and Criss-Cross Errors // Proc. of the XII international symposium on problems of redundancy in information and control systems. — Saint-Petersburg, Russia, 26-30 May, 2009. — P. 130–136.
2. *Gabidulin E.M., Paramonov A.V., Tretjakov O.V.* Rank errors and rank erasures correction // Proc. Fourth Int. Colloquium on Coding Theory. — Dilijan, Armenia, 30 September. — 7 October, Yerevan, 1992. — P. 11–19.
3. *Ouriuski Alexei, Gabidulin E.M.* Column Scrambler for the GPT Cryptosystem // Discrete Applied Mathematics. — 2003. — N. 128(1). — P. 207–221.
4. *Kshevetskiy A., Gabidulin E.* High-weight errors in reducible rank codes // Proc. of the 8th International Symposium on Communication Theory & Applications. — 2005. — P. 71–76.

УДК 004.9

С.И. Смирнов

imshodan@gmail.com

Московский физико-технический институт
(государственный университет)

Исследование линейных регистров сдвига специального вида

В [1, 2] введено понятие управляемого регистра сдвига (УРС). В работе исследуются свойства линейных управляемых регистров сдвига (ЛУРС). Проведены анализ ПСП на основе ЛУРС и сравнение этих последовательностей с ПСП на основе ЛРС (линейных регистров сдвига). Свойства последних хорошо изучены. Исследования ПСП на основе ЛРС более сложных видов, в частности, затронутого в данной работе, призваны устранить уязвимости, присущие классическим ЛРС.

Отличие ЛУРС от классического состоит в возможном наличии инверторов между элементами задержки.

Рассмотрим, какие же преимущества даёт исследуемая структура.

Во-первых, при криптоанализе системы, построенной на ЛУРС, существенно увеличивается количество переборных для решения задачи полным перебором — с 2^n до 2^{2n} .

Во-вторых, криптоанализ принципиально затруднён. Действительно, рассмотрим зависимость между элементами выходной последовательности $\{x_n\}$ классического ЛРС. Она имеет вид $x_n = \sum_{i=0}^{n-1} \alpha_i x_i$, где x_0 — первый элемент, с которого началось наблюдение выходной последовательности ЛРС криптоаналитиком. Для вскрытия ЛРС, то есть для нахождения всех α_i , достаточно пронаблюдать последовательность длиной $2n$ и решить систему линейных уравнений. Наличие инверторов усложняет картину. Зависимость будет иметь следующий вид: $x_m = \sum_{i=0}^{n-1} \left(\alpha_i \left[x_{m-n+i} \oplus \sum_{j=0}^i \beta_j \right] \oplus \beta_i \right)$.

Такое выражение показывает, что получаемая в случае ЛРС специального вида система уравнений для криптоанализа нелинейна.

Литература

1. Орлов В.А. Об одном представлении конечных автоматов // Тезисы докладов XV международной конференции «Проблемы теоретической кибернетики». Казань: Отечество, 2008.

2. Орлов В.А. Об одном способе реализации конечных автоматов // Вестник МГУПИ. — 2009. — № 21. — С. 77–80.

УДК 519.725.2

И.Ю. Сысоев

Igor.Sisoev@gmail.com

Московский физико-технический институт
(государственный университет)

Одновременное использование стандартного и нормального базисов при реализации алгоритмов кодирования и декодирования рангового кода

На сегодняшний момент ранговые коды являются перспективными для использования в системах, использующих идею сетевого кодирования. Актуальной задачей является эффективная реализация алгоритма рангового кода.

При реализации криптографических алгоритмов самыми ресурсозатратными являются операции возведения в степень (в том числе, операция инвертирования) и умножения. Чаще всего вычисление в конечных полях выполняется либо только в нормальном, либо только в стандартном базисе. Умножение эффективнее производить в стандартном (линейная сложность), а возведение в степень — в нормальном базисе (циклический сдвиг координат). Существует метод, позволяющий использовать оба этих преимущества. В таком методе операции умножения и возведения в степень используются в наиболее подходящем для операции базисе. Дополнительную вычислительную нагрузку в этом случае даёт переход от нормального базиса

к стандартному и наоборот. Сложность перехода называется транзитивной сложностью. Низкой транзитивной сложностью обладают оптимальные нормальные базисы. При оптимальном выборе базисов сложность перехода будет сравнима с $O(n)$, если используется оптимальный базис первого типа, или с $O(pn \log_p n)$ — в случае базисов второго и третьего типа.

Оценим сложность операции умножения в стандартном базисе:

$$M_g^s(n) \leq M_2(n) + kn,$$

где — $M_2(n) = O(n \log_2 n \log_2 \log_2 n)$ — сложность операции умножения многочленов степени $n - 1$ над полем $GF(2)$; с высокой вероятностью $k = 3$, в худшем случае $k = 5$.

Оценим сложность операции умножения в оптимальном нормальном базисе. Для выполнения умножения выполняется переход к стандартному базису со схемной сложностью $2n - 2$, затем производится умножение в стандартном базисе и осуществляется переход обратно к нормальному базису со сложностью $n - 1$.

Оценка сложности в оптимальном нормальном базисе 1-го типа:

$$M^{O1}(GF(2^n)) \leq M_2^s(n) + 7n - 8.$$

В оптимальном нормальном базисе 2-го или 3-го типа

$$M^{O2}(GF(2^n)) \leq 3M_2^s(n) + \frac{3n}{2} \log_2 n + O(n).$$

Оценим сложность операции инвертирования в стандартном базисе:

$$I(GF(2^n)) \leq (n - 1)K_2(n) + (\lambda_2(n - 1) + \nu_2(n - 1) + 1 + o(1))M^O(GF(2^n)),$$

где $K_2(n)$ — оценка сложности возведения в степень 2, $K_2(n) = O(n)$. Заметим, что указанная оценка сложности инвертирования в случае стандартного базиса справедлива и для операции деления.

Сложность операции инвертирования в оптимальном нормальном базисе:

$$I^O(GF(2^n)) \leq (\lambda_2(n - 1) + \nu_2(n - 1) + 1 + o(1))M^O(GF(2^n)),$$

где $\lambda_2(n) = \log_2(n - 1)$, а $\nu_2(n)$ — число единиц в двоичной записи числа n . Стоит указать, что возведение в степень 2 выполняется практически без затрат.

Сложность операции возведения в произвольную степень $d \leq n$:

$$O\left(\frac{M^O(GF(2^n)) \log_2 d}{\log_2 \log_2 d}\right).$$

Применим предложенные оценки для сравнения быстродействия реализаций алгоритма декодирования ранговых кодов. Стандартный метод — метод проведения вычислений только в стандартном базисе. Оптимальный метод — метод проведения умножения в стандартном базисе, а возведения в степень — в нормальном базисе. Для простейшего случая с кодовым расстоянием $d = 3$, потребуются две операции умножения для вычисления синдрома s , две операции деления и одна операция возведения в квадрат для определения вектора ошибки.

Оценим сложность алгоритма в случае стандартного базиса:

$$\begin{aligned} L_{std} &= 3(M(n) + 3n) + \\ &+ 2((n-1)O(n) + (\lambda_2(n-1) + \\ &+ \nu_2(n-1) + 1 + o(1))M(n)) + O(n) = \\ &= 2((n-1)O(n) + (\lambda_2(n-1) + \nu_2(n-1) + \\ &+ 2 + o(1))M(n)) + M(n) + 9n + O(n), \end{aligned}$$

Если производить операцию возведения в степень только в нормальном базисе, а умножение многочленов — в стандартном, то получаем оценку сложности алгоритма:

$$\begin{aligned} L_{opt} &= 3(M(n) + 7n - 8) + \\ &+ 2(\lambda_2(n-1) + \nu_2(n-1) + 1 + \\ &+ o(1))(M(n) + 7n - 8) + 1 \cdot 0 = \\ &= 2(\lambda_2(n-1) + \nu_2(n-1) + 2 + \\ &+ o(1))(M(n) + 7n - 8) + \\ &+ (M(n) + 7n - 8), \end{aligned}$$

Случай использования только оптимального нормального базиса не рассмотрен, поскольку сложность операции умножения в таком варианте может достигать верхней границы $O(n^3)$.

Для сравнения скорости алгоритмов оценим асимптотическую разность сложностей реализации:

$$L_{std} - L_{opt} \sim O(n^2 - n \log n).$$

Полученный результат позволяет сделать вывод, что для случая ранговых кодов смешанное использование умножения в стандартном базисе и возведения в степень в нормальном базисе эффективней, чем реализация вычислений только в стандартном базисе.

Литература

1. Габидуллин Э.М., Афанасьев В.Б. Кодирование в радиоэлектронике. — М.: Радио и связь, 1986. — 176 с.
2. Болотов А.А., Гашков С.Б. О быстром умножении в нормальных базисах конечных полей // Дискретная математика. — 2001. — Т.13, № 3. — С. 3–31.
3. Сагалович Ю.Л. Введение в алгебраические коды. — М.: МФТИ, 2007. — 260 с.

УДК 004.492.3

А.Д. Чорняк

a.chornyak@gmail.com

Московский инженерно-физический институт (Национальный исследовательский ядерный университет)

Автоматизированное выделение и методы оценки сигнатур разрушающих программных воздействий

В настоящей статье предлагается следующий подход к автоматизации процесса выделения сигнатур.

Для начала необходимо выделить множество файлов, подверженных разрушающему программному воздействию (R). С этой целью производится запуск модели разрушающего программного воздействия (РПВ), после чего осуществляется сравнение контрольных сумм файловой системы до и после заражения. Элементы множества

R можно получить не только исполнением самого РПВ, но и запуском его потомков. Чем больше множество R , тем меньше вероятность ошибок типа «false negative», а это является наиболее важным фактором при сигнатурном анализе.

Далее необходимо создать максимально полное множество всех программ, не подверженных РПВ. Это множество будет представлять собой «чистую систему» (C). Чем оно больше, тем меньше вероятность ошибок типа «false positive».

По сути, задача выявления сигнатур сводится к нахождению последовательности, присутствующей во всех файлах, зараженных данной моделью РПВ, и не встречающихся ни в одном файле «чистой системы». То есть необходимо найти новое множество потенциальных сигнатур (SP) представляющее собой вычитание множества C из множества R . При этом нет никакой необходимости в дизассемблировании файлов, что даёт существенный выигрыш в производительности.

Любую из полученных последовательностей можно использовать в качестве сигнатуры. Однако то, что выбранные последовательности не встречаются в «чистой системе», ещё не гарантирует отсутствия ложных срабатываний. Это связано с тем, что невозможно составить полное множество «чистой системы» C , так как оно будет стремиться к бесконечности. Данная задача разрешима только для узких областей применения, где набор возможного программного обеспечения ограничен. Следовательно, необходимо провести дополнительный этап анализа множества SP , с целью получения последовательности, представляющей собой качественную сигнатуру. Здесь под качественной сигнатурой понимается последовательность, обеспечивающая минимальное количество ложных срабатываний.

Качественная сигнатура должна содержать как можно больше эвристических признаков, характерных для моделей РПВ. Чем качественнее будет сигнатура, тем меньше вероятность ложных срабатываний. Исходя из этого, предлагается создать множество эвристических признаков (E), каждому из которых поставить в соответствие некоторый «вес» (P_i , $i = 1, k$, где k — число эвристических признаков), который будет соответствовать вероятности ложного срабатывания в случае наличия данного признака в сигнатуре. Соответственно чем более характерный признак, тем меньше вероятность.

В качестве характерных для РПВ признаков можно использовать следующие: определение местоположения, то есть определение текущего IP, открытие файла на запись или смещения указателя запи-

си в конец, изменение даты или атрибутов у файла, строки типа « [Ee] [Ll] [Ff] », « [Ss]cript », « [Jj]ava », и т. д.

При обнаружении нескольких эвристических признаков в оцениваемой потенциальной сигнатуре, для определения общей вероятности предлагается применить метод, основанный на байесовском подходе к вычислению вероятностей.

Суть его состоит в следующем. Проверке подлежит некоторая гипотеза H , с которой связаны некоторые утверждения A_1 , A_2 и т. д. Тогда, проверив одно из них (A), мы можем пересчитать вероятность гипотезы H с учётом истинности или ложности этого утверждения.

В рамках данной работы в качестве гипотезы H будет выступать предположение о том, что файл, содержащий данную сигнатуру, будет являться инфицированным, либо самим РПВ. В качестве утверждений (A) предлагается использовать эвристические признаки, присутствующие в оцениваемой сигнатуре.

Таким образом, будут получены конечные вероятности выдвинутой гипотезы для каждой выделенной сигнатуры:

$$P_1(H/A_1A_2\dots A_s), \quad P_2(H/A_1A_2\dots A_s), \quad \dots, \quad P_i(H/A_1A_2\dots A_s),$$

где s — количество эвристических признаков в сигнатуре. Из вычисленных вероятностей берется максимальная, так как она обеспечит наименьшую вероятность ложного срабатывания.

Оценка полученных вероятностей в соответствии с заданным уровнем значимости α осуществляется на основе анализа вхождения параметра P_i в доверительный интервал.

$$\bar{P} - t \frac{\sigma}{\sqrt{n}} < P < \bar{P} + t \frac{\sigma}{\sqrt{n}},$$

где $t \frac{\sigma}{\sqrt{n}} = \delta$ — точность оценки, σ — среднеквадратическое отклонение, n — объём выборки, P — выборочное среднее, t — аргумент функции Лапласа, при котором $\Phi(t) = \frac{\alpha}{2}$.

Среднеквадратическое отклонение рассчитывается как

$$\sigma = \sqrt{\frac{1}{n} \sum_{i=1}^n (P_i - \bar{P})^2}.$$

В данном случае выборка будет включать в себя совокупность всех весов P_i , $i = 1, k$, а также все вариации их произведений.

Вместо множества эвристических признаков в описанном выше методе можно также использовать частотные признаки. Для этого рассчитывается распределение значений байт по частоте их использования в базе данных известных сигнатур (или вирусов), которое принимается как эталонное. «Вес» ставятся в соответствие каждому значению байт, отражая частоту появления последнего в базе сигнатур. Чем больше частота, тем больше «вес». Далее производится поиск значений байт из эталонного распределения во множестве потенциальных сигнатур SP . Если в одной последовательности множества SP содержатся несколько байт, значения которых совпадают со значениями из эталонного распределения, то «веса» суммируются. Следовательно, из множества SP выбирается сигнатура с минимальным значением P .

Если вероятность ложного срабатывания у всех полученных сигнатур достаточно велика, то для исключения ошибки типа «false positive» предлагается совместно использовать пару потенциальных сигнатур. При этом общая вероятность ложного срабатывания будет равна произведению вероятностей каждой сигнатуры. Однако недостатком данного подхода является увеличение времени сканирования вдвое.

Эффективность применения описанного метода обусловлена следующими факторами:

- использование жёстко заданных алгоритмов позволит существенно повысить скорость выявления сигнатур;
- исключение человеческого фактора сведёт к минимуму случайные ошибки;
- автоматизация процесса позволит использовать методы, требующие больших временных затрат;
- использование аппарата теории вероятности позволит дать оценку полученным сигнатурам.

Литература

1. *Cohen F.* Computer viruses // Computers & Security. — 1987. — N. 6. — P. 22–35.
2. *Christodorescu M., Jha S.* Static analysis of executables to detect malicious patterns // SSYM'03: Proc. of the 12th conference on USENIX Security Symposium. — Berkeley, CA, USA: USENIX Association. — 2003. — P. 12.
3. *Касперски К., Рокко Е.* Искусство дизассемблирования. — СПб.: БХВ-Петербург, 2008. — 891 с.

Секция инфокоммуникационных систем и сетей

УДК 519.715

В.А. Абдрашитов

snooker@rt.mipt.ru

Московский физико-технический институт
(государственный университет)

Оценка эффективности систем управления работой персонала для операторов связи

В настоящее время активно разрабатываются и внедряются так называемые системы управления работой персонала (СУРП, англ. Workforce Management System) — программное обеспечение, которое оптимально составляет расписание задач, планирует рабочее время и поддерживает работу сотрудников компании, основная деятельность которых связана с выполнением заданий или поручений в разных местах, например, инженеров крупных телекоммуникационных операторов.

Формирование перечня инженерных работ, их назначение на определённое время определённому исполнителю, передача ему информации о задачах, получение от него отчёта об успешном или неуспешном выполнении задачи, координация транспортных средств осуществляются в центре управления. Расписание составляется с учётом свободных рабочих часов инженеров, их физического расположения, профессиональных навыков, имеющегося в наличии оборудования, временных ограничений и связанности задач. Далее может производиться так называемая оптимизация — модификация расписания для улучшения его параметров, повышения эффективности, например: уменьшение промежутков в расписании, сокращение дальности перемещения инженеров путём назначения им задач в близко расположенных локациях, равномерная загрузка задачами разных инженеров.

Эффективное планирование и управление инженерными работами представляет собой нетривиальную задачу и требует значительного количества времени и персонала. Специфика этих процессов допускает их автоматизацию. Современные СУРП могут взять на себя функции составления и оптимизации расписания по различным параметрам, составления поручений для инженеров и обработки отчёты о задачах, прокладывания маршрутов для водителей и прогнозирования спроса на инженерные ресурсы. Это позволит избежать ошибок, допускаемых диспетчером-человеком, увеличить производительность и снизить операционные расходы, принося этим дополнительную прибыль компании. Поэтому представляет интерес величина прироста эффективности при применении СУРП по сравнению с ручным выполнением перечисленных функций.

В работе представлена математическая модель СУРП, в рамках которой проведена формализация входных и управляемых параметров, выбраны критерии эффективности расписания и в упрощённом виде сформулирована математическая задача составления оптимального расписания задач и распределения работников по задачам. В более общих предположениях выведены выражения, связывающие совокупные выходные параметры расписания со средними показателями по выполнению отдельных задач и характеристиками трудовых ресурсов компании. Также выведены формулы для некоторых ключевых показателей эффективности основной деятельности оператора связи в зависимости от используемой функциональности системы управления работой персонала.

Для получения численных результатов были собраны модельные данные о характеристиках типичного крупного оператора связи и произведён расчёт и сравнение модельных значений средней пропускной способности (общее количество выполняемых задач в сутки) при использовании некоторых функций СУРП. В данной модели функции уплотнения расписания, навигатора, навигатора с учётом трафика и наличие связи с реестром оборудования дают прирост пропускной способности на 2.1, 0.8, 2.3 и 0.7 процентов соответственно.

Литература

1. *Melik R.* The Rise of the Project Workforce. — Hoboken: John Wiley & Sons, 2007. — 344 с.

2. *Guido B., Roberto G., Di Tria P. [et al.]*. Workforce management (WFM) issues // IEEE Network Operations and Management Symposium. — 1998. — V. 2. — P. 473–482.

3. *Azarmi N., Smith R.* Intelligent scheduling and planning systems for telecommunications resource management // BT Technology Journal. — 2007. — V. 25, N. 3. — P. 241–248.

4. *Pinker E.J., Larson R.C.* Models of Flexible Workforces in Stochastic Service Environments, the One-Job Case. — Boston: Massachusetts Institute of Technology, Operations Research Center, 1995. — 53 с.

5. *Карманов В.Г.* Математическое программирование. — М.: Физматлит, 2004. — 264 с.

УДК 004.896

А.В. Васильев¹, М.Н. Антоненко²

zolenenok@gmail.com, a.antonenko@fcs.coe.nagoya-u.ac.jp

¹ Московский физико-технический институт
(государственный университет)

² Институт автоматизации проектирования РАН

Модели и методы оценки количественных характеристик комплекса работ в рамках проектов

Данная работа является исследованием, относящимся к области применения средств и методов интеллектуального анализа данных для оценки количественных характеристик работ в рамках проектов.

Цель работы — показать состоятельность и применимость методов интеллектуального анализа данных (Data Mining) к задачам оценки количественных характеристик работ в рамках проектов.

В работе решаются две задачи: из множества предоставленных данных выделить наиболее важные — те, которые оказывают наибольшее влияние на конечный результат; на основе наиболее значимых данных решить задачу регрессии и показать состоятельность решения.

Для обнаружения этих «скрытых» знаний применяется алгоритм, включающий:

1) обработку пропущенных значений — таблицы данных часто содержат пропущенные значения которые могут быть интерпретированы как редкие значения или их отсутствие;

2) удаление выбросов — резко выделяющихся значений экспериментальных величин;

3) нормализацию — преобразование числовых атрибутов таким образом, чтобы значения атрибутов лежали в одном и том же диапазоне;

4) определение важности атрибутов и выделение наиболее релевантного их набора для оценки целевого атрибута. Это необходимо для увеличения скорости и, возможно, точности для модели классификации, построенной на таблице с большим числом атрибутов;

5) решение задачи регрессии;

6) тестирование регрессионной модели.

В качестве задачи выбран пример с семью значимыми атрибутами и одним незначимым (количество выпитых чашек кофе). В целевой атрибут мы записываем значение некоторой функции, в которой вводим множитель «ноль» перед атрибутом, означающего количество выпитого кофе. Хочется заметить, что итоговая функция содержит не только линейные члены, но также и произведения значений атрибутов, возведение в степень и деление. После того как построена задача, мы определяем те атрибуты, которые более всего оказали влияние на ответ. На рис. 1 представлено в графическом виде решение этой подзадачи — видно, что атрибут наименее значим.

После того как модель построена и выбраны атрибуты, по которым предлагается делать оценку, данные разбиваются на два множества. На одном множестве производится «обучение»; на втором множестве проверяется качество (состоятельность) предсказания на основе имеющихся действительных значений.

Для решения задачи были сняты метрики — среднеквадратичное отклонение, абсолютная погрешность. Если считать, что отклонения имеют нормальное распределение, то получится график, представленный на рис. 2. Видим, что достоверность предсказания составляет порядка 80 процентов, что по мировой практике является очень приемлемым результатом.

Таким образом, по результатам выполнения работы средства Data Mining показали перспективность и состоятельность применения к задачам оценки количественных характеристик работ. Результаты,

полученные на тестовых модельных данных, показали достаточную точность, чтобы считать средства Data mining пригодными для анализа проектов на основе накопленных данных, полученных с выполненных проектов.

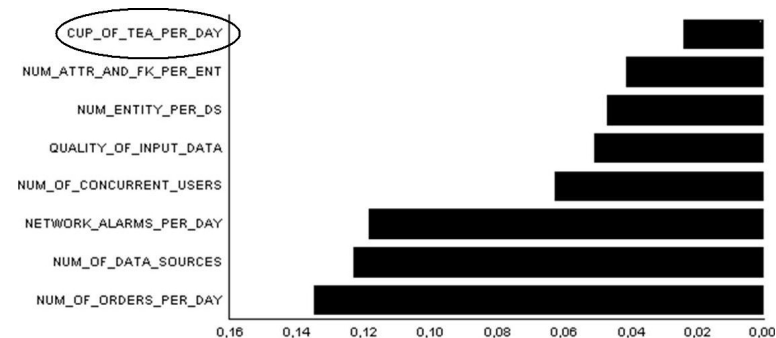


Рис. 1

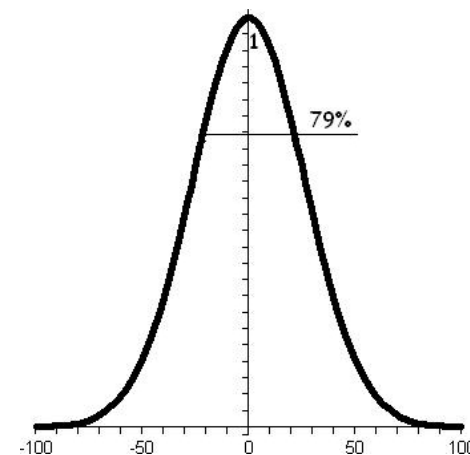


Рис. 2

Литература

1. Барсегян А.А., Курпьянов М.С., Степаненко В.В. [и др.]. Технологии анализа данных: Data Mining, Visual Mining, Text Mining, OLAP. — СПб.: БХВ-Петербург, 2008. — 384 с.

2. Хардман Р., МакЛафлин М. Oracle PL/SQL для профессионалов. — М.: Лори, 2007. — 418 с.

УДК 519.816

А.В. Евдокимов¹, А.Е. Харичкин^{1,2}

alexey.evdokimov@gmail.com, akharitchkin@gmail.com

¹ Московский физико-технический институт

(государственный университет)

² ООО «НетКрэкер»

Итерационные многометодные алгоритмы многокритериального ранжирования

Данная работа посвящена разработке нового класса методов для задач многокритериального принятия решений, в частности, задач линейного упорядочения альтернатив (альтернативных решений). Задачи подобного рода называются ранжированием, а их решение обычно сводится к получению какого-либо рейтинга для каждой из альтернатив (с последующей сортировкой альтернатив по рейтингу).

Специалисты в области задач ранжирования обладают широким арсеналом средств. Среди этих средств различают полностью автоматические методы, основанные на запрограммированных алгоритмах, методики принятия решений человеком (экспертом) и так называемые человеко-машинные процедуры, в которых результат компьютерных вычислений корректируется лицом, принимающим окончательное решение [1, 2].

Однако первые из указанных методов не исключают «запрограммированных» ошибок из-за неполноты и/или неточности компьютерных моделей, вторые требуют привлечения дорогостоящих экспертов, и даже «золотая середина» — человеко-машинные процедуры — имеют ограничения как по точности результата, так и по стоимости процесса принятия решений.

Цель данной работы — избежать использования мнений экспертов (которых затруднительно привлечь во многих реальных ситуациях), но при этом повысить степень доверия к результатам автоматического ранжирования. Эта цель достигается путём согласования рейтингов, получаемых разными классическими методами (которые по сути эвристические и поэтому не имеют никакой меры качества, позволяющая их сравнивать друг с другом), а также путём построения итерационного процесса, где каждый следующий шаг использует

результаты предыдущего шага (рейтинги, полученные разными методами) в качестве входных данных (оценок по критериям).

Одним из наиболее удачных оказался следующий вариант итерационного многометодного алгоритма. На первом шаге алгоритма каждым методом из k классических методов решается исходная задача, состоящая из оценок альтернатив по критериям c и весов критериев w :

$$(c_m^n, w^n) \xrightarrow{k} (s_{m,1}^k).$$

Полученную в результате таблицу рейтингов $s_{m,1}^k$ можно рассматривать как входные данные для тех же методов при решении новой задачи. В этой задаче место критериев занимают методы, оценкам c присваиваются значения рейтингов s , а веса w задаются одинаковыми.

На i -м шаге решается задача

$$(s_{m,i-1}^n, w_{i-1}^n) \xrightarrow{k} (s_{m,i}^k, w_i^k); i > 1,$$

при этом используются те же методы, что и на первом шаге, так что размерность результата будет такой же, как и после первого шага. Для расчёта вектора весов на втором и последующих шагах предлагается умножать вектор весов предыдущего шага на матрицу, отражающую углы между векторами рейтингов, полученных различными методами:

$$\vec{w}_i = C_i \vec{w}_{i-1}; C_{k,i}^j = \frac{1 + \cos \left(\frac{\vec{S}_i^k, \vec{S}_i^j}{|\vec{S}_i^k| |\vec{S}_i^j|} \right)}{2},$$

где \vec{S}_i^k — вектор рейтингов альтернатив, полученных k -м методом на i -м шаге [3].

С помощью данного и подобных алгоритмов решались несколько различных (по сути, объёму входных данных, требованиям к результату) задач: задача выбора оптимальной схемы подключения (3 альтернативы, 4 критерия), задача отбора студентов в учебный центр (~ 60 альтернатив и 6–7 критериев). Оказалось, что приведённый выше алгоритм достаточно быстро сходится (не более 6 итераций даже на самой объёмной задаче при предъявлении завышенных требований к точности). Само понятие «сходимости» определяется неоднозначно; в данной работе остановка итераций производилась на основе анализа либо норм отклонений векторов результатов в пространстве

решений, либо последовательностей, отсортированных по рангу альтернатив.

В итоге результаты, полученные комбинированием различных методов (трёх-четырёх из семи имевшихся в наличии), отличались лишь единичными различиями в последовательности из 60 альтернатив. Столь же небольшим оказалось отличие от эталонного решения, которое было составлено на основе мнений экспертов. Таким образом, итерационные многометодные алгоритмы позволили повысить адекватность результатов многокритериального ранжирования, не потребовав участия экспертов в процессе решения.

Литература

1. *Ларичев О.И.* Теория и методы принятия решений. — М.: Логос, 2000. — 296 с.
2. *Любченко В.В.* Переключение методов в интерактивной процедуре принятия решений // Труды Одесского политехнического университета. — 2002. — Вып. 2(18). — С. 2–3.
3. *Jaap S.* Interactive multiple objective programming methods. — Springer, 1981. — P. 114–122. — <http://publishing.eur.nl/ir/repub/asset/10709/HFDSTK.5.PDF>.

УДК 519.816

А.А. Кутафин^{1,2}, Д.А. Некрылов^{1,2}

akutafin@gmail.com, mrdan@frtk.ru

¹ Московский физико-технический институт
(государственный университет)

² ООО «НетКрэкер»

Решение задачи планирования модернизации телекоммуникационной сети с учётом технических и экономических критериев

Возможности экстенсивного пути развития сотовой связи стандарта GSM/EDGE близки к исчерпанию. Идет процесс перехода к новым технологическим решениям, которые лучше соответствуют возрастающим требованиям пользователей к качеству и составу предлагаемых услуг. По некоторым оценкам потребуется обеспечить для домашнего пользователя скорость доступа в среднем около 28 Мбит/с. В современных сетях передачи данных пропускная способность составляет всего лишь от 50 Кбит/с до 1,5 Мбит/с на уровне абонентского доступа.

В рамках данной работы была изучена задача планирования реорганизации участка радиорелейной подсети мобильного оператора с целью увеличения доступной полосы пропускания.

В нашей постановке задачи дано исходное дерево, которое представляет собой контроллер базовых станций (BSC — Base Station Controller) и обслуживаемые им базовые станции (BTS — Base Transceiver Station). Будем считать, что был проведён предварительный анализ текущей пропускной способности соединений и найдены участки, на которых она недостаточна для передачи возросших объёмов трафика.

При решении задачи принимались в рассмотрение следующие критерии:

- стоимость модернизации, или «Cost»,
- остаточная пропускная способность, или «Bandwidth»,
- максимальное расстояние между узлами сети, или «Length».

Также были наложены ограничения на максимальную глубину дерева и сгенерированы запреты на связь двух конкретных BTS (в реальной жизни аналог — стоящее между ними здание). Требуется найти все варианты построения дерева с учётом ограничений и отобрать наилучшие.

Для решения поставленной задачи была использована теория важности критериев, в частности, метод количественных оценок, так как многокритериальный подход в сравнении с однокритериальным имеет ряд преимуществ:

- не рассматривается взвешенная сумма, или линейная свёртка, поэтому критерии с малыми весами тоже имеют существенное значение,
- возможность одновременного учёта и анализа критериев разного рода (экономических и технических),
- результатом анализа является небольшое число недоминируемых вариантов, из которых оператором выбирается наилучший на основе дополнительных соображений или предпочтений.

В качестве тестовой модели был выбран участок реальной сети, состоящий из одного BSC и 33 BTS.

В результате решения получено 7 вариантов с использованием многокритериального анализа и 1 вариант с использованием простой линейной свёртки. Этот вариант минимизирует стоимость, но остальные два критерия имеют нелучшие значения. Для передачи голоса такое решение является подходящим, хотя сеть будет испытывать большие перегрузки при увеличении потоков трафика.

Варианты с большей стоимостью, но и большей пропускной способностью будут предпочтительнее во многих других случаях, например, при планировании сети с учётом перспектив модернизации в будущем.

Варианты с высоким значением критерия «Length» были отсеяны однокритериальной оптимизацией из-за малого веса параметра, но, как показывают результаты, они менее дорогие и могут оказаться наиболее предпочтительными для передачи данных.

Преимущества многокритериального подхода оказались очень существенными. В отличие от скалярного он предоставляет лицу, принимающему решение, несколько наилучших (недоминируемых) вариантов, для сравнения которых системе уже не хватает информации. Впоследствии из них выбирается наиболее подходящий с учётом дополнительных предпочтений и требований сервиса. Это придает системе гибкость по отношению к типу решаемой задачи.

Рассмотренный подход имеет большие перспективы для использования в реальной сети. Он позволяет избежать лишних затрат на обновление оборудования сети, более рационально использовать текущие ресурсы. Это особенно актуально в условиях повсеместного перехода к новым технологиям, например, от GSM к 3G, который требует больших вложений.

Литература

1. Подиновский В.В., Потапов М.А. Теоретические основы и системы поддержки принятия многокритериальных решений. Информационные технологии в науке, образовании, телекоммуникации и бизнесе // Приложение к журналу «Открытые конференции». — 2007. — С. 87–89.
2. Олифер В.Г., Олифер Н.А. Компьютерные сети. Принципы, технологии, протоколы: учебник для вузов. — 3-е изд. — СПб.: Питер, 2007. — 958 с.

УДК 004.7

Н.А. Подольская

nap@math.msu.su

Московский физико-технический институт
(государственный университет)

Московский государственный университет им. М.В. Ломоносова
ЗАО «Интел А/О»

О задаче симуляции компьютерных сетей и методах её решения

Симуляция компьютерных сетей используется при решении различных задач: оценка качества сетевого программного обеспечения, тестирование и оценка алгоритма управления сетью, оценка производительности сетевого соединения и т. д.

Решение задачи симуляции компьютерных сетей сводится к задаче реализации набора сетевых протоколов. Каждый протокол может быть реализован одним из двух методов: 1) программная реализация

или 2) переиспользование реализации этого протокола в операционной системе. Недостатком первого метода является его недостоверность, то есть возможное отличие от реализации протокола в реальной сети. Недостатком второго метода является наличие в реальной сети значительного количества посторонних фреймов, что затрудняет анализ данных в симуляторе.

Примером программной симуляции сетевых соединений может служить симулятор ns-2 [1]. В нем симулируются временные характеристики движения данных в сети. В ns-2 программно реализованы все протоколы сетевого стека TCP/IP с точки зрения времени и продолжительности посылки фреймов. Данный симулятор может использоваться для оценки производительности сети или оценки алгоритма управления трафиком.

Автор доклада использовал ns-2 для оценки алгоритма выбора скорости передачи в сетевых соединениях, работающих по протоколу 802.11 (WiFi) в закрытых помещениях. Выбор скорости передачи WiFi фрейма имеет большое влияние на производительность передачи: при более низкой скорости передачи возрастает надежность, но удлиняется время передачи, при более высокой — сокращается время перелачи фрейма, но уменьшается надежность передачи, то есть возрастает вероятность необходимости перепосылки фрейма, что вызывает дополнительные траты времени.

Для решения данной задачи возникла необходимость заново реализовать в ns-2 физический уровень стека сетевых протоколов, поскольку алгоритм выбора скоростей актуален в условиях затухания радиосигнала, а в ns-2 симулируется свободная от помех среда передачи.

Задача симуляции затухания радиосигнала в закрытых помещениях была решена автором доклада совместно в Ф.Н. Шерстюком [2]. Результатом решения явились симулированные значения уровня шума радиоэфира.

Результаты решения задачи затухания радиосигнала были встроены в исходные тексты симулятора ns-2: по ним определялось отношение уровня сигнала к уровню шума (Signal-to-Noise Ratio, или SNR) в данный момент времени. Заметим, что в ns-2 уровень сигнала зависит от расстояния до передающей станции. Исходя из экспериментальных данных вероятности появления ошибок в процессе передачи фрейма при данном SNR, вычислялась вероятность ошибки в принятом фрейме. При наличии ошибки прием фрейма не подтверждался, и передающая станция вынуждена была передавать фрейм повторно.

Описанное усовершенствование ns-2 позволило провести оценку алгоритмов выбора скорости передачи. Коды алгоритмов были встроены в исходные тексты ns-2, и проведены измерения производительности сетевого соединения для каждого из алгоритмов (рис. 1).

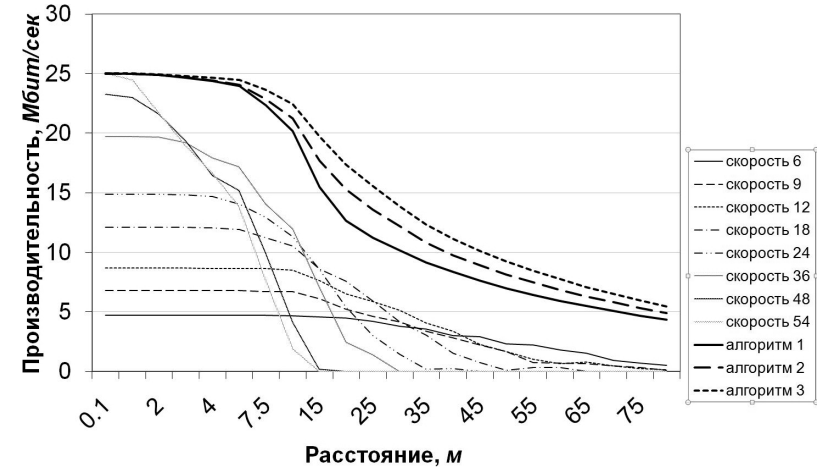


Рис. 1. Графики производительность сетевого соединения при постоянных скоростях передачи и при использовании различных алгоритмов выбора скорости передачи в зависимости от расстояния между WiFi станциями

Переиспользование в симуляторе реализации сетевых протоколов, имеющихся в операционной системе (ОС), часто требует программной реализации недостающей функциональности. Так, например, при решении задачи оценки алгоритма, управляющего сетевым трафиком, требуется включить этот алгоритм в функционирование стека сетевых протоколов ОС. Данное достигается инсталляцией в операционной системе специального сетевого драйвера, например, TUN/TAP [3].

Под руководством автора доклада студентами механико-математического факультета МГУ М.С. Хроповым и А.Д. Ямпольским в рамках выполнения курсовой работы решалась задача симуляции беспроводной сети, состоящей из двух станций, связывающихся по протоколу 802.11. Протокол 802.11 симулировался программно, этот код коммуницировал с драйвером TUN/TAP, инсталлированным в операционной системе, и посылка фреймов имитировала посылку их WiFi станцией, то есть фреймы посылались с задержками, соответствующими стандарту 802.11. Радиопомехи также симулировались

с использованием результатов работы [2]. Симулятор использовался для оценки алгоритма, управляющего посылкой данных через сеть.

Решение задач симуляции компьютерных сетей предполагает комбинирование двух описанных выше методов: программной симуляции сетевых протоколов и переиспользование сетевых протоколов, реализованных в операционной системе. Выбор конкретного решения определяется исходя из требований, предъявляемых к решению, а также из имеющихся ресурсов.

Литература

1. The Network Simulator. — ns-2. URL: <http://www.isi.edu/nsnam/ns/>. Документация и исходные тексты симулятора ns-2.

2. Подольская Н.А., Шерстюк Ф.Н. Задача симуляции затухания радиосигнала: решение и приложения // Фундамент. и прикл. матем. — 2007. — Т 13, № 1. — С. 179–187. URL: <http://mech.math.msu.su/~fpm/ps/k07/k071/k07110.pdf>.

3. Virtual Point-to-Point (TUN) and Ethernet (TAP) devices; <http://vtun.sourceforge.net/tun/>. Документация и исходные тексты драйвера TUN/TAP.

УДК 004.7

М.С. Хропов, А.Д. Ямпольский

agat-mt@mail.ru, corvus57@gmail.com

Московский государственный университет им. М.В. Ломоносова

О решении задачи симуляции сетевого соединения, работающего по стандарту 802.11

Задача симуляции беспроводного сетевого соединения возникла перед авторами доклада при решении задачи разработки алгоритма, управляющего распределением потока данных в сети. Разработанный авторами алгоритм требовал тестирования и оценки производительности сети, получаемой при использовании данного алгоритма. Однако в силу специфики алгоритма недостаточно было исследовать его

поведение на реальной сети. Эффективность работы алгоритма напрямую зависела от состояния радиозэфира, а для его отлаживания было существенно необходимо полностью контролировать трафик. Это привело к задаче моделирования Wi-Fi соединения на отдельно взятом компьютере.

На данный момент существует несколько различных стандартов беспроводной связи. Наиболее распространены IEEE 802.11b/g, также известные под общим названием Wi-Fi. Варианты беспроводного стандарта IEEE 802.11 различаются между собой не только частотами вещания, но и поведением, которое призвано получить максимально возможную скорость при любых условиях среды. Wi-Fi основан на типичном для большинства современных протоколов связи принципе «данные–подтверждение»: после каждой отправленной части информации станция связи ожидает реакции принимающей стороны. В случае, если ответ не приходит, заглушен другим сигналом или же пакет поврежден, предполагается, что в эфире наличествует ещё как минимум одна станция беспроводной связи, оперирующая на той же частоте — обе ждут случайный, увеличивающийся с каждой неудачей отрывок времени перед повторной отправкой. Таким образом, даже в заполненных Wi-Fi сигналами офисных комплексах удаётся создать достаточно пространства для вещания каждой станции. Для случая посторонних шумов, не связанных напрямую с беспроводным обменом информацией, стандарт имеет настраиваемые скорости передачи данных — путём увеличения времени на передачу каждого пакета станция увеличивает шансы восстановления данных при приеме.

Для моделирования беспроводного соединения использовался драйвер Tun/Tap — широко известное виртуальное сетевое устройство, работающее с TCP/IP и Ethernet-фреймами на уровне операционной системы. Конкретно использовался его Win32-аналог, являющийся частью проекта Open-VPN. Это позволило программно симулировать любое поведение пакета в предполагаемой среде, не используя никаких иных средств, кроме компьютера, на котором запущена программа симуляции. С помощью Tun/Tap создаётся виртуальная сетевая карта, которую Windows способен адекватно распознать. Используя специальную C/C++ библиотеку, можно подключать к этой карте драйвер, реализованный пользователем. Взаимодействие с TCP/IP протоколом реализуется с помощью системы событий — драйвер получает все пакеты, посланные на виртуальную карту, и

может в свою очередь посылать другие пакеты назад на уровень приложения, симулируя входящий трафик (рис. 1).

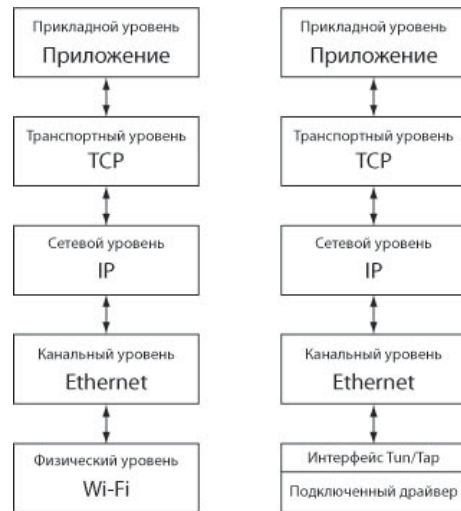


Рис. 1. Стек протоколов TCP/IP с сетевой картой Wi-Fi и с драйвером TAP/UART

Для решения поставленной задачи был написан код драйвера для TAP/UART, моделирующий поведение беспроводного соединения. Были реализованы очередь сообщений, вычисление задержки отправления пакета, возможность потери пакетов, имитирующая помехи, а также настраиваемые условия моделируемой среды. Для вычисления вероятности повреждения данных в зависимости от помех в среде передачи информации были использованы теоретические и экспериментальные результаты статьи [1]. Сами помехи задавались функцией, зависящей от общего уровня шума, проводимости симулируемой среды распространения волн и проектируемого количества других Wi-Fi сетей, вещающих на той же частоте, что в сущности охватывает весь спектр возможных источников помех.

Преимущества такого подхода заключаются в следующем: тестирование алгоритмов стало возможно даже на одной машине без сетевой карты Wi-Fi; любую собираемую по трафику статистику можно отслеживать в процессе работы соединения; на моделируемую карту по определению не посылаются пакеты от сторонних приложений; отладка алгоритмов значительно упрощается благодаря возможности доступа к трафику на любом уровне TCP/IP стека.

Результат решения данной задачи был успешно применен для оценки качества работы алгоритма распределения трафика в беспроводных сетях и его тестирования.

Литература

1. Podolskaya N.A., Sherstyuk F.N. The problem of Wi-Fi radiofading simulation: Solution and applications // Journal of Mathematical Sciences. — 2008. — V. 152, N. 4. — P. 571–577.

Секция высокопроизводительных вычислительных систем

УДК 004.272.42

Д.С. Артамонов

dmitry.artamonov@idm.ru

Московский государственный институт электронной техники
(технический университет)
ООО «ИДМ»

Методы повышения эффективности вычислительного процесса на однородных вычислительных средах

Эффективность вычислений на однородных вычислительных средах [1–4] определяется, с одной стороны, возможностями распараллеливания прикладных вычислительных алгоритмов, а с другой — функциональными возможностями ячеек среды и размерностью матрицы среды, реализуемой в отдельном микрочипе заданной площади.

Таким образом, необходима разработка методов повышения эффективности вычислений (удельной производительности — производительности на единицу площади аппаратуры, отношения производительность/потребление) на однородных вычислительных средах по трём направлениям:

1). Разработка архитектуры процессорного элемента и его функциональности. Разрабатываемый процессорный элемент однородной вычислительной среды представляет собой совокупность независимых устройств, которые предназначены для обработки, хранения и передачи данных [5]. Процессорный элемент включает в себя арифметико-логическое устройство (АЛУ) — для обработки данных по арифметическим и логическим операциям, многофункциональное устройство (МФУ) — для обработки данных по составным алгоритмам — умножении, умножение в полях Галуа, деление, поиск совпадений в

битовой последовательности и т. д., четыре транзитных канала передачи данных — для передачи данных с внешних входов ПЭ на его внешние выходы с программируемой задержкой, равной 0, 1, 2, 3 такта сигнала синхронизации, входной и выходной коммутаторы — для подключения внешних входов и выходов ПЭ к входам и выходам внутренних исполнительных устройств (рис. 1).

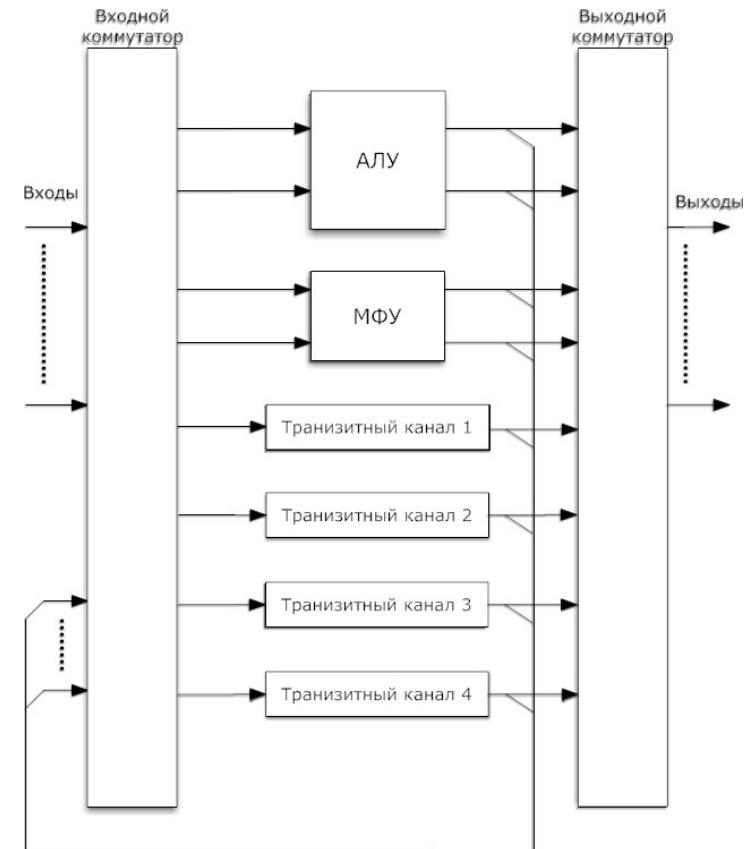


Рис. 1. Структурная схема процессорного элемента

Многофункциональное устройство, в основе функционирования которого лежат принципы конвейерной последовательно-параллельной обработки данных, обеспечивает эффективную с точки зрения затрачиваемых аппаратных ресурсов и времени вычислений обработку данных различных форматов (8-, 16-, 32-, 64-бит).

2). Оптимизация затрат аппаратных ресурсов матрицы вычислительной системы. Вычислительная ячейка однородной вычислительной среды может выполнять прямую или альтернативную функцию из существующего набора функций в зависимости от сигнала условия. Введение поддержки альтернативных вычислений позволяет снизить аппаратные издержки относительно обрабатываемых алгоритмов, включающих ветвления, в 2 раза (рис. 2). В масштабах матрицы процессорных элементов такой выигрыш в аппаратных ресурсах позволяет значительно повысить удельную производительность всей системы в целом.

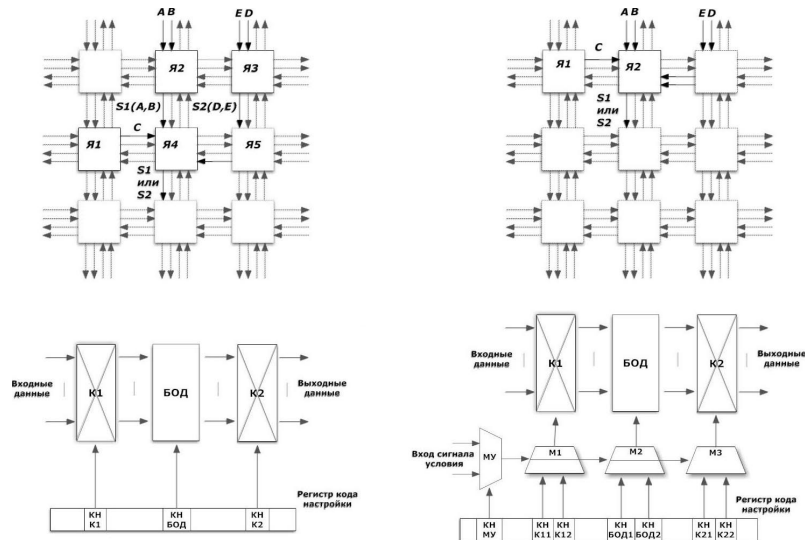


Рис. 2. Аппаратура для реализации алгоритма типа выбор на стандартной ячейке (а) и на ячейке, поддерживающей технологию альтернативных вычислений (б) (активные ячейки и связи выделены жирным, неактивные — пунктиром)

3). Динамическая частичная реконфигурируемость аппаратуры вычислительной системы. Необходимо обеспечить возможность доступа для реконфигурирования к любому индивидуальному ПЭ в любой момент времени. Такая возможность может быть организована посредством введения адресации ячеек матрицы вычислительного процессора по осям X и Y , а также введения индивидуального управляющего сигнала, определяющего режим каждого отдельного ПЭ.

Поддержка динамической частичной реконфигурируемости аппаратуры вычислительной матрицы обуславливает гибкость вычисли-

тельной системы относительно варьируемого диапазона классов обрабатываемых алгоритмов.

Литература

1. Евреинов Э.В. Однородные вычислительные системы, структуры и среды. — М.: Радио и связь, 1981. — 208 с.
2. Богачев М.П. Архитектура вычислительной системы с однородной структурой. — Львов: ФМИ АН УССР, 1981. — С. 15–21.
3. Варшавский В.И., Мораховский В.Б. [и др.]. Однородные структуры. — М.: Энергия, 1973. — 150 с.
4. Шмойлов В.И., Русин Б.П., Кузьо М.Н. Ячейка пульсирующих информационных решёток. — Львов: Меркатор, 2001. — 34 с.
5. Процессор однородной вычислительной среды: пат. 2180969, Российская Федерация: МПК7 G06F15/00, G06F15/163/ Бачериков Г.И., Геворкян В.И., Крохин В.М., Татур В.Ю.; заявка № 2000126004/09, заявл. 18.10.2000, опубл. 27.03.2002.

УДК 004.4'416

В.А. Битнер

wilek@mail.ru

Московский физико-технический институт
(государственный университет)
ЗАО «Московский центр SPARC-технологий»

Система интерпретации промежуточного представления программы в оптимизирующем компиляторе

Начальным этапом трансляции в оптимизирующем компиляторе, созданном и развиваемом для архитектуры «Эльбрус» [1], является преобразование исходного кода программы в промежуточное представление, предназначенное для удобства генерации кода и/или проведения различных оптимизаций [2]. Промежуточное представление исходной программы можно интерпретировать как программный

код, который приводится к машинному коду и исполняется (оно основано на Ассемблере целевой архитектуры, но при этом не является программой на языке Ассемблера).

Принято различать два типа представления исходной программы: высокоуровневый, максимально абстрагированный от особенностей конкретной вычислительной архитектуры и предназначенный главным образом для межпроцедурных оптимизаций, и низкоуровневый, приближённый к архитектуре некоторой конкретной вычислительной системы и предназначенный для оптимизаций внутри процедуры [3]. Данная работа посвящена разработке системы интерпретации промежуточного представления низкоуровневого типа посредством трансляции промежуточного кода в рабочую программную среду, которая обладает необходимыми свойствами надежности, доступности, удобства использования. Техническая реализация трансляции строится на моделировании элементов промежуточного представления Си-кодом, выполняемым с сохранением функциональной эквивалентности. Моделируются не только элементы промежуточного представления, но и аппаратная поддержка исполнения программы на процессоре с архитектурой «Эльбрус», например, моделирование асинхронной предподкачки массивов данных [4] или процесса использования стека данных. Результатом трансляции промежуточного представления является программный код на языке Си, исполняемый на архитектуре x86 и в точности повторяющий функциональность промежуточного кода исходной программы.

Особое внимание в работе уделяется моделированию архитектурных особенностей, при котором достигаются наиболее простые и эффективные решения: реальный регистр моделируется как переменная типа `int`, реальный стек — как массив элементов типа `char`, вызов процедуры — как инициализация «станка» — указатель типа `void` — и вызов по косвенности с использованием средств языка Си и так далее. Общий принцип моделирования основывается на замене всех непереносимых объектов архитектуры «Эльбрус» эквивалентными (в смысле функциональности) и операбельными объектами архитектуры x86.

В процессе разработки оптимизирующего компилятора архитектуры «Эльбрус» система интерпретации промежуточного представления программы актуальна как необходимая основа для создания нового эффективного отладочного средства. Такое отладочное средство будет использоваться для верификации оптимизаций в оптимизирующем компиляторе, особенно в случаях, когда ошибка в оптими-

зации не приводит к явному её обнаружению, то есть проявляется только на этапе исполнения программы.

Литература

1. Волконский В.Ю. Оптимизирующие компиляторы для архитектуры с явным параллелизмом команд и аппаратной поддержкой двоичной совместимости // Информационные технологии и вычислительные системы. — 2004. — № 3. — С. 4–26.
2. Ахо А.В., Сети Р., Ульман Д.Д. Компиляторы: принципы, технологии и инструменты. — М.: Вильямс. — 2003. — 768 с.
3. Дроздов А.Ю., Степаненков А.М. Методы комбинирования алгоритмов анализа и оптимизаций в современных оптимизирующих компиляторах // Компьютеры в учебном процессе. — 2004. — № 11. — С. 3–12.
4. Galazin A.B. [et al.]. A Software Instruction Prefetching Method in Architectures with Static Scheduling // Programming and Computer Software. — 2008. — V. 34, N. 23. — P. 49–53.

УДК 004.35

А.Ю. Богданов

bmstupower@gmail.com

Московский государственный технический университет
им. Н.Э. Баумана
ЗАО «Московский центр SPARC-технологий»

Комплексная отладка контроллера периферийных интерфейсов

В настоящее время компания ЗАО «МЦСТ» разрабатывает контроллер периферийных интерфейсов (КПИ). Он предназначен для использования в высокопроизводительных системах на кристалле с архитектурой «Эльбрус» и поддерживает интерфейсы Ethernet, Sata, IDE, PCI, PCI-Express, USB2.0, AC97, GPIO, IEEE1284, RS232, IOAPIC, PIC, I2C, SPI и IOLink.

На первых этапах разработки отладка КПИ проводилась исключительно программными средствами, реализующими исполнение набора тестов программным модулем, представляющим собой RTL-код

на языке Verilog. Такой подход имел существенные недостатки, основными из которых были:

- низкая скорость выполнения тестов,
- невозможность подключения периферийных устройств.

Возможность их устранения появилась после реализации аппаратного макета КПИ. Было разработано тестирующее устройство (аппаратный контроллер), позволяющее формировать тестовые наборы для непосредственного исполнения на макете.

Через шину PCI инструментальная машина посылает запросы к КПИ в виде PCI фрагментов IOLink пакетов запросов и считывает ответы на ранее поступившие запросы из тестирующего устройства.

Через шину IOLink тестирующее устройство посылает запросы к КПИ и получает ответы на них, которые сохраняются во внешней памяти.

Внешняя память служит для хранения запросов и ответов на них, поступающих в режиме DMA.

Устройство содержит несколько буферов FIFO следующего назначения:

Memory Controller обеспечивает взаимодействие с внешней памятью,

FIFO APIC хранит приходящие от КПИ прерывания,

FIFO Answer накапливает ответы на запросы,

FIFO Sequence последовательно буферизует запросы к КПИ,

Register содержит регистры статуса и управления,

IOlink Controller обеспечивает взаимодействие с IOlink интерфейсом,

PCI Controller обеспечивает взаимодействие с шиной PCI.

Данное устройство позволило повысить скорость тестирования в 100 раз, появилась возможность проверки работы КПИ при подключении различных устройств (USB, SATA и т. д.). Сочетание программных и аппаратных средств позволяет добиться наилучших результатов при отладке устройства (рис. 1, 2).

Литература

1. PCI Local Bus Specification Revision 3.0 // PCI Special Interest Group, 2004.

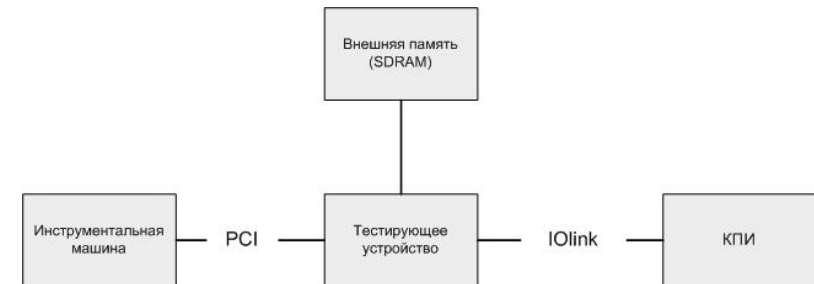


Рис. 1. Функциональная схема макета с тестирующим устройством

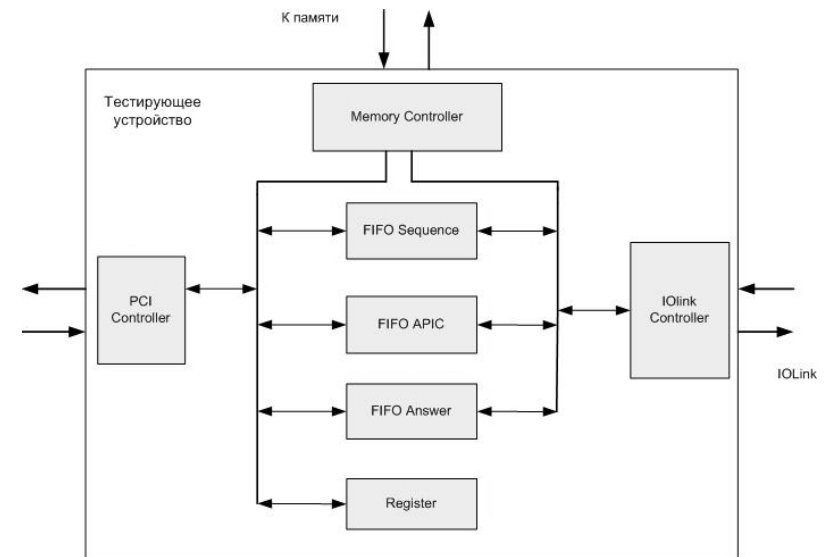


Рис. 2. Функциональная схема тестирующего устройства

УДК 004.3'12

В.В. Воробушков^{1,2,3}, Ю.С. Рябцев^{1,3}

vvv@mcst.ru, ryab@mcst.ru

¹ Московский физико-технический институт
(государственный университет)² ЗАО «Московский центр SPARC-технологий»³ Институт электронных управляющих машин

Использование развязывающих конденсаторов на подложке микропроцессора «Эльбрус-S»

Каждый производитель вычислительной техники стремится получить максимальную производительность вычислительной системы при достаточном уровне отказоустойчивости. Одним из важнейших факторов, влияющих на отказоустойчивость вычислительной системы, а как следствие, и на её производительность, является качественно реализованная система обеспечения питания. Чем стабильнее питание, тем больший существует «запас прочности» по номиналам питания, по температуре, по временным диаграммам. Как только уровень помех по питанию превышает допустимый уровень, приходится либо снижать частоту работы системы, что снижает производительность, либо повышать напряжение питания, что снижает надёжность работы системы. Основными источниками помех в системе питания вычислительной системы являются мощные высокочастотные микросхемы, такие, например, как микропроцессоры, память, системная логика.

Систему фильтрации питания в вычислительной системе можно разделить на три больших части. Первая, это фильтрация питания, приходящего на печатную плату от внешнего источника питания. Фильтрацию помех на этом уровне обеспечивает индуктивность проводов подключения источника питания и электролитические конденсаторы высокой ёмкости на входе в систему. Второй этап фильтрации — это фильтрация на уровне печатной платы. Здесь для обеспечения качественной системы питания необходимо спроектировать качественную топологию подвода номиналов питания, заложить необходимый набор высокочастотных керамических и низкочастотных

электролитических конденсаторов, установить фильтры в местах, где это необходимо. Третий этап фильтрации приходится на сами микросхемы. Даже при создании идеальной системы питания на печатной плате, что в большинстве случаев невыполнимо по техническим или экономическим соображениям, невозможно избавиться от помех по питанию внутри микросхем. Это вызвано падением напряжения на индуктивностях выводов микросхемы. Внутри микросхемы с помехами по питанию можно бороться несколькими способами: уменьшая индуктивность выводов микросхемы, оптимизируя взаимное расположение сигнальных выводов и выводов земли и питания микросхемы и добавляя развязывающие ёмкости на землю и питание. В этом докладе мы рассмотрим второй путь.

Вопрос о стабильности напряжения питания особо остро встал во время отладки, испытаний и отбраковки микропроцессоров «Эльбрус». Был установлен факт, что на номинальном напряжении питания микропроцессор не обеспечивает достаточного «запаса прочности». То, что виной тому стабильность напряжения питания внутри процессора, было подтверждено ухудшением стабильности работы микропроцессора при увеличении индуктивности выводов микропроцессора. Частично эту проблему удалось решить, изменив распиновку процессора, однако полностью от этого эффекта избавиться не удалось.

В настоящий момент готовится к выпуску микропроцессор «Эльбрус-S». По сравнению с микропроцессором «Эльбрус» тактовая частота выросла с 300 МГц до 500 МГц, а общая потребляемая мощность оценочно должна вырасти на 50%. Таким образом, вопрос о стабильности номиналов питания в новом процессоре встал особенно остро. Основываясь на опыте отладки микропроцессора «Эльбрус», становится ясно, что оптимизация распиновки не гарантирует достаточный уровень стабильности напряжения питания и необходимо добавлять развязывающие ёмкости. Ёмкостей, которые можно реализовать на кристалле или образующихся в структуре слоёв земли и питания — десятки, сотни пикофарад, по предварительным оценкам явно недостаточно. Таким образом, остаётся единственный вариант — установка развязывающих конденсаторов на подложку процессора.

Из-за технологических ограничений производства подложки невозможно установить конденсаторы в непосредственную область возникновения помех. Таким образом, требовалось определиться с конфигурацией подложки, обеспечивающей наилучшую помехозащитность питания, и убедиться в целесообразности установки конден-

саторов вообще. На этапе разработки подложки единственный способ провести такую проверку — провести моделирование. В качестве ПО для моделирования была выбрана система AWR Design Environment 2008. Исходные параметры для моделирования, такие, как величина помехи, крутизна фронта, индуктивность выводов, были взяты из экспериментальных данных, полученных при наладке микропроцессора «Эльбрус». Данные по структуре подложки, материалам и параметрам конденсаторов брались из соответствующей документации. При моделировании учитывалась полная структура системы: кристалл — подложка — плата.

В результате моделирования различных вариантов исполнения конденсаторов на подложке были найдены решения, при которых амплитуда помехи уменьшается в 2.5–3 раза. Таким образом, был выбран метод подключения конденсаторов и доказана целесообразность их использования.

Для дальнейшего исследования проблем помехозащищенности питания на уровне корпусирования были введены в распиновку микропроцессора специальные выводы, которые подключаются внутри подложки непосредственно к слоям питания и земли ядра. Если данное решение после проверки окажется успешным, то его можно будет применить к другим микросхемам, выпуск которых запланирован в компании ЗАО «МЦСТ».

В настоящий момент заканчивается разработка корпуса и кристалла микропроцессора «Эльбрус-S» с учётом сформулированных рекомендаций и подготавливаются к производству.

Литература

1. *Kim J. [et al.]*. Separated Role of On-chip and On-PCB Decoupling Capacitors for Reduction of Radiated Emission on Printed Circuit Boards // Proceedings of the IEEE International Symposium on Electromagnetic Compatibility. — 2001. — P. 531–536.
2. *Smith L. [et al.]*. Power Distribution System Design Methodology and Capacitor Selection for Modern CMOS Technology // IEEE Transactions on Advanced Packaging. — 1999. — V. 22, N. 3. — P. 284–291.

УДК 004.318

М.В. Исаев

isaev_m@mcst.ru

Московский физико-технический институт
(государственный университет)
ЗАО «Московский центр SPARC-технологий»

Объединение двух процессорных ядер с архитектурой «Эльбрус» для создания двухъядерной системы на кристалле «Эльбрус-S2»

В настоящее время в ЗАО «МЦСТ» выполняется опытно-конструкторская работа (ОКР), в рамках которой проектируется система на кристалле, содержащая два процессорных ядра архитектуры «Эльбрус» и набор распределенной интерфейсной логики (chipset) из системы на кристалле (СНК) предыдущего поколения «Эльбрус-S», а также кластер из четырёх процессоров обработки сигнальной информации (Digital Signal Processor, DSP). При этом принципиальным требованием является возможность объединения четырёх СНК в рамках одного кластера с аппаратной поддержкой когерентности по протоколу, в котором обращение в память порождает опрос всех процессоров системы (snoop-based протокол) и двух кластеров — с помощью межкластерного коммутатора [1]. Для повышения скорости разработки и снижения затрат было принято решение, согласно которому два ядра СНК объединяются межъядерным коммутатором («контроллером межъядерных взаимодействий» — Core Integration Controller, CIC), организующим их совместный доступ к памяти, периферийному оборудованию и DSP-кластеру. Примечательным является то, что данная система может производиться и совместно с DSP-кластером на одном кристалле как в ведущейся ОКР, так и отдельно от него, являясь двухъядерной СНК «Эльбрус-S2». Процессорное ядро с кэшем первого уровня, кэш второго уровня, устройство доступа к памяти Memory Access Unit (MAU) и набор распределенной интерфейсной логики (chipset) System Interface Controller (SIC) СНК «Эльбрус-S» было решено оставить неизменными (рис. 1).

Основной проблемой при проектировании коммутатора был арбитраж запросов от обоих ядер в систему — необходимо учитывать, что с точки зрения системы запросы должны выглядеть так, будто процессор содержит лишь одно ядро. При этом нужно было решить проблему удваивающегося в пике количества операций чтения и записи в системе. В СНК «Эльбрус-S» и ядро, и chipset могут исполнять до 32 чтений и до 16 записей. Если с удвоением количества ядер число возможных запросов увеличивается, то из-за неизменности SIC количество запросов по чтению/записи, находящихся в системе, должно было оставаться неизменным. Также необходимо было соблюсти корректность семафорных операций с памятью (эксклюзивная работа с определённым адресом) в условиях появления второго ядра, о существовании которого не знает SIC. Для решения поставленной задачи был разработан контроллер запросов MAU (MAU Request Controller, MRQ), состоящий из арбитра с круговым приоритетом, регистров чтения/записи (LDR/STR) в количестве 32 и 16 соответственно по числу регистров чтения и записи MAU, а также добавлены проверки адреса и типа операции на совпадение с используемым семафором в семафорных регистрах. При этом архитектурно реализовано переименование регистров чтения/записи MAU с помощью LDR/STR регистров, что минимизирует изменения в MAU до небольших изменений интерфейсного уровня [2].

Ещё одной проблемой, решённой в процессе разработки контроллера, являлось внесение необходимых доработок в протокол когерентности, использующийся в комплексе на базе СНК «Эльбрус-S» и реализованный в наборе распределенной интерфейсной логики SIC. Для этого был разработан контроллер когерентных сообщений — Coh_Box. Находясь между ядрами и SIC, он, получая запрос из SIC, разбирает его и передаёт нужным ядрам, а когерентные ответы из ядер сохраняет и выдаёт обобщённый когерентный ответ в формате когерентного ответа для одноядерной системы.

Последним функциональным устройством является межинтерфейсный коммутатор. Его основной задачей является сведение внутреннего интерфейса межъядерного коммутатора, достаточно разрозненного, к формату старого интерфейса между MAU и SIC.

Verilog-описание разработанного коммутатора было включено в состав отладочного стенда, где прошло первичное тестирование, доказывающее правильность принятых во время разработки решений. Также стоит отметить, что проведённые тесты производительности показывают превосходство двухъядерной СНК «Эльбрус-S2» относи-

тельно двухпроцессорной системы на базе двух систем на кристалле «Эльбрус-S».

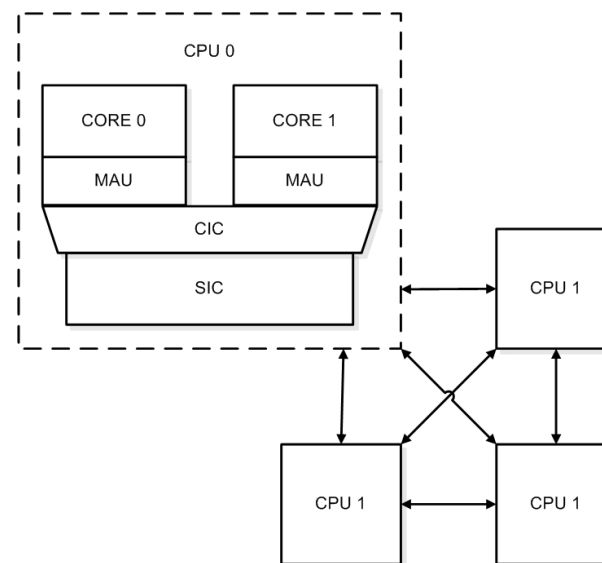


Рис. 1. Структурная схема СНК «Эльбрус-S2»

Литература

1. Ким А.К. [и др.]. Микропроцессорные вычислительные комплексы с архитектурой «Эльбрус» и их программное обеспечение // Вопросы радиоэлектроники. — 2009. — Сер. ЭВТ, вып. 3. — С. 5–36.
2. Таненбаум Э. Архитектура компьютера. 4-е издание. — Питер, 2006. — 704 с.

А.С. Кожин

as_kozhin@mail.ru

Московский физико-технический институт
(государственный университет)
ЗАО «Московский центр SPARC-технологий»

Контроллер памяти DDR2 SDRAM и его система синхронизации в составе системы на кристалле «Эльбрус-S2»

Одной из проблем, возникших при создании системы на кристалле (СНК) «Эльбрус-S2» в совместном проекте ЗАО «МЦСТ» и ГУП НПЦ «ЭЛВИС», выполняемом на базе одноядерной СНК «Эльбрус-S» (предшествующая разработка ЗАО «МЦСТ» [1]), стала поддержка работы с памятью.

Тактовая частота оперативной памяти СНК МП «Эльбрус-S» составляет 250 МГц (соответственно стандарту DDR2-500 [2], пропускная способность одного канала 4 Гбайт/с). В связи с удвоением числа универсальных ядер в СНК «Эльбрус-S2» для обеспечения эффективной работы было необходимо увеличить и пропускную способность памяти (стандарт DDR2-800). С этой целью были проведены все необходимые доработки уже существующего двухканального контроллера памяти для обеспечения пропускной способности 6,4 Гбайт/с на канал.

Контроллер памяти СНК «Эльбрус-S» работал на частоте 250 МГц, которая совпадала с частотой памяти и была в 2 раза меньше системной частоты, на которой формировались запросы и посылались ответы в систему. В новой микросхеме системная частота осталась равной 500 МГц, а физический уровень памяти работает на 400 МГц, причём из-за сложности оптимизирующего ядра контроллера частота его работы составляет 200 МГц.

Доработанный контроллер памяти MC_top (рис. 1) состоит из двух одноканальных контроллеров MC#1 и MC#2, модулей интерфейса системы с контроллерами mc_int#1 и mc_int#2, модуля приёма данных их памяти mc_mrg и модуля формирования служебных сигналов согласования частот mc_clabel. Взаимодействие между па-

мятью и контроллером обеспечивают модули физического интерфейса mc_phy#1 и mc_phy#2.

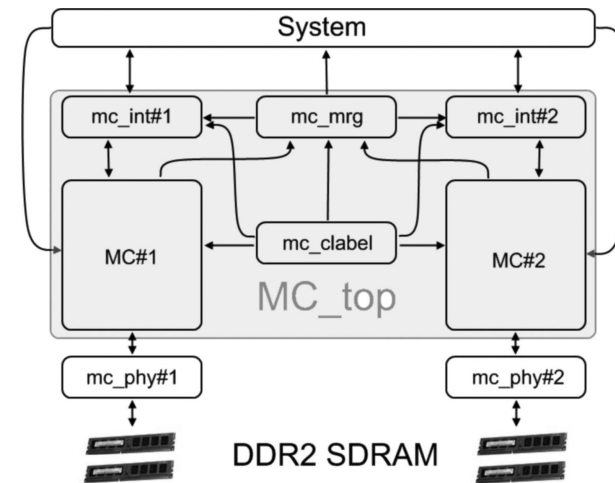


Рис. 1. Структурная схема двухканального контроллера памяти DDR2 SDRAM

Основной сложностью при усовершенствовании контроллера стала разработка новой системы синхронизации. В СНК «Эльбрус-S2» число доменов синхронизации увеличилось до трёх, кроме того, отношение частот стало нецелым ($5/2$, $5/4$), что не позволило использовать старую схему. Для междоменного согласования были использованы метки пересинхронизации, которые учитывают положение фронта высокой частоты относительно фронта низкой. Главными преимуществами этого решения являются минимальная задержка и фиксированная временная диаграмма передачи данных между доменами разных частот, что облегчает поиск и устранение ошибок.

Формирование меток пересинхронизации осуществляется в модуле mc_clabel. Главной задачей при этом стала реализация гибкой настройки сигналов согласования для различных соотношений частот. Была разработана схема на основе управляющих регистров с циклическим доступом. Их длина соответствует периоду биений синхросигналов двух доменов, выраженному в тактах домена с большей частотой, а значения бит — разрешённым фронтам передачи/приёма (для их определения было проведено специальное исследование). Текущее значение метки загружается из управляющего регистра по фронту более высокой частоты. Для передачи данных между доменами с

близкими частотами (с минимальным временным интервалом между фронтами синхросигналов 0,5 нс и меньшим) используются дополнительная метка и регистры, работающие по отрицательному фронту синхросигнала.

В связи с изменением схемы синхронизации существенные изменения введены в модули интерфейса с системой `mc_int#` и модуль приёма данных из памяти `mc_mrg`. Были разработаны модули передачи данных между доменами с учётом меток пересинхронизации для всех соотношений частот. Также в связи с увеличением пропускной способности памяти были расширены каналы выдачи данных по чтению в другие процессоры и при операции Чтение–Модификация–Запись.

Серьезные доработки внесены и непосредственно в контроллеры памяти. Разработан новый модуль формирования команд с интерфейсом DDR2 и поддержкой режима 2T (режим расширения фазы команды/адреса на интерфейсе DDR2 SDRAM до двух тактов). Поскольку весь интерфейс контроллеров с физическим уровнем `mc_phy#` работает на частоте памяти, а ядро контроллера — на частоте в два раза ниже, были реализованы дополнительные схемы пересинхронизации между этими доменами. Кроме того, были изменены все временные ограничения в соответствии со стандартом DDR2 — новые параметры учитывают задержки пересинхронизации и наличие дополнительных регистров, которые пришлось добавить при проведении синтеза.

Несмотря на увеличение числа доменов синхронизации, минимальное время доступа в память осталось равным 60 нс, как и в СНК «Эльбрус-S». Это достигнуто благодаря правильно выбранной схеме синхронизации и оптимизациям при её реализации. (При синтезе контроллера памяти в Design Compiler фирмы Synopsys было проведено исследование по заданию правильных ограничений на передачу между доменами в соответствии с реализованной схемой.)

Описанные доработки контроллера памяти могут быть использованы и в других проектах. Уже сейчас новый контроллер применяется не только в СНК «Эльбрус-S2», но и в микропроцессоре «МЦСТ-4R», построенном на архитектуре SPARC. Внесенные изменения позволяют осуществлять дальнейшее повышение частоты поддерживаемой памяти благодаря работе ядра только на половинной частоте. Существенно, что разработанная система синхронизации, которая имеет минимальные задержки передачи данных и отличается гибкой настройкой рабочих частот (поддерживает различные соотношения частот и задание любого количества доменов), может быть

использована и в других устройствах, где есть разные домены синхронизации.

Литература

1. Волконский В., Ким А., Назаров Л., Перекатов В., Фельдман В. Микропроцессоры и вычислительные комплексы российской компании МЦСТ // Электроника. — 2008. — № 8.
2. DDR2 SDRAM Specification, version JESD79-2E // www.jedec.org, JEDEC. — 2005.

УДК 004.414.23

А.Н. Мешков
alex@mcst.ru

ЗАО «Московский центр SPARC-технологий»

Разработка модели вычислительного комплекса «Эльбрус-S»

В настоящее время моделирование новых компьютерных архитектур является незаменимым средством при их разработке. Возрастающая сложность аппаратуры затрудняет понимание поведения как отдельных устройств, так и системы в целом, её особенностей и нововведений. Моделирование позволяет помочь в разработке как программного обеспечения, так и в проектировании некоторых архитектурных особенностей, и, кроме того, оно незаменимо, когда проектируемая аппаратура ещё не готова.

Симуляторы вычислительного комплекса [1] целиком моделируют целевой компьютер с момента загрузки, что позволяет исполнять системные приложения, такие, как программа начальной загрузки и операционная система. Возможность загрузки немодифицированной операционной системы и пользовательских приложений крайне важна, поскольку она позволяет исследовать широкий спектр задач.

Архитектурная линия «Эльбрус» представлена в составе текущих разработок системой на кристалле «Эльбрус-S». С точки зрения реализации моделирующий комплекс представляет собой развитие модели вычислительного комплекса «Эльбрус-3M1» [2, 3].

Микросхема «Эльбрус-S» представляет собой одноядерную систему на кристалле (System-on-Chip, SoC), структурная схема которой представлена на рис. 1. На одном кристалле размещаются процессорное ядро «Эльбрус» с кэшем второго уровня (Elbrus Core), системный контроллер (System Controller), 2 контроллера памяти (Memory Controller) для микросхем памяти DDR2 SDRAM, контроллеры каналов доступа к другим процессорам (InterProcessor Channel Controller), контроллер канала ввода-вывода (Input/Output Channel Controller) [4]. Поскольку модули памяти принадлежат каждому процессору по отдельности, то многопроцессорный вычислительный комплекс на основе «Эльбрус-S», в отличие от вычислительного комплекса «Эльбрус-3М1», является системой с неоднородным доступом в память (NUMA — Non-Uniform Memory Architecture), то есть время доступа процессора к локальной и удалённой памяти различно [5].

Для поддержки описанной архитектурной особенности в модели было реализовано программно-видимое представление системного коммутатора. В него включены:

- поддержка маршрутизации запросов между абонентами кристалла, такими, как процессорное ядро, канал ввода-вывода, контроллеры памяти и внутренние ресурсы системного коммутатора;
- поддержка доступа во внешние по отношению к расположению запросчика кристаллы;
- собственно внутренние ресурсы кристалла, такие, как регистры.

Другим отличием является внесение в систему команд «Эльбрус» некоторых изменений. Был добавлен ряд новых регистров, а существующие регистры были дополнены новыми полями. Появились новые арифметические целочисленные, вещественные и упакованные операции, а также операции преобразования. У отдельных существующих инструкций были внесены изменения в контроль операндов, изменились условия выдачи прерываний при обращении в память и т. д. Все это потребовало внесения изменений и в устройство моделирующего комплекса.

Помимо вышперечисленного, в вычислительном комплексе на базе SoC «Эльбрус-S» была существенно изменена структура и набор устройств ввода-вывода. Моделирование полного перечня устройств, присутствующих в вычислительном комплексе, необходимо для того, чтобы сделать программную модель полностью неотличимой от аппаратной с точки зрения системного программного обеспечения для широкого класса различных задач. Поэтому потребовалась разработка с нуля моделей ряда устройств, таких, как обычный и виртуаль-

ный корневой pci-мосты, контроллер ethernet, контроллер шин i2c и spi, контроллеры USB и SATA и т. д. Некоторые другие устройства, такие, как контроллер последовательного интерфейса, IDE-контроллеры, а также контроллеры прерываний (PIC, IOAPIC), были позаимствованы из других проектов и доработаны.

На основании вышеописанных принципов была разработана программная модель вычислительного комплекса на базе SoC «Эльбрус-S». Модель активно применяется в разработке поставляемого ЗАО «МЦСТ» системного программного обеспечения, на ней успешно загружаются и работают операционные системы семейства Linux, а также Windows с использованием системы двоичной динамической трансляции.

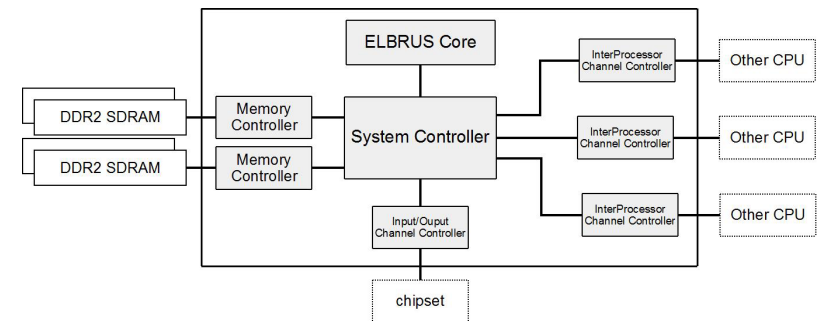


Рис. 1. Структурная схема системы на кристалле «Эльбрус-S»

Литература

1. Herrod S.A. Using Complete Machine Simulation to Understand Computer System Behavior // Doctoral dissertation, Stanford Univ., 1998.
2. Щербаков Е.С. Разработка эталонной потактовой модели микропроцессорной системы «Эльбрус-3М» на базе функциональной модели // Сборник трудов XXXI Международной научной конференции «Гагаринские чтения». — 2006.
3. Щербаков Е.С. Единый структурный подход к объектно-ориентированному проектированию программного моделирующего комплекса. Моделирование новой процессорной архитектуры // Сборник трудов ИМВС РАН № 4. — 2003. — С. 119–122.
4. Волконский В.Ю., Ким А.К. Развитие идей параллелизма в архитектуре вычислительных комплексов серии «Эльбрус» // Институт

проблем управления РАН. Четвертая Международная конференция «Параллельные вычисления и задачи управления» РАСО. — 2008.

5. *Шерстнёв А.Е., Зайцев А.И.* Организация межпроцессорного обмена в многокластерных системах на базе микропроцессоров «Эльбрус-S» и «МЦСТ-4R // Вопросы радиоэлектроники. — 2009.

УДК 004.416.3, 004.428.2

И.В. Москаленко

neron@mcst.ru

ЗАО «Московский центр SPARC-технологий»

Институт электронных управляющих машин

Российский государственный технологический университет
им. К.Э. Циолковского

Вопросы портирования компонент общесистемного программного обеспечения в защищенный режим микропроцессора «Эльбрус»

Защищенной называется такая организация вычислений, при которой одна программная единица (процедура, модуль, объект и т. п.) не может получить несанкционированный доступ к данным другой. Примером архитектурной поддержки этого режима работы является микропроцессор «Эльбрус» [1] и построенный на его базе вычислительный комплекс «Эльбрус-3М1», работающий под управлением Linux-подобных ОС, которые обеспечивают для пользователя, в дополнение к традиционному, и семантический режим защищенного исполнения кода. Для этого режима существенно усложняется задача портирования пользовательских приложений и общесистемных библиотек, в связи с чем в рамках развития общего программного обеспечения (ОПО) «Эльбрус» была поставлена задача — обеспечить в режиме защищенных вычислений использование стандартных библиотек libpthread и libncurses из состава ОПО. Основные аспекты проведенной работы представлены в [2]. В этом докладе рассматривается одна из принципиальных проблем, решенных в ходе её выполнения.

Речь идёт об ошибках, которые могут присутствовать и непосредственно в коде исходных библиотек. Например, в силу контроля обращения к данным, основанного на тегах, в защищенном режиме недопустимы некоторые вольности, которые широко используются при программировании на языке C. Нами были выделены традиционные ситуации, которые приводят к такого рода проблемам, а также предложены решения для некоторых экзотических случаев, обусловленных программными соглашениями целевой архитектуры. Так, в рассматриваемой реализации pthread осуществляется пересылка данных типа «указатель» через программные каналы (pipe), приводящая к потере адресного тега данных. В то же время присвоить тег полученным данным не представляется возможным из-за привилегированности данной операции в архитектуре «Эльбрус». В такой ситуации было найдено альтернативное решение, которое потребовало определённой реорганизации работы с каналами.

В настоящий момент основной этап портирования библиотек завершен, их базовый (а также частично расширенный) функционал можно свободно использовать. Качество работы оценивается тестами, входящими в состав исходных библиотек, а также сторонними открытыми пакетами тестов.

Необходимо отметить, что портированные библиотеки имеют обратную совместимость и могут быть использованы и для незащищенного режима.

Литература

1. *Ким А.К., Волконский В.Ю., Сагин Ю.Х., Семеновичин С.В., Фельдман В.М., Груздов Ф.А., Парахин Ю.Н., Михайлов М.С., Слесарев М.В.* Микропроцессорные вычислительные комплексы с архитектурой «Эльбрус» и их развитие // Труды III Международной научно-практической конференции «Современные информационные технологии и ИТ-образование». — М.: МАКС-пресс, 2008.
2. *Москаленко И.В., Rogov P.Ю.* Портирование библиотек в защищенный режим архитектуры широкого командного слова микропроцессора «Эльбрус // Труды XXXV-й международной молодёжной научной конференции «Гагаринские чтения». — 2009.

УДК 004.318

Н.Ю. Поляков^{1,2}, П.А. Смольянов^{1,2}

potolok2@mail.ru, smolyanov_p@mcst.ru

¹ Московский физико-технический институт
(государственный университет)² ЗАО «Московский центр SPARC-технологий»

Разработка контроллера встроенного интерфейса AXI в составе системы на кристалле

В ходе выполнения совместной опытно-конструкторской работы ЗАО «МЦСТ» и ГУП НПЦ «ЭЛВИС» ведут разработку СБИС цифровой обработки сигнала типа «система-на-кристалле» (СНК) для применения в комплексах обработки радиолокационной информации в режиме жёсткого реального времени. В состав СНК входят два универсальных ядра с архитектурой «Эльбрус» [1] и контроллер системного обмена разработки МЦСТ и четырёхъядерный сигнальный процессор (DSP-кластер), разработанный компанией «ЭЛВИС». Контроллер системного обмена позволяет через LVDS-каналы объединять до четырёх СНК в одну систему с распределённой памятью (рис. 1).

Связь универсального процессора с процессорными ядрами цифровой обработки сигналов (DSP0 — DSP3) осуществляется на основе интерфейса, выполненного по стандарту AMBA AXI 3.0 фирмы ARM [2], для чего был разработан контроллер AXI. Обмен данными между DSP и универсальными ядрами может инициироваться обеими сторонами.

Ядрам «Эльбрус» для управления работой DSP-ядер доступны все адресуемые ресурсы кластера. Процессорные чтения и записи памяти и регистров DSP не отличаются от IO-обмена с внешними устройствами и характеризуются низким темпом поступления запросов, порядка 1 запроса в 60 тактов от каждого ядра.

DSP-кластер содержит в своём составе 8-канальный контроллер DMA (Direct Memory Access), который, выполняя задания от универсальных ядер, осуществляет обмен данными между ядрами DSP, памятью DSP и памятью всей СНК в качестве устройства AXI Master.

Обмен с памятью СНК именуется DMA-обменом и предназначен для передачи данных DSP-кластеру на обработку. Для него характерна передача больших массивов данных.

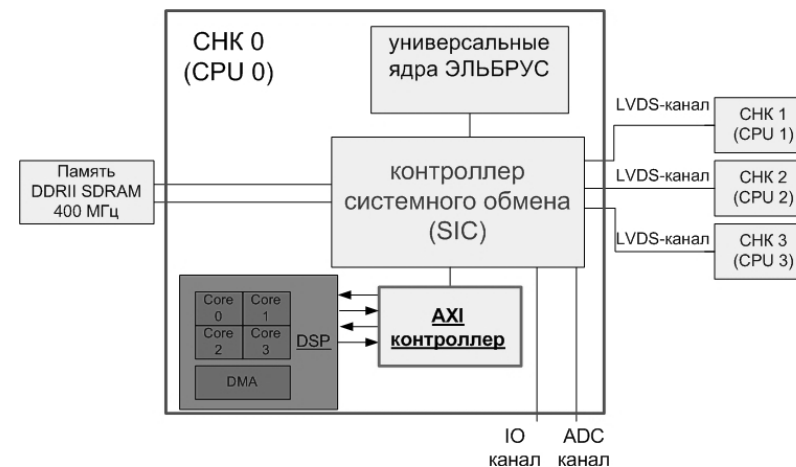


Рис. 1. Структурная схема СНК

В процессе работы была создана синтезируемая Verilog-модель 4-канального AXI-контроллера (DMA-чтение, DMA-запись, IO-чтение, IO-запись), учитывающая особенности обоих типов обменов. Контроллер состоит из интерфейсной части и самого контроллера и отличается следующими свойствами:

1. В интерфейсной части реализовано преобразование пакетов данных и запросов из формата AXI в формат, поддерживаемый ядрами «Эльбрус». При этом выдача запросов оптимизирована таким образом, что запросы со смежными адресами объединяются в один. Это позволяет увеличить темп обработки DMA-чтений.

2. Реализована поддержка когерентности: данные в ответ на запрос по DMA-чтению могут одновременно прийти как из памяти DDRII SDRAM, так и из кэш-памяти любого процессорного ядра.

3. В устройстве используются буфера данных с размером ячейки, изменяемым в зависимости от объёма запрашиваемых данных, что обеспечивает полное использование буфера.

4. Оптимизирована обработка запросов на чтение из процессора. В контроллер системного обмена на запрос по чтению всегда выдаётся 32 или 64 байта, но в AXI отправляется запрос на чтение необходимого количества двойных слов (8 байт). Пришедшие данные встав-

ляются в ячейку буфера данных, соответствующую последним разрядам адреса.

С помощью Verilog-модели AXI-контроллера были измерены фактические характеристики устройства: для DMA-обмена измерялась средняя скорость передачи данных, а для IO-запросов — время обработки (от поступления в контроллер запроса до выдачи ответа). Средняя скорость передачи данных при DMA-чтении достигает 2,7 Гбайт/с, при записи — 4 Гбайт/с (частота синхронизации каналов данных 500 МГц). Длительность IO-чтения составила в среднем 11 тактов, IO-записи — 18 тактов (отметим, что средняя длительность доступа к оперативной памяти составляет 60 тактов).

Помимо применения в рамках данного проекта разработанный контроллер может быть использован в будущих проектах. При включении в состав системы других устройств ввода-вывода можно изменить лишь его интерфейсную часть, а остальное оборудование оставить без существенных изменений.

Литература

1. Волконский В., Ким А., Назаров Л., Перекатов В., Фельдман В. Микропроцессоры и вычислительные комплексы российской компании МЦСТ // Электроника. — 2008. — № 8.
2. AMBA AXI Protocol Specification, version 1.0 // ARM Limited. — 2004.

УДК 004.31

Г.М. Расулов

gadji_r@mcst.ru

Московский государственный институт электронной техники
(технический университет)
ЗАО «Московский центр SPARC-технологий»
Институт электронных управляющих машин

Конструктивно-технологические задачи создания прототипа двухъядерного микропроцессора с архитектурой «Эльбрус»

Создание отечественного двухъядерного микропроцессора с архитектурой «Эльбрус», предназначенного для применения в системах обработки радиолокационной информации, потребовало предварительной разработки прототипа на программируемых логических схемах (ПЛИС). В нем использованы одиннадцать ПЛИС фирмы Altera семейства Stratix EP3SL340F1760 и одна ПЛИС EP2S130F1020. Расчётная тактовая частота прототипа — 150 МГц. Конструкция состоит из системной платы и 2-процессорных плат (рис. 1). Для улучшения качества сигнала и сохранения целостности сигналов процессорные платы подключаются к системной через высокочастотные технологические соединители.

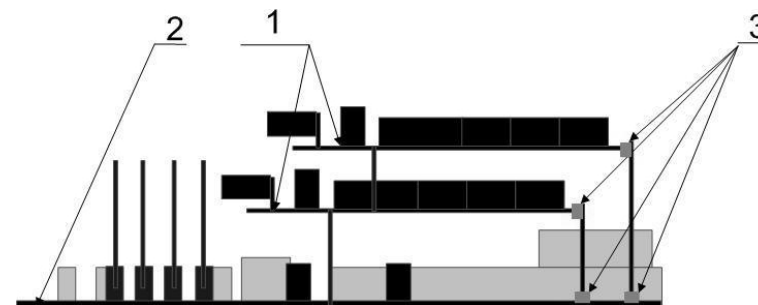


Рис. 1. Конструкция прототипа «Кубик — ку» (вид сбоку): 1 — процессорные платы, 2 — системная плата, 3 — высокочастотные соединители

Основной проблемой при проектировании печатных плат прототипа было большое количество связей. Как видно из рис. 2, связь одной ПЛИС с тремя соседними на процессорной плате обеспечивается 40, 14 и 30 десятиразрядными высокочастотными шинами. Все интерфейсы процессора работают в стандарте SSTL 1.8V. Высокая плотность межсоединений, удовлетворение требований спецификации по перекрёстным помехам и волновому сопротивлению [1], недостаточное пространство для размещения всех компонентов и сигнальных шин привели к очень высокой плотности трассировки и монтажа. Возникла необходимость перехода на 14 слоев, из которых сигнальных только 6.

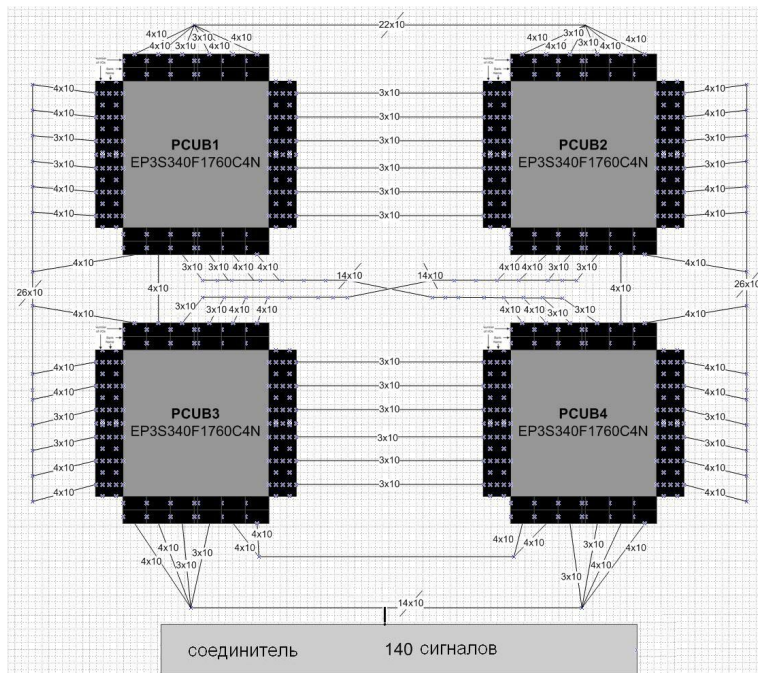


Рис. 2. Связи между ПЛИСами процессорной платы

Одна из возникших проблем состояла в том, что 14 десятиразрядных высокочастотных шин приходили через разъем от системной платы на процессорную плату и разветвлялись в полном составе на две ПЛИС (рис. 2). Моделирование в NuregLinux выявило искажение формы сигналов на этом участке в результате разветвления. Из-за высокой плотности трассировки и монтажа на плате не удалось вы-

ровнять ветви каждого сигнала между собой при том, что не было возможности разместить резисторы для согласования. Анализ показал, что функциональность 14-ти шин, сохранение которой и требовало разветвления, путём ряда схемотехнических решений может быть обеспечена 7-ю шинами. В связи с этим возникла возможность разделить полный состав шин на две равные группы, каждая из которых соединялась с одной из двух ПЛИС без разветвления.

Ещё одной из важных проблем было охлаждение. У каждой из ПЛИС по расчётам Quartus II рассеиваемая мощность составила 33 Вт. Учитывая эти данные, были подобраны радиаторы, и конструкция рассчитана так, чтобы обеспечить обдув ПЛИСов от вентиляторов корпуса.

На данный момент прототип изготовлен и находится в стадии отладки. Текущие результаты отладки свидетельствуют о том, что основные задачи, поставленные при проектировании данного прототипа, решены успешно.

Литература

1. Воробушков В.В., Рябцев Ю.С., Тимофеев В.К. Особенности разработки топологии системы питания в многослойных печатных платах современных вычислительных устройств // Вопросы радиоэлектроники. — 2009.

УДК 004.052.42

М.П. Рыжов

ryzhov@mcst.ru

Московский физико-технический институт
(государственный университет)
ЗАО «Московский центр SPARC-технологий»

Создание генератора тестов с самомодифицирующимися кодами для многопроцессорной системы на базе микропроцессора «Эльбрус-S»

Высокая стоимость повторного выпуска микросхем вынуждает разработчиков тестировать интегральные схемы до их физического производства. Задача верификации состоит в создании тестирующего окружения, обеспечивающего корректность процесса проектирования на всем его протяжении так, чтобы интегральная схема была полностью протестирована до её физического выпуска. Согласно статистике, 70% цикла разработки посвящено верификации [1].

Одной из отличительных особенностей микропроцессоров серии «Эльбрус-S» является возможность исполнения кода, написанного для платформы x86, путём использования бинарной трансляции [2]. В то же время тесты для микропроцессора «Эльбрус-S» верифицируют только выполнение программ для исходной архитектуры и не учитывают особенности работы бинарного транслятора. Непосредственное использование бинарного транслятора для тестирования вычислительного комплекса на этапе проектирования осложнено высокой сложностью транслятора. Тестированию с помощью RTL-модели препятствует недостаточная скорость работы. Использование прототипа вычислительного комплекса не позволяет передавать достаточный для локализации ошибки объём информации о ходе исполнения [3]. Таким образом, возникает необходимость создания тестов, обеспечивающих корректность работы процессора в режимах, характерных для исполнения бинарного транслятора. Одной из его особенностей является модификация кода программы во время её исполнения, что вызвано необходимостью перехода от интерпретации фрагмента к компиляции или его перекомпиляции с измененной степенью опти-

мизации. Протестировать работу процессора в этом режиме можно с помощью тестов, модифицирующих свой код в процессе исполнения.

В данной работе рассматривается развитие генератора тестов с самомодифицирующимися кодами применительно к использованию создаваемой микросхемы в двухпроцессорной конфигурации вычислительного комплекса. Реализована система синхронизации, позволяющая сократить влияние динамических эффектов, вызванных параллельным исполнением кода, на воспроизводимость результатов. Для обеспечения массового поточного тестирования разработана система массовой генерации случайных тестов, обеспечивающая полную загрузку прототипа путём параллельной генерации и корректность генерируемых тестов. В систему запуска тестов на прототипе вычислительного комплекса внесены доработки, позволяющие запускать многопроцессорные тесты.

Использование разработанной системы генерации и прогона тестов позволило выявить ряд ошибок в логике микропроцессора.

Литература

1. *William K.Lam*. Hardware Design Verification: Simulation and Formal Method-Based Approaches, 2005.
2. ТВГИ. 00448-01 33 Микросхема интегральная 1891ВМ4Я. РУКОВОДСТВО ПРОГРАММИСТА.
3. ТВГИ. 469555.090ПЗ Прототип микропроцессора «Эльбрус-S». Пояснительная записка.

УДК 004.222

Х.В. Чан¹, А.И. Грушин²

trankhanh.miptvn@gmail.com, aigrushin@ipmce.ru

¹ Московский физико-технический институт
(государственный университет)² Институт точной механики и вычислительной техники
им. С.А. Лебедева РАН

Использование десятичной арифметики в вычислительной технике

Первые вычислительные машины использовали десятичную систему для представления чисел, так как эта система привычна для человека. Но уже в 1946 году фон Нейман предложил перейти к использованию двоичной системы из-за её экономичности, простоты и большей скорости, она была оптимальна для использования в научно-технических вычислениях. Важным аргументом за использование двоичной системы являлось то, что элементная база, используемая для построения вычислительных машин, имеет два устойчивых состояния и больше подходит для реализации именно двоичной системы. Кроме того, неарифметические операции, выполняемые вычислительной машиной, имеют логическую природу и описываются средствами булевой алгебры, в которой переменные имеют только два значения: 0 и 1. К 60-м годам прошлого века большинство вычислительных машин использовали двоичную систему.

В последние 10 лет опять возник интерес к использованию десятичной системы в вычислительной технике. Это обусловлено как развитием технологии, которая позволяет изготавливать микропроцессоры, содержащие более 2 миллиардов транзисторов, и тем самым использовать более затратные с точки зрения аппаратуры алгоритмы, так и недостатками, присущими двоичной системе. К ним относится необходимость преобразования входных данных, обычно представляемых в десятичном виде, в двоичный вид и обратный перевод результатов в привычный для человека десятичный вид. Особенно это сказывается на вычислениях, связанных с финансами, налогами, коммерцией. Такие вычисления обычно характеризуются большим объёмом входных данных и малой глубиной вычислений. Кроме того, финансовые вычисления выполняются по правилам, отличным от

правил научных вычислений. Результат, полученный компьютером, должен соответствовать результату, который можно вычислить вручную.

Существенным недостатком является то, что не все десятичные числа можно точно представить в двоичном виде, например, 0,1 можно представить в двоичной системе только приблизительно. В 1991 году во время войны в заливе батарея американских зенитных ракет «Пэтриот» не смогла перехватить запущенную иракцами ракету «Scud» из-за погрешности определения времени. При вычислениях нужно было умножать время, задаваемое тактовым генератором компьютера, на 0,1. В компьютере это число хранилось с точностью 24 двоичных разряда. Скорость ракеты «Scud» равнялась 1700 м/с, за 100 часов работы системы ошибка в определении времени достигла 0,34 секунды, что соответствовало ошибке в измерении расстояния 500 метров. Ракета попала в казарму американских солдат, при этом погибли 28 человек.

Поддержка десятичной арифметики осуществляется как в языках программирования, таких, как COBOL, Basic, Rexx, Java, C#, так и в библиотеках. Исследования показывают, что этого недостаточно, так как некоторые прикладные программы от 50 до 90% времени тратят на десятичные вычисления, что свидетельствует в пользу аппаратной реализации десятичной арифметики, которая позволяет ускорить выполнение операций десятичной арифметики в 10^2 – 10^3 раз по сравнению с программной реализацией.

Целочисленная десятичная арифметика не обеспечивает необходимую точность и диапазон. Например, ряд последовательных умножений при вычислении сложных процентов приводит к резкому увеличению количества разрядов, необходимых для представления точного произведения, тем самым требуется округление и использование десятичной арифметики с плавающей запятой.

Сначала в некоторых микропроцессорах (Motorola 68x, IBM PowerPC, HP PA-RISC) появилась небольшая аппаратная поддержка десятичной арифметики, а в 2007 году в микропроцессорах IBM Power6 и z9 появилось DFU (десятичное устройство с плавающей запятой). В советском суперкомпьютере Эльбрус 2 ещё в 70-е годы было устройство двоично-десятичных преобразований.

В 2008 году была принята пересмотренная версия стандарта на двоичную арифметику с плавающей запятой, в новой редакции этого документа IEEE Std 754-2008, кроме двоичных чисел с плавающей запятой, рассматриваются десятичные числа с плавающей запятой.

Требования стандарта могут выполняться в аппаратуре, программным способом или любой их комбинацией. Принятие этого стандарта даёт новый импульс к созданию аппаратуры, реализующей десятичную арифметику с плавающей запятой.

Стандарт определяет 3 десятичных формата: 32, 64 и 128 разрядов. Все эти форматы можно использовать для обмена информацией, а для вычислений используются основные форматы 64 и 128. Десятичное число представляется в виде

$$(-1)^S \cdot 10^{(E-bias)} \cdot C,$$

где S — знак числа, E — смещённый порядок, $bias$ — смещение, C — мантисса. Существуют два метода кодирования мантиссы: двоичное (Binary Integer Decimal) и десятичное (Densely Packed Decimal). В первом случае вся мантисса рассматривается как двоичное число, во втором — каждые 10 двоичных разрядов соответствуют трём десятичным цифрам.

Реализация, особенно аппаратная реализация, десятичной арифметики с плавающей запятой — актуальная задача современной вычислительной техники, об этом свидетельствует тот факт, что на последней конференции по компьютерной арифметике 11 докладов из 34 были посвящена этой проблеме.

Литература

1. *Buchholz W.* Fingers or Fists? (The Choice of Decimal or Binary Representation) // Communications of the ACM. — 1959. — V. 2, N. 12. — P. 3–11.
2. *Cowlshaw M.* Decimal Floating-Point: Algorithm for Computers // Proc. 16th IEEE Symposium on Computer Arithmetic. — 2003. — V. 1. — P. 104–111.

УДК 004.3'1

А.Е. Шерстнёв

andrewsh84@gmail.com

ЗАО «Московский центр SPARC-технологий»
Институт электронных управляющих машин

Применение программируемых логических интегральных схем для решения задачи автоматической генерации тестовых кодов

Автоматическая генерация тестовых кодов (ATPG — Automatic Test Pattern Generation) — процесс формирования тестов для электронных схем, при котором определяется последовательность входных воздействий (тестовых векторов), позволяющая делать заключение о корректности работы цифровой схемы.

Сформированные таким образом наборы тестовых векторов используются для отбраковки кристаллов после изготовления, а в некоторых случаях и для локализации неисправности [1]. Эффективность (покрытие) ATPG выражается в количестве протестированных дефектов, которые могут возникнуть в схеме, а также типом этих дефектов. Дефекты могут возникнуть на стадии производства кристалла. Модель дефекта — математическое описание того, как рассматриваемая неисправность изменяет поведение дефектной схемы по отношению корректной (исходной). Тест обнаруживает дефект, если при подаче на вход дефектной схемы тестовых воздействий имеются различия в значении хотя бы одного выходного сигнала по сравнению с исходной схемой. Процесс ATPG, направленный на обнаружение определённого дефекта, состоит из двух фаз: внесение ошибки и распространение ошибки. Внесение ошибки заключается в подаче такого входного тестового вектора, при котором поведение исходной и дефектной схем будут различны. Процесс распространения ошибки нужен для вывода искажённого в результате дефекта сигнала на выходные порты тестируемой логической схемы.

Несмотря на то, что на ранних этапах развития ATPG рассматривалось множество моделей дефектов [2], в последние десятилетия в ATPG используется модель константного логического значения (stuck-at fault), то есть модель «залипания» сигнала. В этой модели

предполагается, что некоторый проводник в схеме имеет постоянное значение «0» или «1» независимо от входных воздействий на схему.

С увеличением числа входных портов и количества внутренних состояний тестируемого устройства множество тестовых векторов, а значит, и время проверки, увеличиваются по экспоненциальному закону. Поэтому для сокращения времени подбора достаточного числа тестовых векторов применяют технику декомпозиции сложной логической схемы на несколько относительно небольших комбинационных блоков, хорошо поддающихся анализу. Каждый такой блок анализируется на предмет возможных дефектов. Далее формируется набор тестовых векторов, выявляющий каждый из этих дефектов. Поиск в большинстве случаев производится посредством компьютерного моделирования, при котором сравнивается поведение нормального и дефектного блока при подаче определённого входного воздействия. Если поведения различны, дефект считается определяемым, а соответствующее входное воздействие заносится в список тестовых векторов для анализируемого блока. Данный процесс заканчивается, когда будут подобраны тестовые векторы для всех возможных дефектов. Для больших схем рассмотренный процесс может потребовать значительных затрат временных и вычислительных ресурсов.

Предлагается оптимизация решения задачи АТРГ за счёт использования ПЛИС¹ для моделирования поведения тестируемого блока. Частоты функционирования современных микросхем ПЛИС (> 200 МГц) позволяют добиться 10–100-кратного выигрыша в скорости моделирования. Реальное повышение скорости зависит от тестируемой схемы, а также от используемого алгоритма программного моделирования. На рис. 1 приведена тестовая система. Она представляет собой управляющий компьютер, подключенный к модулю с ПЛИС. Управляющий компьютер производит выбор очередного тестируемого блока и типа дефекта, конфигурирует ПЛИС и подаёт последовательность входных воздействий. Результатом работы ПЛИС является ответ, подходит ли очередной тестовый вектор для выявления данного дефекта. При положительном ответе управляющий компьютер сохраняет тестовый вектор и производит переконфигурацию ПЛИС для определения следующего дефекта.

Рассмотренный метод позволяет значительно повысить скорость формирования тестовых векторов для задачи АТРГ при анализе больших логических схем. На данный момент в качестве ПЛИС удоб-

¹Программируемая логическая интегральная схема

нее всего выбрать микросхему фирмы Xilinx семейства Virtex5 [3], поскольку она обладает наиболее подходящей к данной задаче методикой переконфигурации отдельных областей в реальном времени.

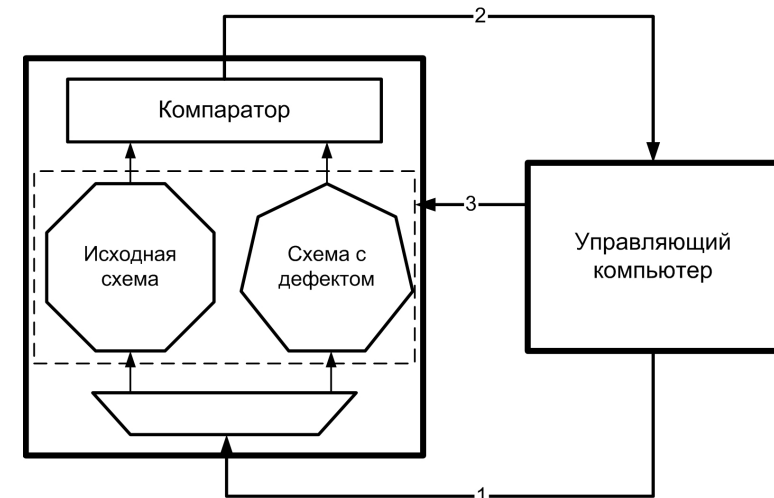


Рис. 1. Структурная схема системы поиска тестовых векторов; 1 — канал подачи входных воздействий, 2 — результат сравнения откликов схем на входное воздействие, 3 — канал переконфигурации ПЛИС

Литература

1. Lavagno, Martin, Scheffer. Electronic Design Automation For Integrated Circuits Handbook. — Ohio: ASM International 2004.
2. Brian Chess, Tracy Larrabee. Generating Test Patterns for Bridge Faults in CMOS ICs. — Santa Cruz: Department of Computer Engineering, University of California 1994.
3. Harsh Dhand, Neeraj Goel, Mukesh Agarwal, Kolin Paul. Partial and dynamic reconfiguration in Xilinx FPGAs. — a quantitative study. In Proc of VLSI Design And Test Symposium (VDAT 2005), Bangalore, India, August 2005.

УДК 004.2

Н.А. Щербина

shcherbina@mcst.ru

Московский физико-технический институт
(государственный университет)
ЗАО «Московский центр SPARC-технологий»

Разработка системного коммутатора для микропроцессора МЦСТ-4R

В настоящее время потребность в высокопроизводительных вычислительных комплексах привела к созданию нескольких классов многопроцессорных систем. Среди них можно выделить класс систем с неоднородным доступом к памяти (NUMA), где память физически распределена между процессорами, но логически общедоступна.

Вычислительные комплексы NUMA класса в основном строятся на базе систем на кристалле (System on Chip), в которых большинство системных контроллеров (контроллер оперативной памяти, контроллер удалённого доступа и т. д.) вместе с процессорами интегрируются в одном кристалле.

В настоящий момент компания ЗАО «МЦСТ» выполняет проект по созданию NUMA системы на базе процессора МЦСТ-4R. В нем параллельно с разработкой основных процессорных модулей проектируется подсистема памяти, включающая в себя оборудование распределенного чипсета (Chip Set). Оно обеспечивает аппаратную поддержку коммутационной среды, объединяющей отдельные процессоры. В данной работе рассматривается разработка системного коммутатора (СК), который является частью этой среды, отнесенной к одному процессору.

Одна из основных проблем создания распределенного чипсета заключается в тестировании на предмет логических ошибок. Поскольку верификация его модулей в составе всей системы возможна только на последних стадиях разработки rtl модели процессора, то изменение алгоритмов функционирования проектируемой аппаратуры в случае некорректной (либо неоптимальной в плане производительности) работы — очень трудоёмкий процесс. К этому можно добавить длительное время разбора ошибочных ситуаций, что, главным образом, является свойством процесса отладки оборудования в составе всей

системы. Поэтому не менее важной и трудоёмкой задачей, чем непосредственная разработка rtl модели коммутационного оборудования, включающего СК, является осуществление его автономного тестирования на наличие логических ошибок и на производительность.

Для тестирования производительности как всей коммутационной среды, так и СК, был использован метод, подробно описанный в [2] (Chapter23 Performance Analysis). Он основывается на генерации входного трафика, задаваемого некоторыми вероятностными параметрами. При этом информация, необходимая для измерений на выходах из СК, передается в полях пакетов, не используемых при штатной работе коммутационной среды. В результате доработки этого метода был получен автономный тест, производящий как измерения производительности коммутационной среды, так и выполняющий проверку на наличие логических ошибок. Он применим к любой коммутационной среде. Это позволяет активно применять его в процессе разработки коммутационного оборудования путём варьирования ключевых элементов, таких, как арбитры [1, 2], механизмы контроля потока данных (flow control [1, 2]) и наблюдения за изменениями в поведении системы (изменениями производительности).

В результате на данный момент разработана rtl модель СК, полностью удовлетворяющая всем функциональным требованиям. В них входит обеспечение обмена абонентов, находящихся на том же узле, что и коммутатор (L2-кэш, контроллер памяти и Ю-зонд), между собой и с абонентами, расположенными на удалённых узлах многопроцессорной конфигурации, через межсистемные линки.

При выполнении своих функций СК решает ряд задач. В первую очередь это получение пакетов от абонентов либо из системных линков; их десериализация (при необходимости); определение абонентов-получателей при помощи маршрутизирующей таблицы; коммутация пакетов к соответствующим выходным портам; их сериализация (при необходимости); отправка их получателю либо в один из системных линков.

К более сложным задачам можно отнести следующие:

- обеспечение обслуживания абонентов согласно политике приоритетов;
- обеспечение независимости пакетных потоков между различными парами абонентов (пакет из одного потока, по каким-либо причинам не получающий право на дальнейшее прохождение по коммутационной среде, не должен блокировать обработку пакетов из другого потока);

— максимально возможную загрузку пропускных способностей межсистемных линков при использовании уже разработанных для NUMA системы контроллеров межсистемного взаимодействия;

— обеспечение независимости потоков различных типов командных пакетов (запросы на различного рода чтения/записи, ответы на выполненные команд, снузирование и. т. д.), что осуществляется посредством механизма виртуальных каналов (virtual channel [3]).

Литература

1. *Dally W.J., Towles B.* Principles and practices of interconnection networks. — Morgan Kaufmann, 2004.
2. *Webber M.* Arbiters: design ideas and coding styles // SNUG Boston. — 2001.
3. *Dally W.J.* Virtual-channel flow control // IEEE Transactions on Parallel and Distributed Systems. — March 1992. — V. 3, N. 2. — P. 194–205.

Секция микропроцессорных технологий

УДК 004.45

А.Б. Беляев

belyaevab@gmail.com

Санкт-Петербургский государственный политехнический университет

Верификация системы поддержки транзакционной памяти для программирования многоядерных процессоров

Широкое внедрение многоядерных процессоров ставит проблему эффективного использования ресурсов параллельного компьютера. Одним из наиболее важных её аспектов является полное использование параллелизма с доступом параллельных процессов к общей памяти.

Классическим подходом к решению этой проблемы является синхронизация процессов, использующих разделяемые ресурсы. Для этого применяются различные примитивы синхронизации. Данное решение обладает рядом известных недостатков. В частности, неосторожное использование примитивов может приводить к взаимным блокировкам процессов.

Принципиально иной подход к параллельному программированию предоставляет транзакционная память — технология, гарантирующая полное отсутствие ошибок параллелизма при выполнении параллельных программ на многоядерных процессорах.

Однако алгоритм транзакционной памяти не совсем прост — он работает с параллельными процессами, поэтому в самом этом алгоритме могут быть ошибки. Следовательно, корректность самой системы поддержки транзакционной памяти должна быть формально

доказана. Ручное доказательство ничем не лучше ручной разработки систем, и оно так же подвержено ошибкам, как и ручное проектирование. Для автоматической верификации может быть применен метод model checking.

Формальная верификация на основе model checking состоит в исчерпывающей проверке всех возможных состояний модели системы на соответствие некоторому набору требований (спецификации), сформулированных на формальном языке. Как правило, системы верификации позволяют не только ответить на вопрос, удовлетворяет ли система предъявляемым требованиям, но и в случае отрицательного ответа получить ту последовательность действий, которая иллюстрирует невыполнение требований (так называемый контрпример, который позволяет найти ошибку).

Автором доклада был исследован алгоритм работы транзакционной памяти (window-based software transactional memory system), ручное доказательство которого было предложено на 10-й Международной конференции PaCT 2009 [1].

Основные результаты исследования.

1. Для алгоритма построена модель на языке PROMELA [2].
2. Исходя из общих требований к работе транзакционной памяти формально построена спецификация этого алгоритма на языке темпоральной логики линейного времени (LTL).
3. Проведена верификация модели алгоритма с помощью системы верификации Spin [2].
4. Проведен анализ контрпримера, автоматически найденного системой Spin при верификации модели, и выявлена причина некорректности алгоритма.
5. Проанализировано опубликованное в [1] ручное формальное доказательство правильности этого алгоритма и выявлена причина того, почему это доказательство не обнаружило ошибку.
6. Алгоритм window-based software transactional memory system изменен и верифицирован снова, в результате чего некорректных траекторий его работы не обнаружено. Модифицированный корректный алгоритм может быть рекомендован к реализации в многоядерных процессорах.

На рис. 1 представлен пример некорректного выполнения транзакций, управляемых рассматриваемым алгоритмом. Выполнение несериализуемо (результат не эквивалентен никакому последовательному выполнению транзакций), что противоречит одному из требований к алгоритму транзакционной памяти.

Последующий анализ алгоритма показал, что причина некорректности — использование атомарной инструкции Fetch&Increment (v) [1], увеличивающей значение переменной v и возвращающей её старое значение. При использовании аналогичной инструкции, возвращающей новое значение, некорректных поведений системы не обнаруживается.

Проведённое исследование, кроме всего прочего, иллюстрирует необходимость в использовании процедуры автоматической формальной верификации алгоритмов работы транзакционной памяти наряду с их ручным доказательством.

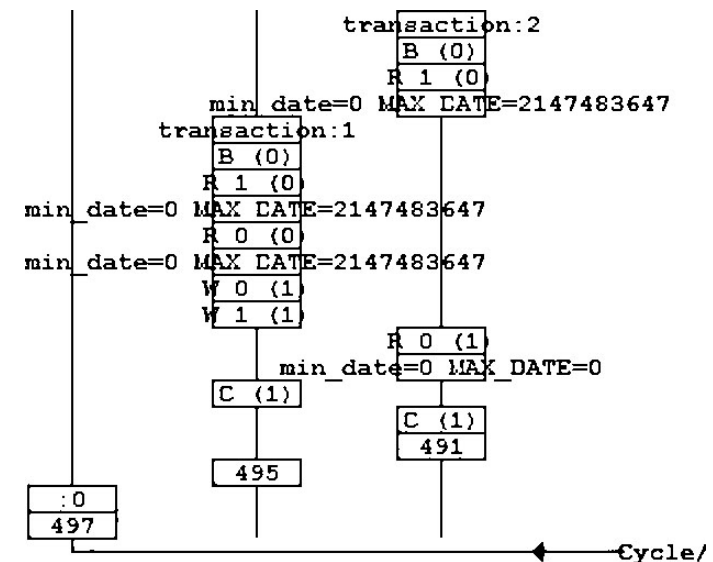


Рис. 1. Контрпример: R и W — события чтения и записи соответственно, число рядом — адрес, C — событие успешного завершения транзакции, число в скобках — значение внутренней переменной алгоритма

Литература

1. Imbs D., Raynal M. Software Transactional Memories: An Approach for Multicore Programming // PaCT 2009. LNCS. — 2009. — V. 5698. — P. 26–40.
2. Holzmann G. Spin Model Checker. The Primer and Reference Manual. — Addison-Wesley, 2003. — 608 p.

УДК 681.3

Е.А. Гаврин

eugene.gavrin@guap.ru

Санкт-Петербургский государственный университет

Программный симулятор параллельной вычислительной системы

С развитием технологий у разработчиков появилась возможность использовать для вычислений большее количество процессорных элементов. Такие комплексы, обладающие несколькими процессорными элементами, имеют огромный потенциал производительности и могут использоваться для решения гораздо более ресурсоемких задач по сравнению с классическими однопроцессорными системами.

На сегодняшний день многопроцессорные системы применяются для очень широкого круга задач, например, в научных вычислениях, различных пользовательских системах (персональные компьютеры, игровые приставки и мобильные системы) и т. д. Такие комплексы достаточно дорогостоящи и сложны в производстве, поэтому для того чтобы окупить финансовые и технологические затраты на производство подобных систем, постоянно требуется искать вычислительные алгоритмы, способные максимально эффективно их использовать.

Алгоритм программы, предназначенный для симуляции на вычислительной системе, описывается графом. Граф программы — это структурная модель программы, которая показывает связь между её элементами. Вершинами являются операторы и данные, дугами — направленные связи между операторами.

Для решения подобной задачи был разработан симулятор — программный комплекс, моделирующий работу алгоритма на распределенной вычислительной системе, предназначенный для симуляции работы параллельных алгоритмов. С его помощью должна осуществляться проверка работоспособности исследуемых и разрабатываемых алгоритмов без низкоуровневого моделирования.

При работе было принято решение применять систему, основанную на передаче сообщений. Основанием для подобного решения стало то, что данные системы лишены явных недостатков систем с общей памятью и в то же время позволяют организовывать многопроцессорные вычислительные системы с размерностью, которая многократно

превышает размерность систем с общей памятью. Кроме того, незначительные усовершенствования позволят с некоторой общностью реализовать такой популярный класс многопроцессорных систем, как системы-на-кристалле.

В разработанном симуляторе реализована возможность подключения пользовательского исходного кода, редактирования настроек симуляции и написания пользовательских дополнений. Кроме того, симулятор ориентирован на осуществление моделирования программной части, а аппаратной — лишь в степени достаточной для адекватной оценки работы алгоритмов. В то же время при создании была заложена возможность расширения функциональности в сторону более подробной симуляции аппаратной части платформы.

Работа симулятора была разделена на значимые составляющие. Такое разделение структуры программы потенциально позволяет организовать работу над проектом силами нескольких человек, тем самым разграничив зоны ответственности, что, например, позволит проводить тестирование над независимыми друг от друга частями. Кроме того, такая модульная структура даёт возможность программисту изменять интересующие его модули, добавляя или сокращая функционал, не затрагивая соседствующие модули всей оставшейся системы.

Программная часть симулятора состоит из реализации совокупности вычислительных задач (процессов). В рамках данной работы были внедрены различные виды процессов, каждая из которых обладают специализированными свойствами и характеристиками, достаточными для описания подавляющего большинства алгоритмов. У всех типов процессов имеются общие свойства, такие как: уникальный номер процесса, название процесса, а также время выполнения вычислительной задачи на заданном типе процессора. В том числе у процессов имеется набор входных и выходных портов, в которых происходит обмен данными с другими процессами. Процессы разделены по своим свойствам на несколько типов: терминальные операторы, управляющие операторы и распределенные данные.

В качестве исходных данных для работы симулятора примем, что в нашем распоряжении имеется платформа с полностью связанной топологией сети, где каждый процессорный элемент может напрямую принимать и получать данные от любого другого процессорного элемента. В качестве входных файлов следует использовать стандартизованное описание платформы и описание графа программы, содержащего все необходимые данные.

В качестве результатов работы симулятора генерируется набор выходных файлов, содержащих всю информацию, необходимую для отладки параллельных программ. В процессе работы симулятора создаётся три файла: специализированный дамп данных, предназначенный для отладки в специализированной программе, упрощённый лог работы и файл, дублирующий поток вывода на экран (рис. 1, 2).

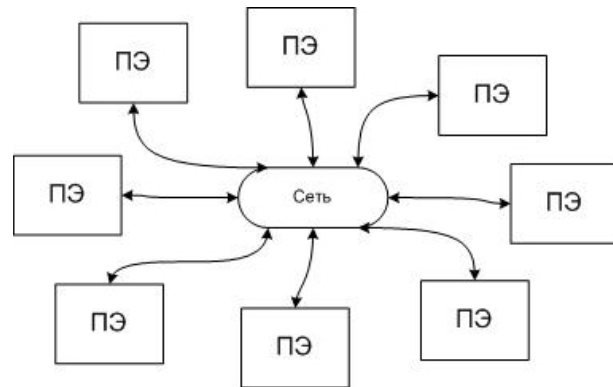


Рис. 1. Общая структура вычислительной сети

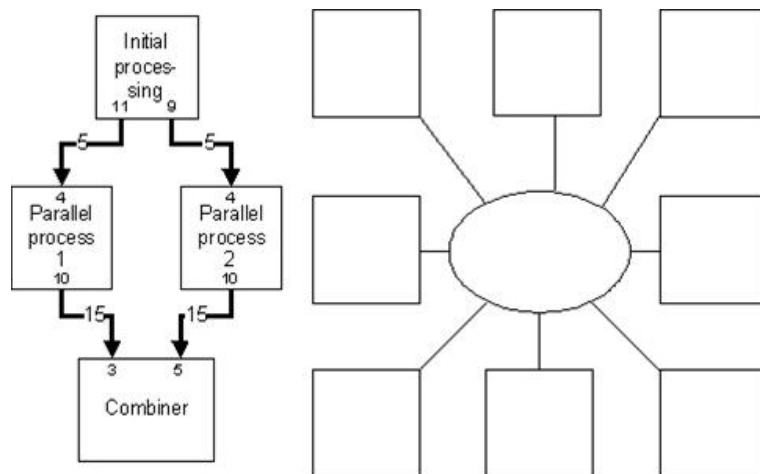


Рис. 2. Пример графа задач и платформы соответственно

Литература

1. *Almasi G.S., Gottlieb A.* Highly Parallel Computing. Benjamin-Cummings publishers, Redwood City, CA, 1989.
2. *Maslennikov O.* Systematic Generation of Executing Programs for Processor Elements in Parallel ASIC or FPGA-Based Systems and Their Transformation into VHDL-Descriptions of Processor Element Control Units». Lecture Notes in Computer Science, 2000. — 2328/2002: P. 272.
3. Architecture share for 06/2007. TOP500 Supercomputing Sites. Clusters make up 74. 60% of the machines on the list. Retrieved on November 7, 2007.

УДК 004.332.34

М.А. Дунаева

margr@rambler.ru

Samsung Electronics

Новый зарядовый усилитель считывания

Усилители считывания, принцип работы которых основан на улавливании разности напряжений в узлах усилителя (Voltage Sense Amplifiers), широко используются в цепях считывания оперативных запоминающих устройств (SRAM). Явным недостатком усилителей напряжения является зависимость их работоспособности от дисбаланса таких параметров парных транзисторов, как пороговое напряжение и длина канала. Альтернативой подобным усилителям считывания являются зарядовые усилители считывания (Charge Transfer Sense Amplifier). Далее будет описан новый усилитель считывания (Novel Charge Transfer Sense Amplifier), а также проведено сравнение с двумя известными усилителя считывания [1] и [2].

Новый зарядовый усилитель считывания (рис. 1) работает в три этапа. На первом этапе происходит предзаряд битовых линий bl и blb , линий данных dl и dlb . На втором этапе начинается чтение. Предположим, что читают 1 (высокий уровень), ток на dl — i_{dl} , ток на dlb — i_{dlb} , где $|i_{dl}| < |i_{dlb}|$. Устройства MP25 и MP26 работают как ёмкости, и они разряжаются токами i_{dl} и i_{dlb} . На третьем этапе на линии

read оставляют низкий уровень напряжения, линия pch находится на высоком уровне, на линию saen подают низкий уровень напряжения. Токи на линиях dl и dlb меняют направление и значения $|i_{dl}| > |i_{dlb}|$. Эта разница токов формирует заряд в узлах vb1 и vb2. Также она помогает «защелкнуть» верный результат благодаря разнице токов через устройства MP23 и MP24. Результат записывают с помощью nМОП MN15 и MN14. Важным фактом является то, что устройства MP25 и MP26 дольше, чем MP23 и MP24, остаются в режиме насыщения. Это значит, что ток через них не зависит от напряжения в узлах vb1 и vb2, что помогает в случае дисбаланса транзисторов защелки.

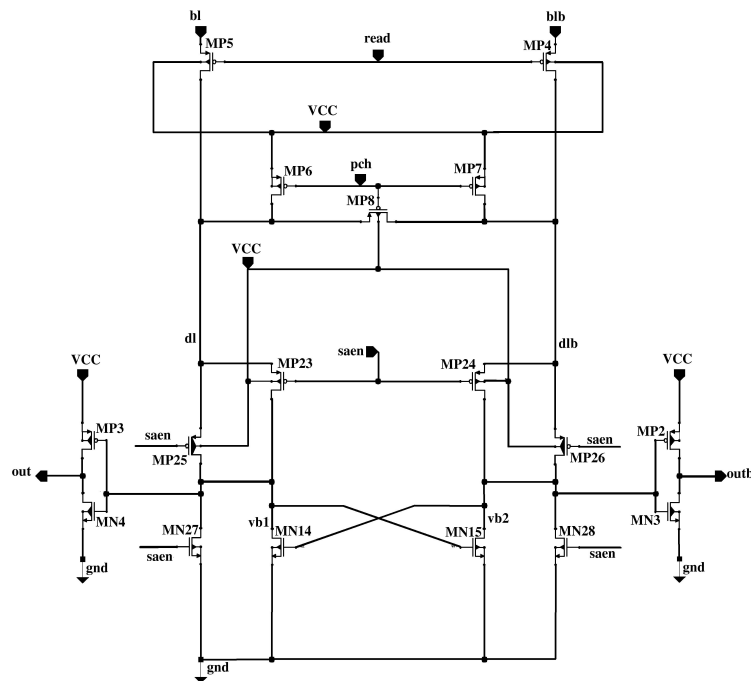


Рис. 1. Новый зарядовый усилитель считывания

LVT транзисторы (с низким пороговым напряжением) MP23 и MP24 увеличивают ток разряда MP25 и MP26. Фактически, в начале третьей стадии токи разряда MP25 и MP26 текут через MP23 и MP24. MP25 и MP26 открываются позже. В то же время вклад дисбаланса пороговых напряжений LVT транзисторов меньше.

Таким образом, если убрать HVT (с высоким пороговым напряжением) транзисторы MP25 и MP26, то это приведёт к большей чувствительности усилителя к дисбалансу в защелке. Если убрать LVT транзисторы MP23 и MP24, это приведёт к большей чувствительности усилителя к дисбалансу параметров HVT транзисторов MP25 и MP26.

Заметим, что не последнюю роль в работе нового зарядового усилителя считывания играет ширина затвора HVT транзисторов MP25 и MP26. Сразу после переключения saen в 0, ток i_{dl} может быть представлен в виде: $i_{dl} = i_0(e^{-|a| \cdot t} - e^{-|b| \cdot t})$, где $|a| < |b|$, минимум $|b|$ достигается при максимально мощных pМОП MP25 и MP26. Наиболее выраженный скачок тока i_{dl} будет при минимальном по модулю значении b , то есть в случае использования мощных транзисторов MP25 и MP26.

Результаты моделирования. По сравнению с усилителем [1] новый зарядовый усилитель считывания имеет более высокую скорость срабатывания. Например, реализованный на технологии 65 нм, он показывает задержку 43 пс выхода после включения сигнала sense enable. При этом усилитель считывания [1] показывает задержку 84 пс при тех же условиях (все три усилителя считывания оптимизировались по максимальному выходу годных). Разница процента выхода годных обоих усилителей считывания оказалась порядка погрешности.

Недостаток усилителя [2] заключается в сложности его конструкции, использовании дополнительного уровня напряжения (V_b), что увеличивает стоимость усилителя, а также в недостаточно высокой скорости срабатывания (50 пс против 43 пс при тех же условиях) и низком проценте выхода годных. Усилитель [2] показал 23% ошибок в результате считывания, тогда как новый усилитель считывания показывает 8% ошибок в результате считывания при $|V_{blb} - V_{bl}| = 50$ мВ. Также новый усилитель считывания оказался наименее чувствительным к разбросу ёмкостей битовых линий.

Литература

1. Michael Anthony Ang. Charge transfer sense amplifier // US Patent 5,668,756. — 1997.
2. Sinha M., Hsu S., Alvandpour A. [et al.]. Highperformance and low-voltage sense-amplifier techniques for sub-90nm SRAM // Proceedings of IEEE Int. SOC Conference. — 2003. — P. 113–116.

УДК 004.4'422

М.В. Золотухин^{1,2}, Д.Ю. Бабочкин²

zolutukhin@gmail.com, babokin@gmail.com

¹ Московский физико-технический институт
(государственный университет)² ЗАО «Интел А/О»

Преобразование графа потока управления в список инструкций

В современных оптимизирующих компиляторах основным представлением программы является граф потока управления — ГПУ ([1], с. 169–171). Каждая вершина этого графа — линейный участок, то есть участок программы, исполнение которого начинается с первой инструкции путём передачи на неё управления, и далее все инструкции участка исполняются ровно один раз до передачи управления следующему линейному участку. Таким образом, инструкции перехода могут находиться только в конце линейного участка, а метки — только в начале. ГПУ также имеет выделенный блок-вход, с которого начинается исполнение функции.

Однако изначально программа представляет из себя не граф, а последовательный набор инструкций — другой вариант внутреннего представления, наиболее характерный для ранних стадий компиляции и имеющий свои преимущества ([1], с. 70–71).

Часто возникает задача совершить перевод внутреннего представления из одного типа в другой, причём только в одном направлении такой перевод является однозначным. Перевод графа в список инструкций не является однозначным, и поэтому выбор оптимального варианта зависит от цели и критерия оптимальности, с которой производится данное преобразование. В данной работе рассматривается один из вариантов данного преобразования, целью которого является получение как можно более удобочитаемого текста программы (то есть списка инструкций). Такая задача возникает, в частности, при отладке программ — для лучшего визуального восприятия человеком больше подходит представление программы в виде текста, чем в виде графа.

Рассмотрим, какими свойствами должны обладать блоки и составленный из них список, чтобы при последовательной распечатке эле-

ментов данного списка получалась синтаксически корректная программа, ГПУ которой совпадал бы с оригинальным. За основу возьмем синтаксис языка С. Заметим, что в представлении в виде списка мы вынуждены вставлять инструкции goto для корректного отражения передачи управления в программе.

Будем считать, что:

— если у некоторого блока один последователь в ГПУ и его последняя инструкция при представлении в списке не является инструкцией перехода (goto), то в искомом списке этот последователь должен находиться непосредственно за данным блоком;

— если у некоторого блока один последователь в ГПУ и его последняя инструкция при представлении в списке является инструкцией перехода (goto), то в искомом списке этот последователь должен иметь соответствующую метку и может находиться в любом месте;

— если у некоторого блока два последователя в ГПУ и его последняя инструкция является инструкцией условного перехода (if), то в искомом списке должно находиться два подсписка блоков, каждый из которых в искомом списке должен быть окружен фигурными скобками, а первыми блоками данных подсписков должны быть последователи данного блока. Заметим, что один из списков может быть пустым;

— если у некоторого блока нет последователей (то есть он завершается инструкцией возврата из функции), он должен быть либо последним в списке, либо за ним должен идти блок, не имеющий предшественников или имеющий метку (его предшественник в этом случае может находиться в любом месте списка);

— в остальных случаях последней инструкцией блока должна быть инструкция перехода switch, все последователи которой должны иметь соответствующие метки, перечисленные в инструкции switch как возможные цели перехода. В искомом списке последователи могут находиться в произвольных местах.

Для произвольного ГПУ задача может быть решена следующим образом: начиная с блока-входа будем выполнять поиск в глубину ([1], с. 177–181). Если текущий блок уже был посещён ранее, то в список он не добавляется — добавляется лишь инструкция перехода на него. Таким образом будет построен искомый список для произвольного графа.

Однако такой способ даёт далеко не самое оптимальное решение с точки зрения читаемости кода — в этом легко убедиться, рассмотрев граф, соответствующий структуре if-then-else-joint, — данный алго-

ритм добавит одну дополнительную инструкцию перехода, хотя такой граф можно представить без её использования.

Существует другой способ решения задачи, однако применим он не для произвольных графов. В нём учитывается, что простейшими элементами программ на языке С (данная работа относится в первую очередь для компиляторов этого языка, но может быть адаптирована и для работы с многими другими языками) являются следующие структуры: if-then, if-then-else, while-do, switch. Поэтому графы большинства программ представляют собой комбинацию подграфов, представляющих такие структуры, причём каждая такая структура может быть «вершиной» более крупной структуры. Поскольку представить каждую из этих структур в виде списка не представляет труда, весь граф также может быть представлен в виде списка в случае, если с помощью таких трансформаций (замены таких структур одной вершиной) он упрощается до единственной вершины.

Чтобы объединить достоинства обоих методов, был предложен следующий алгоритм:

- 1) некоторым образом выделить подграф,
- 2) максимально упростить его вторым способом,
- 3) к упрощённому графу применить первый способ,
- 4) заменить данный подграф одной вершиной,
- 5) продолжить обработку графа.

После замены подграфа одной вершиной теряется информация о дугах, идущих в него (кроме дуг, входящих в первый блок), а также выходящих из него (кроме дуг, выходящих из последнего блока), поэтому если такие дуги есть, для них необходимо добавить инструкции переходов goto. Критерием оптимальности представления подграфа мы выбираем наименьшее количество добавленных инструкций перехода (goto).

Если в качестве подграфов для данного алгоритма использовать циклы, то число добавленных инструкций оказывается близким к минимальному. Действительно, циклы часто содержат единственный вход и один или несколько выходов, причём дополнительные выходы изначально реализуются с помощью конструкций, аналогичных goto (например, инструкций return или break). Данный алгоритм добавляет одну дополнительную инструкцию goto на каждый дополнительный выход из цикла в случае, если такая инструкция не присутствовала изначально.

Реализация описанного алгоритма в промышленном оптимизирующем компиляторе и тестирование на широком классе программ поз-

воляют утверждать, что предложенный метод хорошо решает поставленную задачу поиска удобочитаемого представления ГПУ.

Литература

1. *Muchnick S.S.* — Advanced Compiler Design & Implementation.
2. *Aho A., Sethi R., Ullman J.* — Compilers: Principles, Techniques, and Tools.

УДК 004.414.2

Н.Е. Косарев

kolya_kos@mail.ru

Московский физико-технический институт
(государственный университет)
ЗАО «Интел А/О»

Исследование подходов для получения распределения потерь производительности микропроцессора с векторным счётчиком инструкций

За последние два десятилетия суперскалярные архитектуры и архитектуры со сверхдлинным командным словом за счёт использования параллелизма на уровне инструкций практически достигли своего потенциального предела производительности. Тенденции в разработке микропроцессоров изменяются в сторону распределённых архитектур, работающих в тесном взаимодействии с компилятором [1, 2]. Одной из возможных реализаций архитектуры данного типа является микропроцессор с векторным счётчиком инструкций. Важнейшей задачей при разработке микропроцессора является анализ производительности, который в значительной степени основан на анализе распределения потерь тактов, получаемого на этапе имитационного моделирования.

В настоящей работе были исследованы два подхода для получения распределения потерь производительности микропроцессора с векторным счётчиком инструкций на критическом пути исполнения программы. Порядок исполнения инструкций программы может

быть представлен в виде графа микроархитектурных зависимостей, узлами которого являются микроархитектурные события, такие, как отправка запроса в память, окончание декодирования инструкции и т. д. Направленные и взвешенные дуги графа представляют собой микроархитектурные зависимости, а именно зависимости по данным, управлению и ресурсам. Вес дуги отражает время в тактах, необходимое для разрешения соответствующей зависимости. Критическим путём данного графа является самый длинный взвешенный путь, начинающийся с выборки первой инструкции и оканчивающийся записью данных в память или выполнением последней инструкции, и определяющий время исполнения программы.

Первый подход основан на моделировании динамического графа микроархитектурных зависимостей. Для получения распределения потерь производительности на критическом пути отслеживается порядок разрешения микроархитектурных зависимостей и подсчитываются веса всех последних приходящих дуг (*last-arriving edges*) графа [3]. Подсчёт весов выполняется в специальных маркерах, перемещающихся по графу вдоль последних приходящих дуг. Каждый маркер представляет собой набор счётчиков, в котором каждый счётчик соответствует определённому типу зависимости. Маркер последней выполненной инструкции содержит искомое распределение потерь производительности.

Второй подход получения распределения потерь производительности основан на исследовании большого числа идеализированных моделей микропроцессора [4]. При каждом запуске модели намеренно устраняется одна из причин потерь, например, обнуляется задержка обращения за данными в память и замеряется время, необходимое для выполнения программы. Разница во времени выполнения идеализированной и стандартной модели микропроцессора определяет искомую компоненту распределения потерь.

Оба подхода имеют как преимущества, так и недостатки. Первый подход достаточно трудоемок в реализации, но позволяет получить распределение потерь с любой необходимой степенью детализации за единственный запуск имитационной модели, что особенно важно для тестов, состоящих из сотен миллионов инструкций. Второй подход, напротив, прост в реализации, но требует, чтобы имитационная модель позволяла запрограммировать все необходимые идеализации, что не всегда возможно. При его применении для получения распределения потерь из n компонент необходимо $n + 1$ запусков имитационной модели. К тому же подход не учитывает возможного наложения

микроархитектурных потерь, и поэтому к ошибке, соответствующей количеству наложенных тактов, причём эта ошибка растёт с увеличением n .

В результате выполнения работы для микропроцессора с векторным счётчиком инструкций разработана детальная модель микроархитектурного критического пути, запрограммирован алгоритм подсчета потерь в маркерах, а также выполнена его интеграция в потактовую имитационную модель микропроцессора. В рамках второго подхода реализован набор программ (сценариев) для полностью автоматического выполнения многократных запусков идеализированной имитационной модели с последующей обработкой полученных замеров производительности и представлением распределения потерь в виде гистограммы. В результате применения исследуемых подходов было получено распределение потерь производительности (рис. 1) для группы из нескольких наиболее часто используемых участков (*hotspots*) контрольных тестов, включая тесты из наборов SPEC92–2000. Группа состоит из скалярных и из потоковых контрольных тестов. Из полученных распределений видно, что потоковые тесты (*livermore*, *seria*) в значительной степени зависят от пропускной способности подсистемы памяти и длительности обращения к ней. В двух тестах (*go*, *scalc*) проявляются потери, обусловленные межкластерной передачей данных. Остальные тесты ограничены пропускной способностью функциональных блоков микропроцессора.

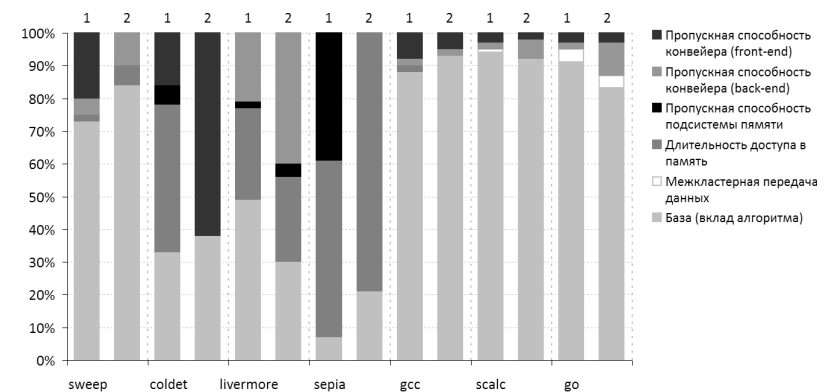


Рис. 1. Распределение потерь производительности микропроцессора, полученное в результате моделирования графа микроархитектурных зависимостей (1) и путём многократных запусков идеализированной имитационной модели (2)

Литература

1. *Kim H.S., Smith J.E.* An Instruction Set and Microarchitecture for Instruction Level Distributed Processing // The 29th International Symposium on Computer Architecture. — 2002.
2. *Caulfield I.* Complexity-effective superscalar embedded processors using instruction-level distributed processing // Technical Report UCAM-CL-TR-707, Computer Laboratory, University of Cambridge. — 2007.
3. *Fields B., Rubin S., Bodik R.* Focusing processor policies via critical-path prediction. In 28th International Symposium on Computer Architecture. — 2001.
4. *Fields B., Bodik R., Hill M. [et al.].* Using Interaction Costs for Microarchitectural Bottleneck Analysis. In proceedings of the 36th International Symposium on Microarchitecture. — 2003.

УДК 519.6 + 681.3.012

Д.В. Макошенко

makden@mail.ru

ЗАО «Интел А/О»

Южный федеральный университет
Московский физико-технический институт
(государственный университет)

Назначение переменных на регистры с помощью новых алгоритмов раскраски графа

В данной работе рассматривается задача назначения переменных на физические регистры, важная при трансляции программы с языка высокого уровня в машинные коды. Если физических регистров для хранения переменных не хватает, то требуется добавление дорогостоящих инструкций доступа в память. Поэтому актуальна задача оптимального распределения переменных на физические регистры.

Для поиска распределения переменных на физические регистры введём понятие графа несовместимости (ГН) [1]. ГН — неориентированный граф, описывающий зависимости между временами жизни переменных. Вершины графа — переменные. Ребро соединяет две вершины, если времена жизни соответствующих вершинам переменных пересекаются.

Пусть число регистров, доступных в данной архитектуре, равно N . Будем отождествлять каждую краску с одним физическим регистром.

Если ГН удалось раскрасить N красками, то распределение физических регистров завершено. В самом деле, никакие две переменные в этом случае не будут назначены на один физический регистр — условие корректности соблюдено. В противном случае нужно трансформировать программу, чтобы уменьшить хроматическое число соответствующего ей ГН. Проблема трансформации программы выходит за рамки данной работы.

Если граф G полный, то его единственная возможная раскраска — каждая вершина получает свой собственный цвет. Если граф G не полный, воспользуемся методом из [2]. Выберем две несмежные вершины u и v . Обозначим $(G; \langle u-v \rangle)$ исходный граф, в котором вершины u и v соединены ребром. Обозначим $(G; \langle u=v \rangle)$ исходный граф, в котором вершины u и v стянуты в одну.

Рассмотрим множество $M(G)$ всех раскрасок графа несовместимости G . Тогда $M(G)$ представимо в виде объединения двух непересекающихся множеств:

$$M(G) = M((G; \langle u-v \rangle)) \cup M((G; \langle u=v \rangle)). \quad (1)$$

Пользуясь соотношением (1), рассмотрим построение дерева C перебора всех раскрасок графа G (рис. 1). Граф G соответствует корню дерева перебора C . На рис. 2 и рис. 3 изображены графы соответствующие остальным вершинам дерева C . На рис. 3 все графы полные, у вершин, которым они соответствуют, нет потомков. Обозначим краски номерами. Сопоставим вершинам краски в порядке возрастания нумерации вершин, тогда $M(G) = \{\{1, 2, 3, 4\}, \{1, 2, 3, 1\}, \{1, 2, 2, 3\}, \{1, 2, 2, 1\}\}$.

Пусть C — некоторое поддерево дерева C всех вариантов раскраски, причём C содержит корень C . Введём параметр LeftChilds — максимальное количество левых сыновей на любом пути от корня к листу в дереве C . Обозначим через $\text{Built}(n)$ количество левых сыновей на пути от корня дерева C к вершине n .

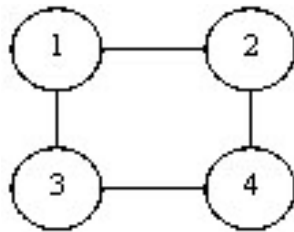


Рис. 1. Исходный граф G с пронумерованными вершинами

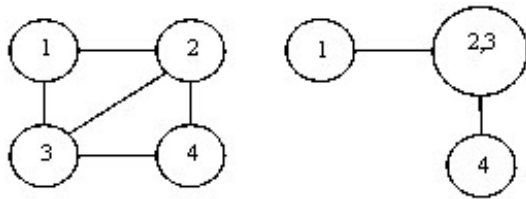


Рис. 2. Графы, соответствующие потомкам K и L корня C

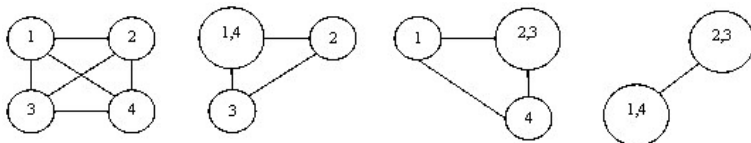


Рис. 3. Графы, соответствующие слева направо: левому сыну K , правому сыну K , левому сыну L , правому сыну L

Параметрический алгоритм строит дерево C . Чем выше значение $LeftChilds$, тем большее количество шагов будет выполнять параметрический алгоритм. Таким образом, в зависимости от требований к качеству раскраски алгоритм позволяет тратить больше времени для получения лучшего результата.

Рассмотрим метод построения сыновей для такой вершины n дерева C , что соответствующий ей граф $G(n)$ не является полным. Выбирается пара вершин u и v , принадлежащих множеству вершин $G(n)$, не соединённых ребром. Правый сын вершины n — это вершина, которой соответствует граф $(G(n); < u = v >)$. Левый сын вершины n , которому соответствует граф $(G(n); < u-v >)$, может существовать в дереве C , а может и не существовать.

Обозначим множество $\Gamma(v)$ как множество всех вершин, смежных вершине v . Определим предикат $I(u, v)$:

$$I(u, v) = (\Gamma(v) \subset \Gamma(u) == TRUE) \vee (\Gamma(u) \subset \Gamma(v) == TRUE). \quad (2)$$

Левый сын вершины n существует, если выполняются два условия: предикат $(Built(n) < LeftChilds)$ истинен; для пары вершин u и v предикат (2) ложен.

Лемма. Рассмотрим граф G . Возьмём две его произвольные вершины u и v . Пусть выполняется $\Gamma(v) \subset \Gamma(u)$ или $\Gamma(u) \subset \Gamma(v)$. Тогда хроматическое число графа $(G; < u = v >)$ не больше, чем хроматическое число графа G .

Согласно лемме для пары вершин u и v истинность предиката (2) означает, что $(G(n); < u = v >)$ имеет не худшее хроматическое число, чем $G(n)$. В этом случае построение левого сына излишне, поскольку его хроматическое число не лучше хроматического числа правого сына.

Можно показать, что при некотором, возможно, даже нулевом значении параметра $LeftChilds$ данный алгоритм может находить наилучшую раскраску.

Литература

1. Briggs P. Register allocation via graph coloring. A thesis submitted in partial fulfillment of the requirement for the Degree Doctor of Philosophy. — Houston, Texas, USA, 1992.
2. Зыков А.А. Основы теории графов. — Ростов-на-Дону: ЗАО Издательское предприятие «Вузовская книга», 2004. — 382 с.

УДК 621.3

Д.О. Нурждин¹, С.С. Ткачёв^{1,2}

nuzhdin@phystech.edu, stevens_1@mail.ru

¹ Московский физико-технический институт
(государственный университет)² Институт прикладной математики им. М.В. Келдыша РАН

Определение ориентации КА на солнце посредством солнечных батарей с использованием микроконтроллера Atmega8535

Одним из необходимых условий управления космическим аппаратом (далее КА) является определение ориентации до начала управления. Существует множество систем ориентации в пространстве, но целью данной работы было создание системы определения угловой ориентации по солнцу. Получая сведения об угле направления КА на солнце и привлекая дополнительные сведения из других систем навигации, мы можем определить ориентацию КА в пространстве.

Идея проекта состоит в определении угла направления на солнце посредством солнечных батарей. Батареи располагаются по корпусу аппарата, по кругу. Измеряя напряжения питания, выдаваемые батареями, и сравнивая их между собой, мы можем определить направление на солнце. Измерять напряжение и определять угол предполагается с помощью микроконтроллера.

Задачей данного проекта являлось собрать работающую схему, которая будет реализовывать описанную выше идею.

Для решения поставленной задачи был выбран 8-битный AVR микроконтроллер фирмы Atmel, ATmega8535. Используя встроенный аналого-цифровой преобразователь (АЦП), контроллер опрашивает солнечные батареи и преобразовывает аналоговое напряжение в его цифровое значение. Для определения напряжения на солнечных батареях используются ноги порта А согласно спецификации. Далее, производя некоторые вычисления, микроконтроллер определяет угол направления, выдавая нам требуемый результат.

При разработке схемы при использовании АЦП возникли проблемы с помехами и соответственно с точностью преобразования, схема

была модифицирована и добавлены дополнительные элементы для исключения влияния помех.

Для правильной работы и взаимодействия микроконтроллера с окружающими устройствами, пишется программа, использующая бесплатное программное обеспечение AStudio от фирмы Atmel. Для программирования микроконтроллера был применён программатор, подключение к микроконтроллеру которого осуществляется по схеме, приведённой на рис. 1.

В результате проделанной работы получено достаточно простое и дешёвое устройство, позволяющее ориентироваться на источник света.

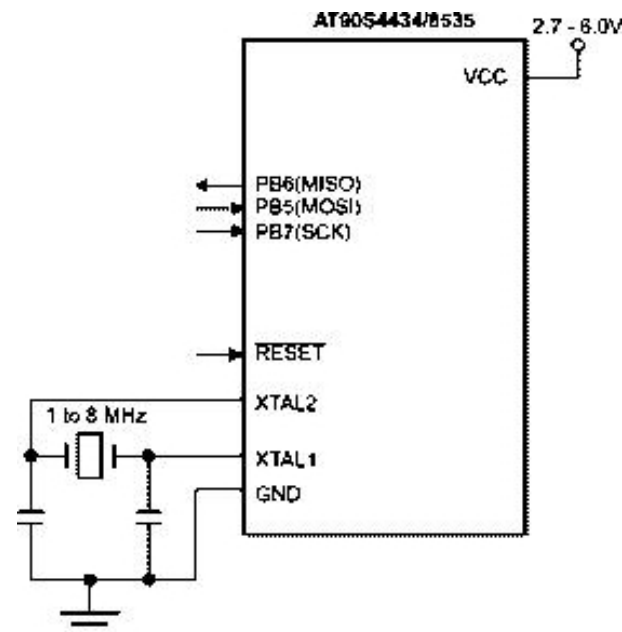


Рис. 1. Подключение микроконтроллера при программировании

Литература

1. Донов Г.И. Применение микроконтроллеров. — М.: Физтехполиграф, 2007. — 160 с.

УДК 519.876.5

Г.С. Речистов

grigory.rechistov@intel.com

Московский физико-технический институт
(государственный университет)
ЗАО «Интел А/О»

Методика бинарной трансляции с динамической модификацией кода в Intel Platform Simulator

Данная работа описывает систему модификации исполняемого кода, разработанную автором в процессе работы над эмулятором ЦПУ, входящем в состав полноплатформенного эмулятора Intel Platform Simulator [1].

В работе используются следующие определения:

— платформа — набор некоторых устройств, таких, как ЦПУ, память, жёсткий диск, ПЗУ с BIOS, сетевая карта, монитор, связанных друг с другом и представляющих некоторую ЭВМ;

— хозяин (host) — компьютер, на котором запущена программа эмуляции;

— гость (guest) — моделируемая платформа.

В данной работе рассматривается эмуляция только компонента, отвечающего за центральный процессор. Для того чтобы обеспечить большую скорость симулирования гостевого кода, используется вариант бинарной трансляции [2]. Бинарная трансляция широко используется во множестве приложений для таких задач, как обеспечение запуска немодифицированных приложений на «неродных» архитектурах (Apple Rosetta, DEC FX! 32, Intel IA-32 Execution Layer, Эльбрус Lintel), динамическая оптимизация (HP DynamoRIO), инструментация кода для исследовательских целей (PIN [3]).

В данном эмуляторе ЦПУ машинный код гостя покомандно транслируется в одну или более инструкций хозяина, образуя страницы кода, который затем выполняется. Тем самым достигается высокая скорость симуляции для часто повторяемых участков (например, циклов).

Фрагменты кода, соответствующие отдельным гостевым инструкциям, получили названия капсул. Одна капсула может содержать десятки ассемблерных инструкций, оперирующих с состоянием симулируемого процессора, производящих всевозможные проверки времени исполнения и т. п. операции. Общий вид заголовка капсулы с именем CAPS_NAME выглядит так:

$$CAPS_NAME(param1,param2, \dots),$$

где param1, param2, ... — это параметры капсулы, могущие быть именами регистров или целыми числами. Так, чтобы создать код для эмуляции инструкции `addl $ 0 × 1234, % ecx`, вызывается капсула `ADDL (0 × 1234, ECX)`.

Задача, которую необходимо решить, — каким образом передавать указанные аргументы ассемблерным инструкциям. Старый подход, использовавшийся в предыдущих версиях эмулятора, позволял создавать капсулы с не более чем двумя параметрами-регистрами, а для параметров-целых было необходимо вручную находить смещения байтов, которые необходимо модифицировать от начала капсулы. При таком подходе было неудобно вносить изменения.

Метод, предложенный в этой работе, подразумевает автоматическое распознавание типа и кодировки инструкции, требующей передачи параметров, вычисление необходимых смещений целых, а также снятия ограничения на число параметров-регистров капсулы.

Описание даётся на псевдоассемблере следующего вида:

```
// имя капсулы, в скобках — тип и имя её параметров
CAPS_NAME (R32 @reg1, U8 @imm1) {
  movl $44, @reg1 // параметр используется как приёмник
  xchgl @reg1, %eax // используется обыкновенный регистр EAX
}
```

Далее оно подаётся на скрипт, который разбирает его и выдаёт два файла: чистый ассемблер, в котором параметры заменены не имеющими смысла значениями, и процедуру на языке C, которая копирует машинный код капсулы (полученный из ассемблера) и модифицирует его, настраивая параметры.

В процессе создания и отладки вышеописанной системы возникли проблемы различного характера. Очень большая их доля связана непосредственно со спецификой набора инструкций архитектуры Intel [4], развивавшейся в течение сорока лет и претерпевшей за это время множество модификаций, породивших постепенное запутывание набора инструкций, так как новые режимы адресации памяти,

размеры операндов и т. п. не согласованы со старыми вариантами и приводят к неоднозначности кодировок.

Некоторые примеры встретившихся сложностей.

Команды набора инструкций SSE2, а также других расширений имеют обратный порядок операндов в сравнении со «старейшими» командами, такими как MOV, ADD и т. д.

Адресация памяти одним базовым регистром может быть записана на минимум тремя способами.

Инструкции с одинаковым мнемоническим обозначением имеют различные типы и количество операндов, что усложняет разбор псевдоассемблера. Особо выделяется инструкция IMUL, которая может иметь количество операндов от одного до трёх.

Подобные проблемы разрешались с использованием максимально гибких (из допустимых) нотаций для каждого встреченного способа адресации операндов. Одинаковая длина машинного кода в байтах до и после его модификации было основным требованием на генерируемый ассемблер.

Воплощение описанных выше принципов позволило добиться гибкости записи капсул и избавиться от недостатков, присущих предыдущей их версии. Кроме того, был упразднён сложный многоступенчатый процесс сборки капсул благодаря объединению процесса в один скрипт.

Разработанный подход динамической генерации кода планируется использовать в последующих версиях эмулятора ЦПУ Intel Platform Simulator.

Литература

1. Uhlig R., Fishtein R., Gershon O. [et al.]. SoftSDV: A Presilicon Software Development Environment for the IA-64 Architecture // Intel Technology Journal. — 1999. — P. 112. — 126.
2. Sites R.L., Chernoff A., Kirk M.B. [et al.]. Binary translation // Communications of the ACM. — 1993. — V. 36, N. 2. — P. 69–81.
3. Luk C., Cohn R., Muth R. [et al.]. Pin: building customized program analysis tools with dynamic instrumentation // Proceedings of the 2005 ACM SIGPLAN Conference on Programming Language Design and Implementation (Chicago, IL, USA, June 12. — 15, 2005). — P. 190–200.
4. Intel Corporation. Intel®64 and IA-32 Architectures Software Developer's Manual. Vol. 2A, 2B.

УДК 519.6

Е.Е. Хатъко

evgeniy.khatko@gmail.com

Московский физико-технический институт
(государственный университет)
ООО «Даймонд Кволити»

Один из подходов к анализу системы тестирования сложных программных комплексов

В данной статье вводится следующее понятие: мультизадачный пользовательский комплекс (МПК) — это система аппаратного и программного обеспечения, совмещающая функции смартфона и коммуникатора. Существует широкий класс операционных систем, которые поддерживаются МП комплексами. Примерами таких систем могут быть: Symbian OS (Nokia, Samsung, Sony Ericson), Windows Mobile (HTC, T-mobile, Samsung), Palm OS (Palm), Android (Samsung, LG, Palm).

Проведём оценку эффективности автоматизации тестирования МПК. Тестирование МПК — это для большинства проектов ручное тестирование методом чёрного ящика. Эффективность оценивается покрытием тестами формальных требований спецификации продукта. Поэтому нужно автоматизировать некоторый набор регрессионных тестов, который покрывает требования к наиболее часто используемому функционалу. При написании и выполнении ручных и автоматизированных тестов существуют свои особенности, которые мы будем использовать в дальнейшей оценке. Разработку тестов и проведение тестирования будем оценивать в человеко-часах (чел.-час).

Рассмотрим некоторый проект по разработке ПО для МПК. Пусть $\{\alpha_1, \dots, \alpha_N\}$ — некоторый набор тестов, покрывающих формальные требования спецификации, $\{\alpha'_1, \dots, \alpha'_K\}$ — некоторое подмножество этих тестов, которое покрывает наиболее часто используемый функционал. Будем считать время разработки i ручного теста равным t_i^{MD} . Пусть t_i — время вылонения i -го теста, а t_i^{AD} — время анализа результата прохождения автоматизированного i -го теста, $N = |\{\alpha_1, \dots, \alpha_N\}|$, $K = |\{\alpha'_1, \dots, \alpha'_K\}|$. Пусть t_i^A — время разра-

ботки i -го автоматизированного теста. Пусть также D_M — время разработки и выполнения всех тестов без процессов автоматизации, $D_M = \sum_{i=1}^N t_i^{MD} + \sum_{i=1}^N t_i$ (чел.-час). При автоматизации K тестов их время разработки и выполнения равно $\sum_{i=1}^K t_i^A + \sum_{i=1}^K t_i^{AD}$. После их автоматизации ещё останется $N - K$ ручных тестов, время работы с которыми составит $\sum_{i=K+1}^N t_i^{MD} + \sum_{i=K+1}^N t_i$, тогда $D_A = \sum_{i=K+1}^N t_i^{MD} + \sum_{i=K+1}^N t_i + \sum_{i=1}^K t_i^A + \sum_{i=1}^K t_i^{AD}$, где D_A — время разработки и выполнения всех тестов с условием автоматизации тестов $\{\alpha'_1, \dots, \alpha'_K\}$ (здесь время выполнения автоматизированных тестов считаем равным 0). Рассмотрим величину эффективности автоматизации тестирования как отношение времени ручного тестирования и времени тестирования с частичной автоматизацией:

$$Eff = \frac{D_M}{D_A} = \frac{\sum_{i=1}^N t_i^{MD} + \sum_{i=1}^N t_i}{\sum_{i=K+1}^N t_i^{MD} + \sum_{i=K+1}^N t_i + \sum_{i=1}^K t_i^A + \sum_{i=1}^K t_i^{AD}}.$$

После выполнения A раундов регрессионного тестирования эта формула примет вид

$$Eff(A) = \frac{\sum_{i=1}^N t_i^{MD} + A \sum_{i=1}^N t_i}{\sum_{i=K+1}^N t_i^{MD} + A \sum_{i=K+1}^N t_i + \sum_{i=1}^K t_i^A + \sum_{i=1}^K t_i^{AD}}.$$

Ручной тест — это некоторый набор инструкций, которые нужно выполнить с тестируемой программой, значит, t_i^{MD} — время затраченное на написание инструкций. Пусть t_i — время выполнения i -го ручного теста, оно складывается из времени прочтения и выполнения всех инструкций тестировщиком. Для оценки все тесты можно считать одинаковыми, то есть $\exists t = t_i = t_i^{MD}$. Время интерпретации результатов выполнения каждого автоматизированного теста не зависит от самого теста, обозначим его t^{AD} . Тогда формула примет вид

$$\begin{aligned} Eff(A) &= \frac{\sum_{i=1}^N t_i^{MD} + A \sum_{i=1}^N t_i}{\sum_{i=K+1}^N t_i^{MD} + A \sum_{i=K+1}^N t_i + \sum_{i=1}^K t_i^A + \sum_{i=1}^K t_i^{AD}} = \\ &= \frac{Nt(A+1)}{(N-K)t + A(N-K)t + K\overline{t^A} + Kt^{AD}} = \\ &= \frac{Nt(A+1)}{t(A+1)(N-K) + K\overline{t^A} + Kt^{AD}}, \text{ где } \overline{t^A} \text{ — среднее время разработки одно-} \\ &\text{го автоматизированного теста. Пусть } \mu = \frac{t^{AD}}{t} \text{ } (\mu > 0). \text{ Для оценки} \\ &\text{примем } K = N/2. \end{aligned}$$

Для исследования были взяты следующие программные продукты, позволяющие автоматизировать процессы тестирования ПО для МПК.

Run-on-Device (Jamo solutions) — запись тестовых скриптов производится на МПК, поэтому можно было бы считать $t^A = t$, но инструмент поддерживает лишь 2 из 4 ОС, поэтому при разработке автоматизированных тестов нужно учитывать время портирования (переноса) тестов на другую платформу, то есть $t^A = k_1 t$, где k_1 — количество поддерживаемых платформ (рассмотрим $k_1 = 3$), $Eff(A) = \frac{2(A+1)}{A+4+\mu}$.

DeviceAnywhere Automation™ for Smartphones (DeviceAnywhere) — требует интеграции со специальным средством тестирования Desktop-приложений. Это увеличивает время разработки автоматизированных тестов в k_2 раз, где k_2 — коэффициент, характеризующий степень интеграции и квалификацию тестировщика. В результате исследования выявлено, что $k_2 \approx 2$, тогда

$$Eff(A) = \frac{2(A+1)}{A+3+\mu}.$$

UserEmulator (Symbian foundation) — аналогичен Run-on-Device, но не позволяет задавать контрольные точки проверки результатов выполнения скрипта (checkpoints), поэтому $\mu \approx 1$.

Проведя статистику по более чем 10 законченным проектам разработки ПО для МПК, было выявлено, что самые распространенные значения A — это 2 и 3. Рассмотрим графики зависимости $Eff(\mu)$ при данных значениях A .

Итак, продукты Run-on-Device и UserEmulator не подходят для тестирования при $A = 2$, а DeviceAnywhere не является особо эффективным. При $A = 3$ значительной эффективности можно достичь при $\mu \rightarrow 0$ и $K \rightarrow N$, то есть уменьшая время анализа результатов и увеличивая количество автоматизированных тестов. Таким образом, нужно разработать методологию, которая позволила бы

- сместить акценты в процессе автоматизации с «интеграции в существующие средства тестирования Desktop-приложений» в сторону автоматизации на целевом устройстве (МПК);
- разрабатывать тестовые скрипты платформонезависимо;
- уменьшить время проверки результатов тестирования (рис. 1).

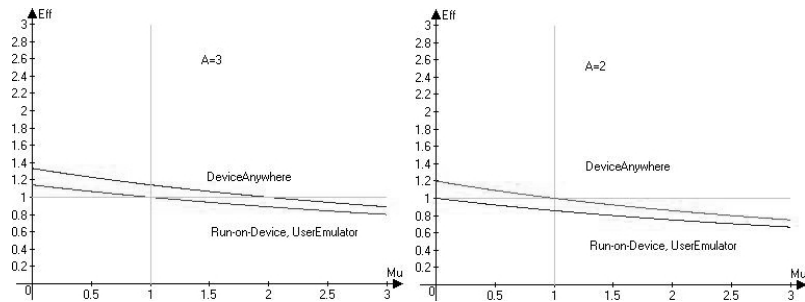


Рис. 1. Эффективность автоматизации тестирования

Литература

1. Канер К., Фолк Д., Нгуен Е.К. Тестирование программного обеспечения. Фундаментальные концепции менеджмента бизнес-приложений.
2. Синицын С.В., Налютин Н.Ю. Верификация программного обеспечения.

Секция проблем передачи и обработки информации

УДК 519.254

М.Г. Беляев^{1,2}, Е.В. Бурнаев²

mikhail.belyaev@phystech.edu, burnaev@iitp.ru

¹ Московский физико-технический институт
(государственный университет)

² Институт проблем передачи информации им. А.А. Харкевича РАН

Обобщающая способность и регуляризация в задаче аппроксимации многомерной зависимости

Задача аппроксимации многомерной функции $y = f(X)$, $X \in R^n$, $y \in R^1$ имеет важное прикладное значение — с помощью полученных моделей удобно исследовать зависимости в целом ряде прикладных задач.

Качество полученной аппроксимации характеризуется значением функции ошибки. Однако даже при малых значениях этой величины на обучающей выборке модель может иметь плохую обобщающую способность, и предсказания для новых точек будут сильно отличаться от истинных значений. Для устранения этого эффекта используется регуляризация, то есть штраф на сложность модели.

Существуют различные способы введения регуляризации. В работе [1] предлагается подбирать параметр регуляризации вручную. В [2] для нахождения оптимального параметра используется разложение в ряд Тейлора асимптотического приближения функции ошибки. Работы [3, 4] предлагают многоэтапное обучение нейронной сети. Первый этап заключается в стандартном обучении сети до момента остановки. Затем вычисляется оптимальное значение параметра, и сеть снова обучается. Исследования по адаптивному подбору параметра регуляризации, приведённые в работе [5], представляют интерес, но требуют

значительного повышения вычислительной сложности процесса обучения.

Предлагаемый алгоритм адаптивного подбора параметра регуляризации улучшает обобщающую способность модели и не требует трудоемких вычислений.

Рассматривается аппроксимация с помощью разложения искомой функции по адаптивному базису параметрических функций $\{\varphi_j(X)\}_{j=1}^p$: $\hat{f}(X) = \sum_{j=0}^p V_j * \varphi_j(X)$, где параметр p определяет сложность модели, задавая количество базисных функций. Для оценки параметров базисных функций используется один из методов градиентного спуска [6], а коэффициенты разложения V_j вычисляются с помощью метода наименьших квадратов.

Оценка оптимальных коэффициентов разложения имеет вид

$$\hat{V} = (\Phi^T \Phi)^{-1} \Phi^T y,$$

где Φ — матрица значений базисных функций в точках выборки, $\Phi_{j,i} = \varphi_j(X^i)$. Применение формул линейной регрессии предполагает, что столбцы матрицы регрессоров Φ линейно независимы. Однако для хорошей обобщающей способности аппроксиматора существенно выполнение более сильного условия: матрица $\Phi^T \Phi$ должна быть хорошо обусловлена. В противном случае возникает эффект переобучения, и аппроксимация становится несостоятельной. Для устранения этого нежелательного эффекта применяется ридж-регрессия [7], в которой используются измененные формулы для подсчета коэффициентов регрессии: $\tilde{V} = (\Phi^T \Phi + \lambda I)^{-1} \Phi^T Y$, где $\lambda > 0$, а I — единичная матрица, что делает оценки более устойчивыми.

Ключевым вопросом является выбор значения параметра λ . Классические подходы, изложенные в [8], не учитывают итеративный характер обучения, поэтому не подходят для решения проблемы обобщающей способности аппроксиматора.

По аналогии с изменением параметров базисных функций предлагается обновлять значение λ на каждой итерации, выбирая направление изменения исходя из текущей обусловленности матрицы: $\Phi^T \Phi + \lambda I$. В качестве меры обусловленности алгоритм использует число обусловленности матрицы, равное отношению максимального собственного числа матрицы к минимальному. Эта величина для хорошо обусловленной матрицы принимает значения порядка 1 и стремится к бесконечности в обратном случае.

Для вычисления нового значения λ алгоритм использует значение псевдоградиента, которое равно 1, если число обусловленности мало

и -1 в обратном случае. Обновление λ и подбор шага происходят с помощью метода градиентного спуска [6]. Таким образом, если на текущей итерации матрица регрессоров $\Phi^T \Phi + \lambda I$ плохо обусловлена, то значение параметра регуляризации увеличивается и уменьшается в случае хорошей обусловленности.

Предлагаемый алгоритм был протестирован на искусственных и реальных данных. В качестве первой выборки использовались значения функции $y = \sin\left(\frac{1}{x^3}\right)$. На этих данных предлагаемый алгоритм регуляризации снизил значение функции ошибки и улучшил качество аппроксимации.

Реальная выборка представляет собой аэродинамические данные, характеризующие прочность структурной единицы фюзеляжа самолёта как функцию от геометрических размеров этой единицы и действующих на неё сил [9]. Использование регуляризации позволило существенно улучшить обобщающую способность и получить модель, гораздо лучше описывающую сложную физическую систему.

Литература

1. *Sjzberg J., Ljung L.* Overtraining, regularization and searching for a minimum, with application to neural networks // International Journal of Control. — 1995. — V. 62, N. 6. — P. 1391–1407.
2. *Wand M.P.* On the optimal amount of smoothing in penalized spline regression // Biometrika. — 1999. — V. 86, N. 4. — P. 936–940.
3. *Larsen J., Hansen L.K., Svager C.* [et al.]. Design and Regularization of Neural Networks: the optimal use of a validation set // Proceedings of the IEEE Workshop on Neural Networks for Signal Processing. — 1996. — V. 4. — P. 62–71.
4. *Larsen J., Svager C., Andersen L.N.* [et al.]. Adaptive Regularization in Neural Network Modeling // Neural Networks: Tricks of the Trade. — 1995. — V. 1524. — P. 113–132.
5. *Chen D., Hagan M.T.* Optimal use of regularization and cross-validation in neural network modeling // Neural Networks. — 1999. — V. 2. — P. 1275–1280.
6. *Riedmiller M., Braun H.* A direct adaptive method for faster backpropagation learning: The RPROP algorithm // In Proceedings of the IEEE International Conference on Neural Networks. — 1993. — V. 1. — P. 586–591.
7. *Hoerl A.E.* Application of ridge analysis to regression problems // Chemical Engineering Progress. — 1962. — V. 58, N. 1. — P. 54–59.

8. Демиденко Е.З. Линейная и нелинейная регрессии. — М: Финансы и статистика, 1981. — 302 с.

9. Burnaev E.V., Grihon S. Construction of the metamodels in support of stiffened panel optimization // Proceedings of the conference MMR 2009. — Mathematical Methods in Reliability. — 2009. — P. 124–128.

УДК 519.248

Е.В. Бурнаев¹, П.В. Приходько²

burnaev@iitp.ru, prihodkop@live.ru

¹ Институт проблем передачи информации им. А.А. Харкевича РАН

² Московский физико-технический институт
(государственный университет)

Об одной методике бустинга для решения задач регрессии

Зачастую в задачах регрессии оказывается эффективным использование ансамблей регрессоров вместо одиночных регрессоров. Беггинг (bagging) [1, 2] и бустинг (boosting) [3, 4] являются наиболее распространенными подходами к построению таких ансамблей.

Идея беггинга состоит в том, что регрессоры независимо и параллельно обучаются на различных (возможно, частично перекрывающихся) подмножествах обучающего множества. Итоговое предсказание беггинга является некоторым усреднением предсказаний всех регрессоров ансамбля.

Идея бустинга заключается в том, что на каждой итерации бустинга приближается какая-то функция от точки обучающей выборки и предсказаний предыдущих регрессоров [4].

В работе предлагается новый метод обучения ансамблей регрессоров, разработанный для увеличения точности итоговой регрессионной модели, и его модификация, уменьшающая время обучения ансамбля регрессоров ценой небольшой потери точности.

Для начала дадим общую постановку задачи регрессии. Задача приближения многомерной зависимости $f(X)$ заключается в том, чтобы, используя выборку точек $\Sigma = \{y_i^j X_i^j\}_{i=1}^N$, $y_i = f(X_i)$, построить

многомерную зависимость $y = \tilde{f}(X)$, $X \in R^M$, $y \in R^1$, для которой некоторая функция $D(f, \tilde{f} | \Sigma) \leq \varepsilon$, где ε — заранее заданное число. В качестве функции ошибки часто используют функцию вида $D(f, \tilde{f} | \Sigma) = \Sigma_{\{y, X\} \in \Sigma} (y - \tilde{f}(X))^2$.

Под построением ансамбля регрессионных моделей для решения задачи регрессии понимается случай, когда $y = \tilde{f}(X)$ строится как функционал $\tilde{f}(X) = F(f_1(X), \dots, f_N(X))$, где $f_i(X), i = 1, \dots, N$ являются регрессионными моделями некоторого класса.

Идея предлагаемого метода состоит в том, чтобы сохранить форму предсказания всего ансамбля в виде усреднения предсказаний всех входящих в него регрессоров (и таким образом уменьшить влияние случайных факторов на итоговое предсказание) и одновременно последовательно обучать регрессоры не на первоначальной функции, а на некоторой разности от неё и предсказаний предыдущих регрессоров. То есть B -й регрессор ансамбля $f_B(X)$ приближает не функцию $f(X)$, а её модификацию следующего вида: $y^B(X) = B \cdot y - (B - 1) \cdot F_{B-1}(X)$, где $F_B(X)$ — выход ансамбля, равный среднему от всех регрессоров ансамбля.

Возможна также модификация предложенного метода, направленная на увеличение скорости обучения ансамбля. В ней делается допущение о том, что для аппроксимируемой зависимости существует достаточная для нас точность приближения. Исходя из этого предположения, мы можем, подсчитав на текущей итерации ошибки приближения на точках обучающего множества, на следующей итерации приближать новой регрессионной моделью ансамбля только те точки, ошибка приближения которых больше допустимой. Эксперименты показывают, что использование данной модификации позволяет получить значительный выигрыш во времени при построении ансамбля регрессоров, при этом потери в точности ансамбля либо малы, либо вовсе отсутствуют.

Эксперименты на реальных и искусственных данных показывают, что предлагаемый метод бустинга имеет большую точность и меньшее время обучения по сравнению с классическими методиками бустинга и беггинга.

Литература

1. Breiman L. Bagging predictors // Machine Learning. — 1996. — V. 24, N. 2. — P. 123–140.

2. *Bruhlmann P., Yu B.* Explaining Bagging // Technical Report, Statistics Department, University of California at Berkeley. — 2000.

3. *Drucker H.* Improving Regressors using Boosting Techniques // Proceedings of the Fourteenth International Conference on Machine Learning. — 1997. — P. 107–115.

4. *Friedman J., Hastie T., Tibshirani R.* Additive logistic regression: a statistical view of boosting // The Annals of Statistics. — 2001. — V. 28. — P. 337–407.

УДК 621.396.626

А.В. Ксендзов

xendzov@yandex.ru

Рязанский государственный радиотехнический университет

Влияние корреляции замираний сигналов в многоантенной системе на число эффективных степеней свободы

Применение в системах связи для передачи и приёма сигналов многоантенных систем (МАС) с ортогональным кодированием на передающей стороне и оптимальной обработкой на приёмной (использование технологии ММО) является эффективным методом улучшения качества связи. Показано, что за счёт применения МАС при приеме сигнала уменьшается негативное влияние на сигнал замираний, что проявляется в уменьшении вероятности ошибки при фиксированном последетекторном отношении сигнал–шум (ОСШ) или позволяет снизить мощность на передатчике при фиксированном уровне вероятности ошибки [1]. Однако эффективность применения МАС существенно зависит от глубины замираний и их корреляции в парах антенн. В связи с этим возникает задача оценивания границ изменения эффективности МАС в условиях коррелированных замираний.

Общепринятым критерием надёжности систем связи является зависимость вероятности ошибочного приёма бита цифрового сигнала P_b от последетекторного ОСШ q_b , что позволяет сравнивать системы с различными видами модуляции и количеством антенн в МАС.

К примеру, вероятность ошибки для канала с одиночной приёмной антенной и без замираний вычисляется как [2]:

$$P_b = 0,5 \left(1 - \sqrt{\frac{\bar{q}_b}{1 + \bar{q}_b}} \right), \quad (1)$$

а вероятность ошибки для того же канала рэлеевскими замираниями без замираний определяется по формуле

$$P_b = Q \left(\sqrt{2q_b} \right),$$

где $Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-\frac{t^2}{2}} dt$, \bar{q}_b — среднее последетекторное ОСШ.

При использовании на приёмной стороне МАС с числом антенн N с последующей обработкой сигнала по методу минимума среднеквадратичной ошибки (МСКО) [3] последетекторное ОСШ является арифметической суммой ОСШ каждого приемного канала. Если замирания в канале отсутствуют, вероятность ошибки для N -элементной МАС определяется по формуле (1). При рэлеевских замираниях сигнала последетекторное ОСШ \bar{q}_b распределено по закону хи-квадрат с $2N$ степенями свободы, и выражение для вычисления вероятности ошибки имеет вид

$$P_b = \left(\frac{1 - \sqrt{\frac{\bar{q}_b}{1 + \bar{q}_b}}}{2} \right)^N \sum_{i=1}^N C_i^{N-1+i} \left(\frac{1 + \sqrt{\frac{\bar{q}_b}{1 + \bar{q}_b}}}{2} \right)^i. \quad (2)$$

Заметим, что аналогичным выражением определяется вероятность ошибки для канала с одиночной антенной и замираниями Накагами с глубиной $m = N$, причём при $m \rightarrow \infty$ формула (2) обращается в (1) для канала без замираний. Введём понятие числа эффективных степеней свободы (ЭСС) z как глубину независимых замираний Накагами в эквивалентном канале с одиночной антенной, обеспечивающих ту же вероятность ошибки, что и рассматриваемый канал с МАС. Очевидно, что при независимых рэлеевских замираниях число ЭСС равно числу антенн МАС, $z = N$. В случае, если в канале действуют независимые замирания Накагами–Райса с глубиной флюктуаций m и райсовским фактором луча прямой видимости k , число ЭСС вычисляется по приближенной формуле

$$z = Nmk.$$

При применении на передающей стороне МАС с числом антенн M и ортогонального кодирования число ЭСС определяется как

$$z = MNmk.$$

Показано [1], что идеально дружные замирания в канале с МАС на приёмной стороне полностью исключают эффективность МАС, обеспечивая для данного канала такую же вероятность ошибки при фиксированном \bar{q}_b , как и для канала с одиночной антенной, то есть для дружных замираний имеем число ЭСС $z = 1$. Обобщая сказанное для случая коррелированных замираний, делаем вывод, что число ЭСС при изменении степени корреляции замираний в передающей и приёмной МАС может меняться в пределах

$$1 \leq z \leq MNmk.$$

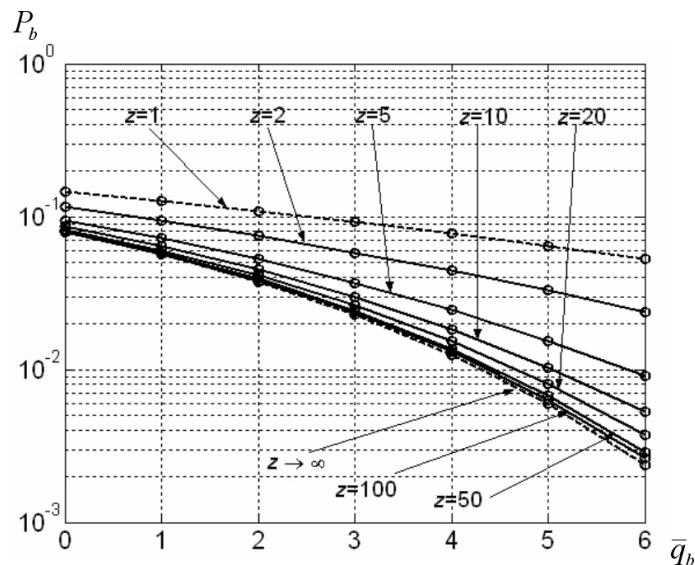


Рис. 1

В заключение отметим, что одним из эффективных способов декорреляции замираний, позволяющим повысить число ЭСС и снизить тем самым вероятность ошибки, является оптимизация пространственной структуры МАС [1], которая заключается в расположении элементов МАС согласованно с текущей пространственно-корреляционной обстановкой (рис. 1).

Литература

1. Паршин Ю.Н., Ксендзов А.В. Влияние пространственной корреляции на эффективность оптимизации пространственной структуры многоантенной системы при разнесенном приеме // Вестник Рязанской государственной радиотехнической академии. — 2006. — № 19. — С. 54–62.
2. Прохис Д. Цифровая связь. / Под ред. Д.Д. Кловского.; [пер. с англ.]. — М.: Радио и связь, 2000. — 800 с.
3. Монзинго Р., Миллер Т. Адаптивные антенные решётки. Введение в теорию. / пер. с англ. — М.: Радио и связь, 1986. — 448 с.

УДК 577.213

В.А. Любецкий, А.В. Селиверстов

lyubetsk@iitp.ru, slvstv@iitp.ru

Институт проблем передачи информации им. А.А. Харкевича РАН

Прямые повторы в некодирующих областях хлоропластов у семенных растений

В настоящее время нет общей модели эволюции некодирующих областей генома. Модель независимых однонуклеотидных замен и вставок-удалений не позволяет объяснить часто наблюдающиеся в хлоропластах вставки и делеции большой длины. В настоящей работе продемонстрировано возникновение длинного повтора участка, существовавшего у общего предка всех видов семейства сосновых. Близкие результаты наблюдаются и в семействе крестоцветных, хотя наблюдаемые там повторы перекрываются однонуклеотидными заменами и вставками-делециями, что обусловлено большим филогенетическим расстоянием между видами. Семейства сосновых и крестоцветных выбраны для исследования из-за большого количества видов, для которых секвенированы геномы хлоропластов.

В семействе сосновых перед геномом *psbM* часто возникают эволюционно независимые трехкратные прямые повторы длиной 11 нуклеотидов. При этом положение и состав повторов отличаются.

У *Pinus koraiensis* повторы отсутствуют, у *Pinus thunbergii* есть повтор слова ATGAGATAAAA, который отсутствует у остальных видов. У *Keteleeria davidiana* есть повтор слова TTCCATAGATA с одним отличием в третьей копии, который отсутствует у обеих сосен и расположен дальше от начала гена, чем сосновый повтор. Можно думать, что это место является удобным для возникновения повторов, которые возникают в случайном месте независимо от состава повторяемого участка. У *Cryptomeria japonica* повторы перед psbM отсутствуют, что подтверждает возникновение (а не потерю) повторов в семействе сосновых.

В семействе крестоцветных большое число длинных (с периодом от 7 до 15 нуклеотидов) tandemных повторов найдено в некодирующих белки участках из локусов, содержащих следующие опероны: *trnSGR-atpA*, *trnL-trnT*, *ndhHAIGE* и *ycf1*. Такие повторы найдены у следующих видов крестоцветных: *Aethionema cordifolium*: *trnSGR-atpA*; *Aethionema grandiflorum*: *trnSGR-atpA*, *ndhHAIGE*; *Arabidopsis thaliana*: *trnSGR-atpA*; *Arabis hirsuta*: *trnL-trnT*; *Barbarea verna*: *trnSGR-atpA*, *ycf1*; *Capsella bursa-pastoris*: *trnL-trnT*; *Draba nemorosa*: *trnSGR-atpA*, *ndhHAIGE*, *ycf1*; *Lepidium virginicum*: *trnSGR-atpA*, *trnL-trnT*; *Lobularia maritima*: *ndhHAIGE*; *Nasturtium officinale*: *trnSGR-atpA*, *ndhHAIGE*. Для *Crucihimalaya wallichii* и *Olimarabidopsis pumila* указанные повторы не найдены.

Авторы благодарны Е.А. Лысенко за обсуждение результатов.

Работа выполнена при частичной поддержке Международного научно-технического центра (проект 3807).

УДК 519.217.8

А.Б. Миллер

amiller@iitp.ru

Институт проблем передачи информации им. А.А. Харкевича РАН

Динамическое управление доступом к ресурсам и скоростью обслуживания при активных пользователях

Динамическое управление доступом предполагает, что вероятность отклонения заявок зависит от длины очереди роутера и величин потоков входящих заявок. Подход, основанный на теории управления марковскими цепями, использует описание управляемой цепи в терминах стохастических дифференциальных уравнений [3] и позволяет сформулировать задачу оптимизации входящего потока как задачу оптимального управления [4]. В данный момент растёт интерес к механизмам активного управления очередью, которые используют, в частности, и возможность изменения скорости обслуживания [1]. Такая постановка задачи укладывается в схему управления марковской цепью и может быть решена методами теории оптимального управления [4].

Рассматривается система массового обслуживания, управляемая посредством ограничения входящего потока, который представляет собой считающий процесс со случайной величиной интенсивности $\lambda \geq 0$, зависящей от временных предпочтений пользователей и текущего состояния очереди. Число заявок в системе ограничено некоторой константой $M < \infty$. Общее число состояний системы есть $M + 1$. Интенсивность обслуживания есть управляемая величина $\mu \in [\mu, \bar{\mu}]$, где $\mu > 0$. Управление $u(t) \in [0, 1]$ есть вероятность принять заявку в момент времени $t \in [0, T]$. Таким образом, часть приходящих заявок может быть отклонена. Входной поток формируется совокупностью активных пользователей, выбирающих интенсивность генерации заявок таким образом, чтобы максимизировать собственную функцию полезности [2]. Используется динамическая модель предоставления ресурсов к обслуживанию, в которой поток на входе роутера зависит от текущего значения вероятности доступа $P(t, M(t-))$, где $M(t)$ —

число заявок. Полагаем, что роутер использует управление марковского типа и имеет информацию о

функциях полезности пользователей. N пользователей используют функции полезности [6]:

$$f_i(v, t) = -\frac{a_i(t)}{v} - \lambda_0 v - vP(t, M(t-)), i = 1, \dots, N$$

и в каждый момент времени имеют информацию о текущем значении $P(t, M(t-))$, устанавливаемом роутером. Максимизируя свою функцию полезности, i -й пользователь устанавливает оптимальное значение интенсивности потока, равное

$$u_i^{opt}(t, P(t, M(t-))) = \operatorname{argmax}_{0 < v < \infty} f_i(v, t) = \sqrt{\frac{a_i(t)}{\lambda_0 + P(t, M(t-))}}.$$

Общее значение интенсивности управляемого входного потока, как функция от вероятности доступа и соответственно состояния очереди, равно

$$\lambda(t, P) = U(t, P) = \sum_{i=1}^N u_i^{opt}(t, P) = \frac{\sum_{i=1}^N \sqrt{a_i(t)}}{\sqrt{\lambda_0 + P}} = \frac{C(t)}{\sqrt{\lambda_0 + P}},$$

где $C(t) = \sum_{i=1}^N \sqrt{a_i(t)}$ — величина, которая характеризует всю совокупность пользователей. Зависимость $C = C(t)$ позволяет учесть временную зависимость потребности в ресурсах.

Оптимизационная задача состоит в минимизации некоторой функции стоимости состояний и управлений марковской цепи. Эта функция учитывает среднюю длину очереди, которая связана со средним временем обслуживания и/или ценой отклоненных заявок, так как они должны либо повторно встать в очередь, либо выбрать другой сервис-центр, также со стоимостью обслуживания. Более того, в случае фиксированного промежутка времени конечное состояние марковской цепи также очень важно, как и в случае, когда необходимо обеспечить «рассасывание» очереди, то есть разрешить проблему перегрузки.

Мы рассматриваем взвешенную сумму критериев качества, соответственно среднее время обслуживания, среднее число принятых заявок и стоимость обслуживания равно

$$J = k_1 J_1 + k_2 J_2 + k_3 J_3 =$$

$$= k_1 E \left\{ \int_0^T \frac{M(s)}{\mu(s)} ds \right\} + k_2 E \left\{ \int_0^T \frac{C(P(\tau) - 1)}{\sqrt{\lambda_0 + P(\tau)}} I\{M(\tau) < M\} d\tau \right\} +$$

$$+ k_3 \int_0^T \mu(\tau) I\{M(\tau) > 0\} d\tau \rightarrow \min,$$

где $k_1 > 0, k_2 > 0, k_3 > 0$.

Выбор соотношения между k_1, k_2, k_3 позволяет решать задачи оптимального управления с ограничениями, то есть когда одни из критериев минимизируется при ограничении, наложенном сверху на другие критерии, или даже в случае, когда таких критериев много. В этом случае мы получаем многокритериальную задачу оптимизации, и решение подобных задач в теории управляемых марковских цепей является предметом современных исследований [5].

Для нахождения оптимальных управлений $u(t)$ и $\mu(t)$ разработана программа моделирования на Maple 12. Для проведения моделирования задаются следующие параметры и терминальные условия:

$$\mu = 3, \bar{\mu} = 6,$$

$$c = 2 + 1,5 \cos(2(2 - t)),$$

$$k_1 = 1, k_2 = 2, k_3 = 1,5,$$

Средняя вероятность отклонения заявок в зависимости от состояния буфера практически совпадает с принятой в протоколе TCP RED.

На рис. 1 показана интенсивность входного потока на интервале времени $[0, 10]$, ограниченная вероятностью доступа, для состояния системы с 4 заявками в буфере.

В работе исследована стохастическая модель управления потоками в Интернет, основанная на использовании теории управляемых марковских цепей. Она учитывает активное поведение пользователей и позволяет получить оптимальное управление доступом и скоростью обслуживания с учётом естественных критериев, характеризующих функционирование роутер-провайдер. Модель основана на методах теории активных систем и учитывает функции полезности пользователей и различные критерии качества, характеризующие функционирование системы пользователи-роутер при нестационарных внешних условиях. Данная модель может служить основой для анализа более сложных систем, связанных не только с управлением потоками

данных, но и с функционированием иерархических систем, возникающих в ситуации клиенты–система обслуживания (склады, снабжение ограниченными ресурсами).

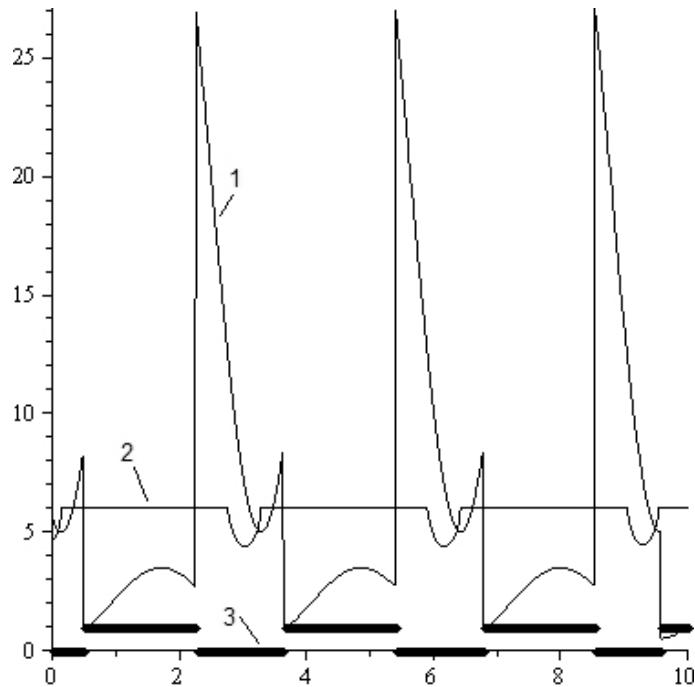


Рис. 1. Состояние с четырьмя заявками в буфере: (1) — входной поток, (2) — интенсивность обслуживания, (3) — вероятность отклонения заявок

Литература

1. Васенин В.А., Симонова Г.А. Математические модели управления трафиком в Интернете. Новые подходы, основанные на TCP/AQM механизмах // Автоматика и телемеханика. — 2005. — № 8. — С. 94–107.
2. Бурков В.Н. Основы математической теории активных систем. — М.: Наука. — 1977. — 255 с.
3. Elliott R.J., Aggoun L., Moore J.B. Hidden Markov Models. Estimation and Control. — New York: Springer Verlag, 1995. — 361 с.

4. Miller B.M. Optimization of queuing systems via stochastic control // Automatica (Journal of IFAC). — 2009. — V. 45, N. 6. — P. 1423–1430.
5. Pinovskiy A.B. Bicriteria optimization of a queue with a controlled input stream // Queueing Systems: Theory and Applications. — 2004. — V. 48. — P. 159–184.
6. Миллер А.Б. Динамическое управление доступом при активных пользователях // Информационные процессы. — 2009. — Т. 9, № 1. — С. 5–6.

УДК 621.39

О.Н. Плаксина

plaksina.olga@gmail.com

Российский университет дружбы народов

О двух системах массового обслуживания с «прозрачными» заявками и их применении к анализу сетей мультивещания

Услуга мультивещания занимает фиксированную ёмкость звена сети в независимости от числа обслуживаемых пользователей, что позволяет значительно экономить сетевые ресурсы. Мультивещание (multicasting) является одной из составляющих пакетной услуги TriplePlay, включающей передачу данных, речи и видео, а один из наиболее актуальных примеров использования технологии — предоставление услуг вещательного телевидения по IP-сетям (IPTV). Кроме того, мультивещание применяется в ряде других приложений, в частности, при организации видеоконференций, дистанционного обучения, рассылке корпоративной информации, реализации сетевых игр и т. д.

Рассматриваются два сценария предоставления услуги мультивещания. В первом случае (дисциплина обслуживания Π_1) принципиальную роль играет пользователь, активизирующий услугу, который определяет начало и окончание мультивещательной сессии. Остальные пользователи могут присоединиться к активизированной услуге,

однако завершение сессий всех пользователей происходит одновременно в момент завершения сессии инициатора услуги. Такой сценарий может применяться в сетевых играх или видеоконференциях. Разработке соответствующих математических моделей посвящены работы [1–2]. Во втором случае (дисциплина Π_2) пользователи могут присоединяться к серверу мультивещания и отключаться в любой момент времени. Это наиболее общий сценарий, используемый, например, для IPTV, частные случаи исследованы в [3]. В отличие от [1–3] в исследуемых ниже моделях рассматривается функция распределения (ФР) длительности обслуживания общего вида, а искомыми характеристиками являются среднее число обслуживаемых пользователей и их среднее время пребывания в системе.

Функционирование звена сети, предоставляющего $N \leq \infty$ однородным пользователям одну услугу мультивещания, описывается с помощью системы массового обслуживания (СМО) с «прозрачными» заявками $M_{N,\Lambda} | G_\mu | 1 | 0 | \Pi_i$, $i \in \{1, 2\}$, где Λ — суммарная интенсивность запросов от всех источников (пользователей), μ — интенсивность обслуживания заявки-инициатора услуги для дисциплины Π_1 (длительности обслуживания остальных сообщений равны остаточному времени обслуживания заявки-инициатора) и всех заявок для дисциплины Π_2 . Исследуются случайные процессы (СП) $\xi^{(i)}(t)$, $i \in \{1, 2\}$ числа заявок в системе для дисциплин Π_1 и Π_2 соответственно, которые проиллюстрированы на рис. 1.

Для СМО с дисциплиной Π_1 и неограниченным числом источников получены стационарные вероятности p_k того, что в системе находится k заявок:

$$p_0 = \frac{1}{1 + \rho}; \quad p_k = \frac{1}{1 + \rho} B_{k-1}, \quad k \geq 1,$$

где $B_k = 1 - \sum_{i=1}^k \int_0^\infty \frac{(\Lambda t)^i}{i!} e^{-\Lambda t} dB(t)$, $B(x)$ — ФР длительности обслуживания, а $\rho = \Lambda/\mu$.

На основе производящей функции $P(z) = \frac{1-z\beta(\Lambda-\Lambda z)}{(1-z)(1+\rho)}$ несложно найти среднее значение и дисперсию СП $\xi^{(1)}(t)$:

$$M\xi^{(1)} = \frac{\Lambda^2 b^{(2)} - 2}{2(1 + \rho)} + 1,$$

$$D\xi^{(1)} = \frac{2\Lambda^3 b^{(3)} + 9\Lambda^2 b^{(2)} + 6\rho}{6(1 + \rho)} -$$

$$- \frac{\Lambda^2 (b^{(2)})^2 + 4\Lambda^2 \rho b^{(2)} + 4\rho^2}{4(1 + \rho)^2},$$

где $b^{(2)}$ и $b^{(3)}$ — второй и третий начальные моменты ФР $B(x)$.

Показано, что для СМО с дисциплиной Π_2 распределение числа заявок в системе инвариантно относительно ФР длительности обслуживания. В случае $N < \infty$ получаем

$$p_0 = \frac{1}{(1 + \rho/N)^N}; \quad p_k = \frac{C_N^k (\rho/N)^k}{(1 + \rho/N)^N}, \quad k = \overline{1, N},$$

$$M\xi^{(2)} = \frac{\rho}{1 + \rho/N}; \quad D\xi^{(2)} = \frac{\rho}{(1 + \rho/N)^2}.$$

С помощью предельного перехода находим распределение и основные характеристики для случая неограниченного числа источников:

$$p_0 = e^{-\rho}; \quad p_k = \frac{\rho^k}{k!} e^{-\rho}, \quad k \geq 1,$$

$$M\xi^{(2)} = \rho; \quad D\xi^{(2)} = \rho.$$

Заметим, что последний результат можно получить и другими способами.

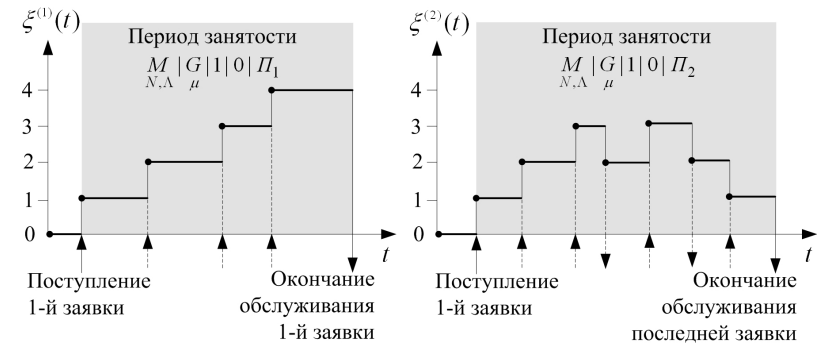


Рис. 1. Диаграммы случайных процессов $\xi^{(1)}(t)$ и $\xi^{(2)}(t)$

Литература

1. Наумов В.А., Самуйлов К.Е., Яржина Н.В. Теория телеграфика мультисервисных сетей: монография. — М.: Изд-во РУДН, 2007.

2. Рыков В.В. Сети обслуживания прозрачных требований // Автоматика и телемеханика. — 2001. — № 5. — С. 147–158.

3. Karvo J., Virtamo J., Aalto S. [et al.]. Blocking of dynamic multicast connections // Telecommunication Systems. — 2001. — V. 16, N. 3–4. — P. 467–481.

УДК 004.312.44

Е.С. Пьянков

zugus@mail.ru

Московский государственный институт электронной техники
(технический университет)

Модернизация аппаратной обработки обратной связи в сканирующем зондовом микроскопе

Для понимания свойств любого объекта необходимо знание его атомной структуры. Для таких исследований наиболее подходит сканирующий зондовый микроскоп. Общий принцип работы микроскопа заключается в следующем [1]. Система управления подает напряжение на керамику. В зависимости от напряжения она движется в одном из направлений. Вместе с керамикой в зависимости от конструкции микроскопа движется либо кантилевер, с закреплённой на нём иглой, либо сам образец. Определение изменения местоположения кантилевера и его высоты происходит так, как показано на рис. 1. Лазерный луч, отражённый от юстировочного зеркала, а затем от сканирующего зеркала и далее от кантилевера, попадает через линзу на четырёхсекционный фотодиод, расположенный в её фокальной плоскости. Линза устраняет паразитные изменения сигнала, связанные с параллельным смещением луча при сканировании. Для дальнейшего преобразования в систему обработки возвращаются 4 сигнала: A , B , C , D . В зависимости от изгибов кантилевера сигналы изменяются. После чего преобразуются в системе обработки и передаются в ЭВМ.

Одной из ключевых проблем при работе со сканирующим туннельным микроскопом является отделение полезной информации поступающей с зонда от шумов и наводок, неизбежных при такой точности

измерений. Необходима очень точная фильтрация, позволяющая усилить полезный сигнал, и отсеять помехи, не уменьшающая скорость работы прибора.

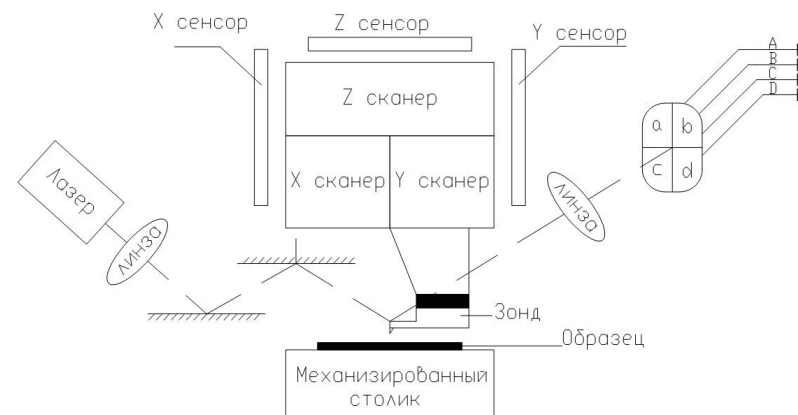


Рис. 1. Схема прохождения луча лазера

Рассмотрим алгоритм обработки обратной связи на примере контроллера сканирующего зондового микроскопа Р8 производства компании НТ-МДТ при контактном режиме сканирования. В начале вычитывается разностный сигнал DFL между верхней и нижней половинами фотодиода $(A + B) - (C - D)$ для определения изгиба кантилевера. Далее из полученного значения вычитается заранее заданная контрольная точка. И полученные данные передаются на интегратор. Интегратор устроен так, что на выходе его всегда будет единица, если на входе 0. Если сигнал на входе отличен от 0, то так как у операционного усилителя большой коэффициент усиления, он подает смещение на керамику. Таким образом, идёт корректирование положения кантилевера в зависимости от рельефа образца сканирования, для вывода информации на ЭВМ. Сигнал DFL сначала подается на матрицу. Матрица работает как мультиплексор — выбирает из нескольких сигналов один в зависимости от входных условий. Дальше сигнал через фильтр второго порядка подается на АЦП. После преобразования в цифровой вид сигнальный процессор преобразует полученные данные во внутренний формат и передает их в ЭВМ (рис. 2).

Для улучшения работы устройства было принято решение осуществлять обратную связь через сигнальный процессор, в котором следует производить вычитание контрольной точки и интегрирование данных после чего производить фильтрацию полученных дан-

ных, для отбрасывания шумов кантилевера. Таким образом, за счёт увеличения точности обратной связи возможно увеличение скорости сканирования и качества конечного изображения (рис. 3).

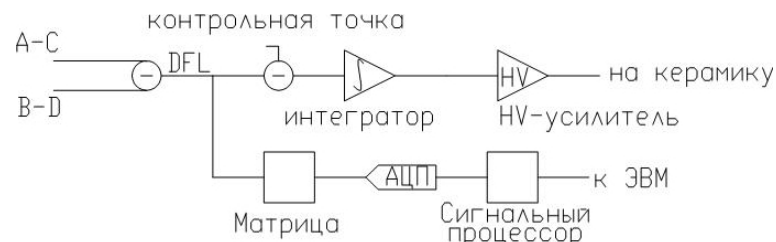


Рис. 2. Контактный режим в контроллере сканирующего зондового микроскопа Р8

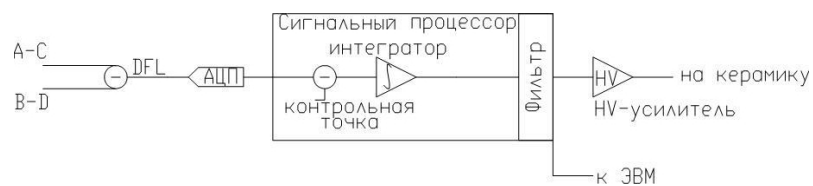


Рис. 3. Контактный режим в контроллере сканирующего зондового микроскопа Р9

Схемотехническое усовершенствование проведено и реализовано в контроллере сканирующего зондового микроскопа Р9 производства компании НТ-МДТ. В настоящее время идёт поиск оптимального фильтра, который бы качественно отфильтровал шум кантилевера без замедления обработки обратной связи.

Литература

1. *Быков В.А.* Приборы и методы сканирующей зондовой микроскопии для исследования и модификации поверхностей: дис. на иск. уч. ст. д. так называемый. — М., 2000. — 393 с.

УДК 512.64

А.В. Селиверстов, В.А. Любецкий

slvstv@iitp.ru, lyubetsk@iitp.ru

Институт проблем передачи информации им. А.А. Харкевича РАН

Об эллипсоидах, внутри которых нет целых точек

Решеткой называется свободная конечнопорожденная аддитивная подгруппа в n -мерном линейном пространстве над полем рациональных чисел. Декодирование кодов для гауссовского канала и задача квантизации связаны с поиском точек решётки, ближайших к данной точке евклидова пространства. Примеры можно найти в [1]. В частности, представляет интерес следующие задачи. Насколько много точек решётки может находиться на сфере, не содержащей внутри точек решётки? Какие алгебраические соотношения связывают точки решётки, лежащие на такой сфере? Очевидно, эти задачи эквивалентны описанию целых точек, лежащих на эллипсоиде, внутри которого нет ни одной целой точки. Действительно, подходящее аффинное преобразование взаимно однозначно переводит точки решётки в целые точки, а сферу — в эллипсоид, причём взаимное расположение точек и эллипсоида сохраняется.

Известно, что в каждой размерности для каждой целой точки a существует такое множество S целых точек, что имеется ровно один эллипсоид с центром в a , не содержащий внутри других целых точек и содержащий на его поверхности все точки из множества S , см. обзор [2]. Мы покажем: если внутри эллипсоида нет целых точек, то в низких размерностях он не определяется лежащими на нём целыми точками. Мы предполагаем, что это остаётся верным и для больших размерностей. А также мы приведём точную верхнюю границу для числа целых точек на эллипсоиде, не содержащем внутри себя целых точек.

Теорема. а) Если внутри эллипсоида нет целых точек, то на нём лежит не более чем 2^n целых точек. Эта граница достигается.

б) В размерностях два и три, если внутри эллипсоида нет целых точек, то через все целые точки, лежащие на нём, можно провести бесконечно много различных эллипсоидов.

Доказательство. а) На сфере, описанной около единичного куба, лежит ровно 2^n целых точек. Допустим, на таком эллипсоиде лежит $2^n + 1$ целых точек. Тогда найдутся точки x и y , для которых соответствующие координаты сравнимы по модулю два: $x_i = y_i \pmod{2}$. Следовательно, полусумма $(x + y)/2$ — целая точка и середина отрезка с концами, лежащими на эллипсоиде; противоречие.

б) Коэффициенты многочлена f от n переменных, определяющего эллипсоид $f = 0$, соответствуют точке проективного пространства, поскольку при умножении многочлена f на ненулевое число эллипсоид не меняется. Размерность множества эллипсоидов равна $(n^2 + 3n)/2$. Эллипсоиды, проходящие через некоторую целую точку x , соответствуют точкам пространства эллипсоидов, принадлежащих гиперплоскости, определяемой линейным однородным уравнением $f(x) = 0$. Чтобы система таких гиперплоскостей имела единственную точку пересечения, их число должно быть не меньше размерности. Поэтому на эллипсоиде должно лежать не менее чем $(n^2 + 3n)/2$ целых точек. Однако из пункта (а) следует, что таких точек не более чем 2^n . При $n < 4$ выполнено неравенство $(n^2 + 3n)/2 > 2^n$. Следовательно, через целые точки, лежащие на эллипсоиде, проходит бесконечно много квадратиков. Эти квадратики образуют связное множество. Поскольку небольшое возмущение сохраняет положительную определённую квадратичной формы, то среди рассматриваемых квадратиков найдется бесконечно много эллипсоидов. Доказательство завершено.

Напомним, что областью Делоне, соответствующей данной n -мерной решётке, называется любой из многогранников, образующих разбиение n -мерного аффинного пространства, двойственное к разбиению этого пространства на области Вороного (разбиение L из статьи [3]). Пункт (а) теоремы можно переформулировать так: число вершин у каждой области Делоне, соответствующей n -мерной решётке, не превышает 2^n .

Доказательство пункта (а) теоремы показывает, что по модулю две точки на эллипсоиде соответствуют вершинам параллелепипеда, через которые можно провести бесконечно много эллипсоидов в каждой размерности. Это позволяет сформулировать следующую гипотезу: в конечной размерности, если внутри эллипсоида нет целых точек, то через все целые точки, лежащие на нем, можно провести бесконечно много различных эллипсоидов.

Авторы благодарны К.Ю. Горбунову и С.А. Пирогову за обсуждение результатов. Работа выполнена при частичной поддержке Международного научно-технического центра (проект 3807).

Литература

1. Конвей Дж., Слоэн Н. Упаковки шаров, решётки и группы. — М.: Мир, 1990.
2. Рышков С.С., Барановский Е.П. Классические методы теории решетчатых упаковок // Успехи математических наук. — 1979. — Т. 34, № 4. — С. 3–63.
3. Делоне Б.Н. Геометрия положительных квадратичных форм // Успехи математических наук. — 1937. — Т. 3. — С. 16–62.

УДК 004.021, 519.2

А.И. Черноиванов, А.В. Королькова
 acher@sci.pfu.edu.ru, akorolkova@sci.pfu.edu.ru
 Российский университет дружбы народов

Моделирование при помощи стохастических дифференциальных уравнений поведения TCP-трафика при взаимодействии с узлом, работающим по алгоритму RED

Несмотря на дискретную природу процессов в сетях передачи данных, для анализа их поведения можно использовать непрерывные модели. В данной работе предлагается использовать стохастические дифференциальные уравнения для моделирования поведения TCP-трафика при взаимодействии с узлом, работающим по алгоритму RED (Random Early Detection) [1].

Рассматриваемая в работе система состоит из узла-источника, генерирующего TCP-Reno трафик, и узла-получателя, состоящего из модуля RED, управляющего сбросом пакетов, буфера конечной ёмкости и модуля мгновенной обработки поступающих из очереди пакетов.

Поведение размера окна TCP-Reno описывается с помощью стохастического дифференциального уравнения:

$$\begin{aligned} \frac{w(t)}{dt} = & \frac{1}{T(t)} [I_w(w_{\max} - w(t)) + \\ & + \left(-\frac{w(t)}{2}\right) (1 - F_{TO}(w(t)))w(t - \tau)[p_M(t - \tau) + p_D(t - \tau)] + \\ & + (1 - w(t))F_{TO}(w(t))w(t - \tau)p_D(t - \tau)], \end{aligned} \quad (1)$$

где $w(t)$ — размер TCP-окна (в пакетах) в момент времени $t \geq 0$, w_{\max} — максимально возможный размер TCP-окна, $F_{TO}(w(t))$ — вероятность потери пакетов по тайм-ауту, $p_M(t - \tau)$ — вероятность маркировки пакета на сброс, $p_D(t - \tau)$ — вероятность сброса пакета с учётом времени $\tau \geq 0$ оповещения источника о потере пакета,

$$I_w(x) = \begin{cases} 1, x > 0, \\ 0, x \leq 0, \end{cases}$$

— функция-индикатор, которая не позволит размеру окна превысить своё максимальное значение w_{\max} , $T(t) = T_b + \frac{q(t)}{C(t)}$ — период времени, состоящий из времени T_b прохождения одного пакета от источника до узла обработки и времени задержки пакета в очереди $\frac{q(t)}{C(t)}$, $q(t)$ — мгновенная длина очереди, $C(t)$ — интенсивность поступления пакетов из очереди в узел на обслуживание.

Поведение функции мгновенной длины очереди $q(t)$ описывается стохастическим дифференциальным уравнением:

$$\frac{dq(t)}{dt} = \frac{w(t)}{T(t)}(1 - p_D(t)) - C(t).$$

Вероятность сброса пакетов $p(t) = p_D(t) + p_M(t)$ определяется согласно алгоритму RED (см., например, [1, 2]). Причём эта вероятность зависит от экспоненциально взвешенной скользящей средней (EWMA — Exponentially Weighted Moving Average) длины очереди $\bar{q}(t)$, определяемой из уравнения

$$\frac{d\bar{q}(t)}{dt} = \frac{\ln(1 - w_q)}{\delta}(\bar{q}(t) - q(t)), \quad (2)$$

где $w_q = 1 - e^{-1/C}$ — весовой коэффициент в RED, C — пропускная способность канала, $C(t) \leq C$.

Полученная в работе система уравнений (1)–(2) решена численно со следующими начальными значениями параметров: $w_0 = 1$ — начальный размер TCP-окна (в пакетах), $q_0 = 0$, $\bar{q}_0 = 0$, $R = 100$ — размер буфера (в пакетах), $q_{\min} = 20$, $q_{\max} = 60$; 80 — минимальное и максимальное пороговые значения для алгоритма RED, $w_q = 0,0007$, $T_b = 0,01$ с, $\delta = 0,01$ с⁻¹, $C = 1600$ пакетов.

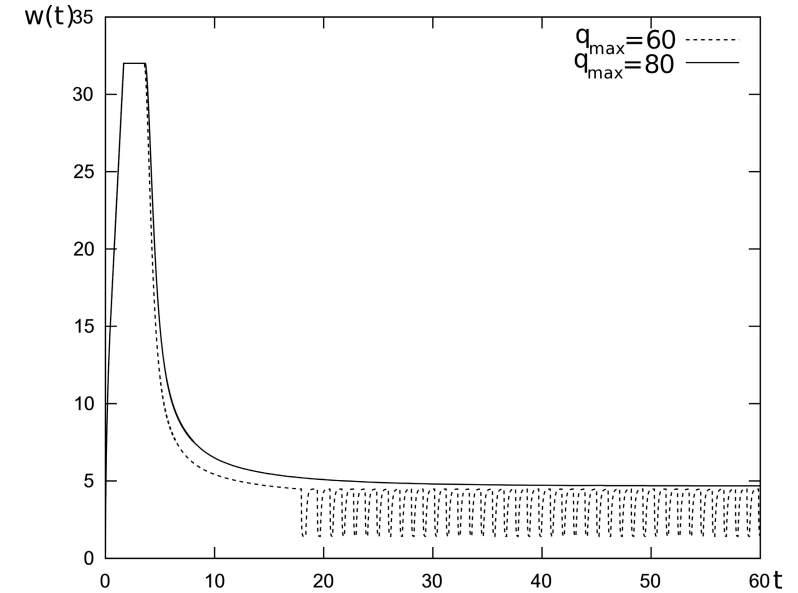


Рис. 1. Изменение TCP-окна

На рис. 1 показано изменение TCP-окна в зависимости от времени. Сначала размер TCP-окна быстро растёт, поскольку нет потерь пакетов. Затем окно некоторое время не изменяется, насыщая очередь пакетами. Как только возникает потеря пакетов, размер окна начинает стремительно сокращаться. Пунктирной линией представлен случай не оптимального подбора параметров RED, поскольку происходят постоянные осцилляции размера окна. Сплошной линией показано поведение системы при оптимальном подборе параметров.

Поведение мгновенной длины очереди $q(t)$ представлено на рис. 2. Сначала происходит заполнение очереди поступающими пакетами. Затем включается механизм RED сброса пакетов и количество пакетов в очереди начинает плавно снижаться. При этом возникает ситуация, когда очередь уже свободна для поступления новых пакетов, однако RED все ещё не позволяет пакетам заполнить очередь, что

проявляется в стремительном падении числа пакетов в очереди. Затем система постепенно выходит на стационарный режим работы при оптимальном подборе параметров, а при не оптимальном — значения длины очереди колеблются около верхнего порогового значения, внося нестабильность в работу системы.

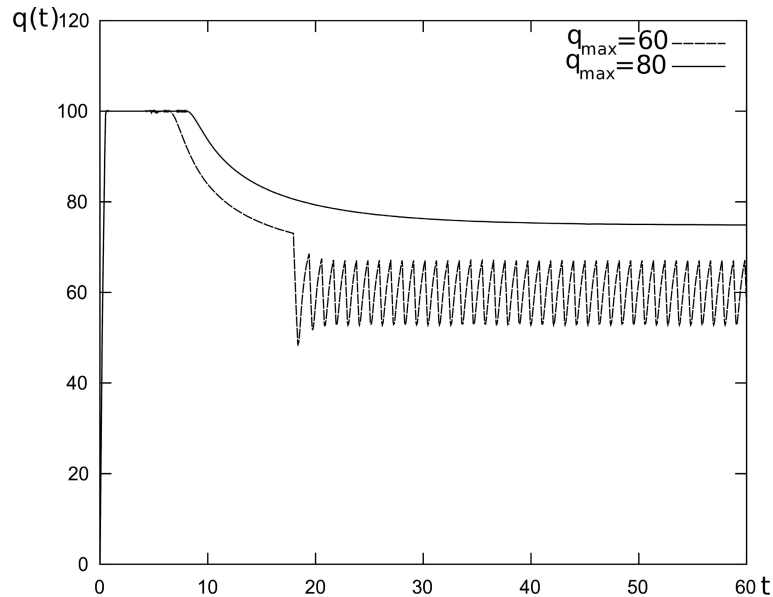


Рис. 2. Изменение мгновенной длины очереди

Литература

1. Floyd S., Jacobson V. Random Early Gateways for Congestion Avoidance // IEEE/ACM Transactions on Networking. — 1993. — No1(4). — P. 397–413.
2. А.В.Королькова, Д.С.Кулябов, А.И.Черноиванов. К вопросу о классификации алгоритмов RED // Вестник РУДН. Серия «Математика. Информатика. Физика». — 2009. — № 3. — С. 34–46.

УДК 519.246.8

Е.С. Шубаева

katerinashibaeva@gmail.com

Российский университет дружбы народов

Анализ сетевого трафика на основе методики ARIMA

При моделировании сетевых процессов часто встречаются ситуации, когда имеется реализация сетевого трафика и надо подобрать соответствующую теоретическую конструкцию, которая могла бы породить такую реализацию. Для изучения трафика можно использовать методы исследования временных рядов:

- методы линейного анализа,
- методы нелинейного анализа.

В данной работе проведено практическое изучение сетевого трафика на основе методики ARIMA при помощи программного средства R [1]. Эксперименты проводились на нескольких реализациях сетевого трафика, представленных в [2]. Для линейного анализа применяется подход Бокса–Дженкинса [3] по построению моделей типа ARIMA.

Рассмотрена реализация BC-Oct89Ext.TL, полученная Will Leland и Dan Wilson. Данная реализация длится 122797,83 секунд, содержит 1 миллион записей об Ethernet-кадрах, проходящих между сетью корпорации Bellcore и сетью Internet. Данные представлены двумя колонками чисел. Первый столбец — время в секундах, второй — длина поля данных Ethernet-кадра в байтах (не включая преамбулу, заголовка и контрольной).

Минимальный и максимальный размер кадра составляет 64 байта и 1515 байт соответственно. Для наглядности представлено распределение размера кадра (рис. 1). Как видно из гистограммы, частота появления кадров размером больше 600 байт существенно ниже, чем частота появления кадров в 600 байт. Это обусловлено тем, что самый распространенный Maximum Transmission Unit (MTU) для передачи данных по внешним линиям связи составляет 576 байт.

По данным первоисточника реализации BC-Oct89Ext.TL [2] точность временных отсчетов составляет порядка 10 микросекунд. Необходимо выполнить агрегирование исходной реализации для приведе-

ния её к шкале с постоянным шагом по оси времени. Агрегирование проводилось с двумя уровнями агрегации: 5 с и 100 с.

В дальнейшем будем обозначать получившиеся временные ряды ВС.5 и ВС.100 соответственно. На рис. 2 показан внешний вид рядов ВС.5 и ВС.100.

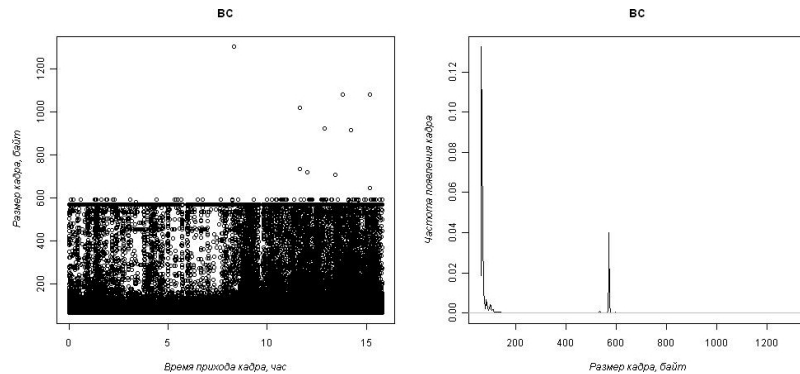


Рис. 1. Реализация ВС-Oct89Ext.TL

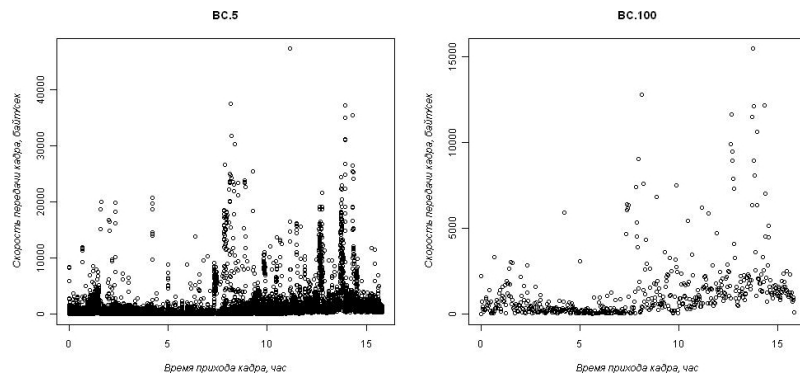


Рис. 2. Распределение рядов ВС.5 и ВС.100

Построим модель ARIMA для временных рядов ВС.5 и ВС.100.

AR — процесс авторегрессии $X_t = \sum_{i=1}^p \alpha_i X_{t-i} + \varepsilon_t$, где α_i — коэффициент ряда, ε_t — белый шум.

MA — процесс скользящего среднего $X_t = \sum_{j=0}^q \beta_j \varepsilon_{t-j}$, где β_j — коэффициент ряда, ε_{t-j} — белый шум.

ARIMA (p, d, q) — авторегрессионная модель интегрированного скользящего среднего. При этом p — параметр AR, d — степень интеграции и q — параметр MA.

Бокс и Дженкинс предложили разбить задачу построения модели типа ARIMA по реализации случайного процесса на несколько этапов:

1) установить порядок интеграции d , то есть добиться стационарности ряда,

2) по полученному ряду построить ARMA (p, q).

При построении моделей временных рядов критерии качества подгонки применяются для сравнения моделей между собой. Наиболее распространенным в настоящее время является критерий AIC [4].

Для рассматриваемых временных рядов ВС.5 и ВС.100 критерий AIC минимален при $d = 1$. С увеличением p и q AIC незначительно уменьшается. Для рядов ВС.5 и ВС.100 минимальный показатель AIC у ARIMA (19,1,10) и ARIMA (18,1,4) соответственно.

Для реализации BC-Oct89Ext.TL была прослежена слишком длинная зависимость, что говорит о фрактальности трафика. Следовательно, линейные методы анализа реализаций плохо применимы для данного трафика. Рекомендуется применять методы нелинейного анализа, такие, как концепция суррогатных данных, метод ложных соседей и вычисление корреляционного интеграла.

Литература

1. The R Project for Statistical Computing: <http://www.r-project.org>.
2. The Internet Traffic Archive: <http://ita.ee.lbl.gov>.
3. Box G.E.P., Jenkins G.M. Time Series Analysis, Forecasting and Control. — rev. Ed. San Francisco: Holden-Day, 1976.
4. Akaike H. A New Look at the Statistical Model Identification // IEEE Transactions on Automatic Control. — 1974. — AC-19.

Секция радио и информационных технологий

УДК 621.396.677

А.В. Бабинцев

leha@frtk.ru

Московский физико-технический институт
(государственный университет)

Многолучевая зеркальная антенна для спутниковых систем связи

На данный момент точность удержания спутников на геостационарной орбите (ГСО) стала очень высокой. Так, на некоторых её участках угловое расстояние между соседними спутниками составляет менее, чем 1 градус. Ввиду этого возникает возможность принимать сигнал одновременно с нескольких ИСЗ, находящихся на геостационарной орбите, на одну антенну, называемую многолучевой антенной (МЛА).

Целью моего доклада является ознакомление слушателей с особенностью построения и принципом работы спроектированной мною МЛА. Антенна рассчитана на прием сигналов со спутников, расположенных в пределах сектора 50 градусов на ГСО, при этом разнос между максимумами диаграмм направленности в направлении на два соседних спутника может составлять 2 градуса.

Направления максимумов приёма такой антенны по отдельным лучам располагаются по образующим конической поверхности, вершина которой совпадает с точкой размещения антенны по отдельным лучам на поверхности земли, а направляющая этой поверхности совпадает с участком обслуживаемой ГСО.

Азимут α и угол места β каждого из направлений на ИСЗ, расположенного на геостационарной орбите, определяются выражениями

$$\beta(\varphi_s) = \arctan\left(\frac{\cos(\varphi_0 - \varphi_s) \cdot \cos(\varphi_p) - 0,15}{\sqrt{1 - (\cos(\varphi_0 - \varphi_s) \cdot \cos(\theta_p))^2}}\right),$$

$$\alpha(\varphi_s) = \arcsin\left(\frac{\sin(\varphi_p - \varphi_s)}{\sqrt{1 - (\cos(\varphi_p - \varphi_s) \cdot \cos(\theta_p))^2}}\right),$$

где φ_p , φ_s — долготы пункта размещения антенны земной станции и подспутниковой точки соответственно; θ_p — широта пункта размещения земной станции. Здесь азимут отсчитывается от направления на юг по часовой стрелке.

Пример зависимости углов места точек ГСО в угловых градусах от восточной долготы подспутниковой точки приведён на рис. 1.

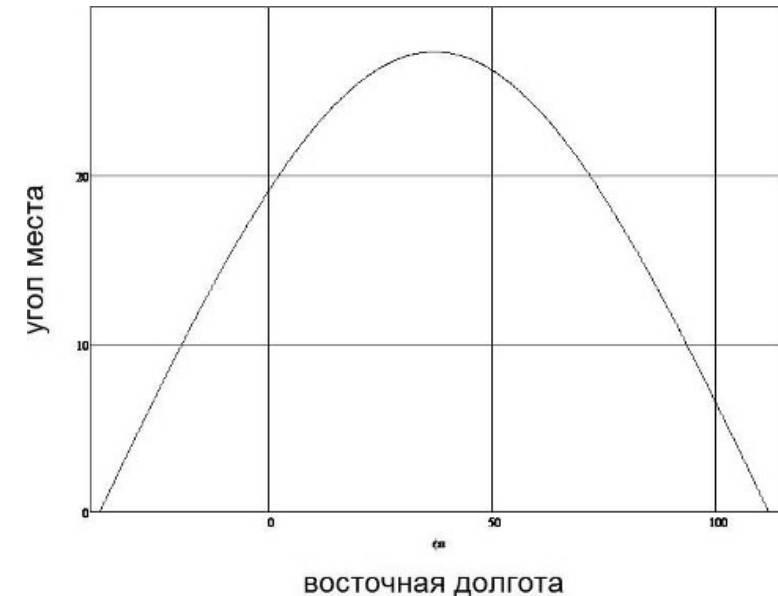


Рис. 1. Видимый участок ГСО

За основу построения МЛА в данной работе взята схема Кассегрена двухзеркальной антенны (рис. 2). Поверхность основного зеркала антенны образуется при вращении несимметричной части параболы относительно оси, лежащей в плоскости этой параболы, причём эта ось перпендикулярна фокальной оси параболы. В плоскости параболы пучок параллельных лучей при приеме отражается в точку расположения фокуса параболы, а в ортогональной плоскости — в параксиальный фокус, находящийся посередине радиуса кривизны тора.

Геометрическую форму поверхности такого зеркала можно представить в виде его сечений в двух ортогональных плоскостях. В одной плоскости сечение имеет вид параболы, в другой — окружности.

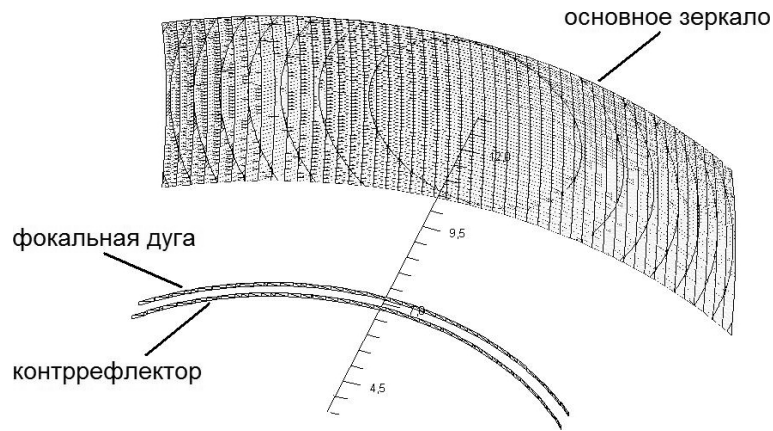


Рис. 2. Внешний вид антенны

Поверхность контррефлектора антенны формируется аналогично поверхности основного зеркала. Сечение в одной из ортогональных плоскостей этого контррефлектора имеет вид гиперболы, в другой — окружности.

Угловой разнос лучей в направлении двух соседних ИСЗ будет определяться угловым разносом облучателей на фокальной дуге. Значение углового разноса определяется размерами раскрыва облучателей, а общий угловой охват углового сектора, соответствующего сектору ГСО, определяется общим угловым размером этой дуги.

В плоскости размещения облучателей в раскрыве антенны появляются квадратичные фазовые ошибки, которые проявляются к некоторому расширению основного лепестка ДН, увеличению боковых и снижению коэффициента усиления.

В результате проделанной работы были получены формулы, определяющие диаграмму направленности (ДН) и основные энергетические параметры МЛА.

ДН, создаваемое всей поверхностью в вертикальной плоскости, определяется выражением:

$$EV(\theta) = -i \cdot \frac{\text{const}}{2r \cdot \lambda} \cdot (1 + \cos(\theta)) \times$$

$$\int_{-\frac{DV}{2}}^{\frac{DV}{2}} \int_{-\frac{DH}{2}}^{\frac{DH}{2}} F \cdot \exp(-i \cdot \frac{2 \cdot \pi}{\lambda} \cdot (r - y \cdot \sin(\theta) + \Delta\Phi)) dx dy$$

В горизонтальной:

$$EH(\theta) = -i \cdot \frac{\text{const}}{2r \cdot \lambda} \cdot (1 + \cos(\theta)) \times$$

$$\int_{-\frac{DH}{2}}^{\frac{DH}{2}} \int_{-\frac{DV}{2}}^{\frac{DV}{2}} F \cdot \exp(-i \cdot \frac{2 \cdot \pi}{\lambda} \cdot (r - y \cdot \sin(\theta) + \Delta\Phi)) dx dy,$$

где DH, DV — ширина и высота основного зеркала МЛА соответственно, F — распределение поля в раскрыве, r — расстояние до спутника, λ — длина волны излучения, $\Delta\Phi$ — фазовые искажения, $\theta = 0$ соответствует направлению на спутник.

Вид ДН при приеме сигнала одновременно с нескольких спутников показан на рис. 3.

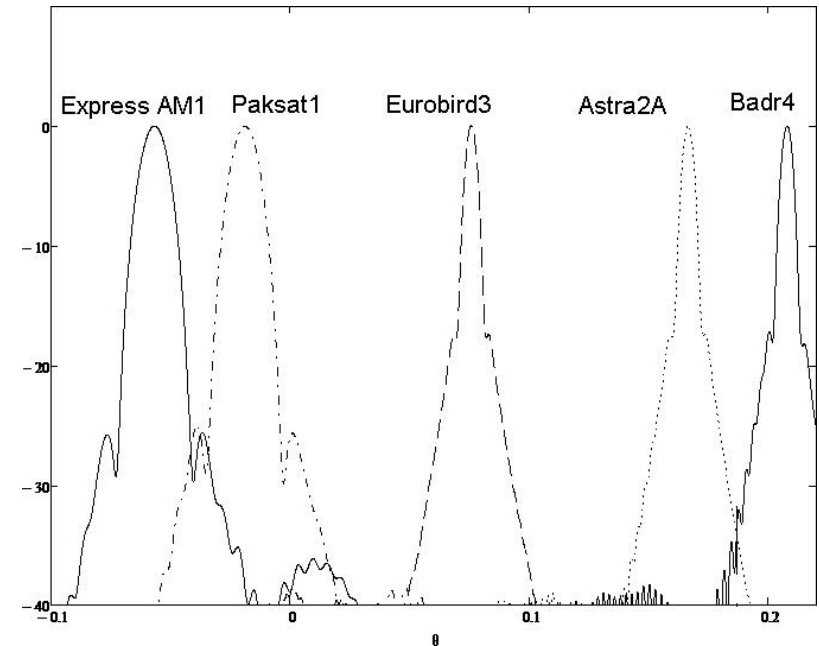


Рис. 3. Одновременный прием сигналов с 5 спутников

Литература

1. Айзенберг Г.З., Ямпольский В.Г., Терешин О.Н. Антенны УКВ. Ч. 1. — М.: Связь, 1977. — 382 с.

2. *Айзенберг Г.З., Ямпольский В.Г., Терешин О.Н.* Антенны УКВ. Ч. 2. — М.: Связь, 1977. — 288 с.
3. *Сомов А.М.* Метод фрагментации для расчёта шумовой температуры антенн. — М.: Горячая линия. — телеком, 2008. — 208 с.
4. Вуд П. — Анализ и проектирование зеркальных антенн. — М.: Связь, 1984. — 208 с.
5. *Бахрах Л.Д., Фельд Я.Н., Зелкин Е.Г.* Справочник по антенной технике. — М.: Радиотехника, 1997. — 256 с.
6. *Шпидлер М.* Практические конструкции антенн. — М.: Мир, 1989. — 448 с.

УДК 004

В.В. Бутенко, А.П. Назаренко, В.К. Сарьян

info@niir.ru, info@niir.ru, sarian@niir.ru

Научно-исследовательский институт радио

Проблемы эволюции инфокоммуникационных сетей

Современный этап построения информационного общества характеризуется прежде всего стремительным ростом инфокоммуникационных технологий (ИКТ). Новые технологии, не успевая широко внедриться, начинают вытесняться новыми, более совершенными технологиями, например: 3G, 4G, Wi-Fi, Wi-Max, LTE, NGN, DVB-T, DVB-T2, HDTV, 3DTV и др., способствуют конвергенции сетей и стимулируют формирование единой инфокоммуникационной среды (ИКС) [1].

Благодаря высоким темпам развития, ИКТ в постиндустриальных странах превратился в мощный инструмент повышения производительности труда и экономического роста, создания новых рабочих мест с расширением возможностей трудоустройства любых категорий граждан, в том числе и с ограниченными возможностями. Развитие ИКТ приводит к заметному повышению качества жизни миллионов людей в этих странах за счёт обеспечения доступа массового абонента в ИКС к широкой номенклатуре инфокоммуникационных услуг, в том числе и услуг, в основе которых лежат высокие технологии [2].

Однако эволюция информационного общества в развитых странах, как только ИКС позволила реально обеспечить доступ уже массового пользователя к инфокоммуникационным услугам, начала сопровождаться возникновением новых барьеров (препятствий, ограничений), которые хотя и не могут ещё повлиять на скорость эволюции, но заметно снижают положительный эффект от её внедрения и поэтому требуют скорейшего решения [3]. К ним можно отнести:

- Возросшие объёмы энергопотребления за счёт использования ИКТ, которое по величине стали сравниваться или даже превосходить объёмы потребления в ЖКХ и промышленности вместе взятых;
- Возросшие масштабы преступлений, мошенничества, злоупотреблений и обмана пользователей при предоставлении инфокоммуникационных услуг, в том числе и нарушений прав граждан на конфиденциальность личных данных;
- Тревожные факты по неблагоприятному воздействию ИКТ на психику детей, в первую очередь Интернета;
- Существенный перевес тематики и времени предоставления инфокоммуникационных услуг в сторону игр и развлечений.
- Возросший риск техногенных катастроф, связанный с обеспечением доступа массового абонента к инфокоммуникационным услугам, основанным на высоких технологиях;
- Возросшее число нарушений авторских и смежных прав владельцев интеллектуальной собственности;
- Возросшее число психических болезней среди молодёжи, связанных с компьютерной зависимостью и др.
- Возросшее влияние неионизированного излучения (СВЧ-излучения), открытым источником которого является мобильный терминал, на здоровье людей.

Тем не менее темпы развития ИКТ продолжают возрастать: заметно растёт число абонентов, объёмы хранения, объёмы и скорости передачи и обработки данных, наращивается экспорт в развивающиеся страны. Предполагается, что создаваемое глобальное информационное общество за счёт процессов самоорганизации сможет справиться с возникшими негативными проблемами «по ходу дела».

Однако такое предположение не оправдывается, о чем свидетельствуют многочисленные публикации. Тем не менее декларациями World Summit on the Information Society (в Женеве и Тунисе) было провозглашено и закреплено, как одна из главных целей мирового сообщества, на ближайшее время, построение глобального информационного общества, в котором каждому жителю планеты обеспечива-

ется равный не дискриминационный доступ к любой информации и инфокоммуникационным услугам, что означает дальнейшее ускорение темпов эволюции глобального информационного общества.

Вместе с тем нынешний опыт функционирования информационного общества, показывает, что его эволюция уже вошла в нелинейную область, и дальнейшее подключение к ИКТ новых абонентов и расширение номенклатуры предоставляемых им инфокоммуникационных услуг вызывает ускорение темпов истощения не только природных и материальных ресурсов, но также психических и физических ресурсов (здоровья) человека.

Самая главная, системная причина всех неприятностей, сопровождающих современный этап эволюции информационного общества, на наш взгляд, состоит в том, что при провозглашении и обеспечении массового доступа к инфокоммуникационным услугам оказался плохо проработан вопрос о формировании предоставляемых услуг и контроля за качеством их предоставления. Существующие структуры и нормативно-правовая база ориентированы только на контроль качества коммуникационной составляющей предоставляемых услуг, они не в состоянии контролировать содержательную (информационную) часть услуги, которая собственно является, как правило, основной при предоставлении инфокоммуникационных услуг. Следствием этого является увеличение расходов, связанных с администрированием, в расчёте на одну услугу, а значит, снижению роста доходов и истощению ограниченных ресурсов. В связи с этим для поддержания постоянного уровня доходов операторам приходится затрачивать средства на разработку все более новых технологий, что приводит к нелинейному увеличению потребления ограниченных ресурсов и стремительно приближает эволюцию информационного общества к точке сингулярности [4].

Важно то, что из-за процессов глобализации надвигающийся системный кризис, связанный с переходом через точку сингулярности, человеческая цивилизация будет преодолевать как единое целое. Либо она как единая система преодолеет кризис, либо её ждёт глобальная катастрофа. Это существенно отличает механизм преодоления кризисов сингулярности от предыдущих цивилизационных кризисов. Тогда эволюция всегда имела возможность принести в жертву недостаточно гибкие подсистемы цивилизации, и, воспользовавшись избыточным разнообразием, передать лидерство более удачливым подсистемам.

Литература

1. *Бутенко В.В., Назаренко А.П., Сарьян В.К.* Внедрение многофункциональной мультимедийной интеллектуальной приставки и / или телевизионного приёмника. — залог успешного развития эфирного телевизионного вещания // Broadcasting. Телевидение и радиовещание. — 2008. — № 6.
2. *Назаренко А.П., Сарьян В.К.* Новый подход к развитию инфокоммуникационных технологий // Заметки с выставки СеВIT. — 2008.
3. *Сарьян В.К.* Сотовый телефон. — от конфронтации к сотрудничеству // Электросвязь. — 2005. — № 5.
4. *Назаретян А.П.* Цивилизационные кризисы в контексте универсальной истории // Синергетика. — Психология. — Прогнозирование. — М.: Мир, 2004.

УДК 621.397

С.А. Канев^{1,2}, И.Н. Красносельский^{2,3}

¹ Московский технический университет связи и информатики

² Научно-исследовательский институт радио

³ Московский физико-технический институт
(государственный университет)

Системы цифрового телевизионного вещания второго поколения

Задача повышения эффективности систем цифрового телевизионного (ТВ) вещания была решена разработкой стандартов второго поколения DVB-S2, DVB-T2, DVB-C2 [1–3], в которых регламентированы новые методы канального кодирования и модуляции. Сочетание стандартов второго поколения DVB и компрессии видеoinформации MPEG-4 AVC/H.264 даёт возможность увеличить число программ в мультиплексе и передавать программы телевидения высокой четкости (ТВЧ).

Системы второго поколения DVB-S2, DVB-T2 и DVB-C2, обладая повышенной, примерно на 30–50% пропускной способностью, макси-

мально приближены друг к другу в плане форматов данных, необходимых для передачи, и уровней адаптации к сети вещания, в которую входит предобработка изначальных данных, адаптации потока в основной полосе и канальное кодирование.

Предобработка данных на входе блока канального кодирования включает в себя: синхронизацию, компенсацию задержки для транспортно-го потока, обнаружение и удаление нулевых пакетов (с дальнейшим их восстановлением на приёмной стороне), вставку битов проверки на четность для обнаружения ошибок при дальнейшей обработке.

Во втором поколении стандартов для канального кодирования применяются коды БЧХ в качестве внешних кодов и коды с малой плотностью проверок на чётность (LDPC) в качестве внутренних кодов. Возможности использования кодовых скоростей расширены до $1/4$, $1/3$, $2/5$, $1/2$, $3/5$, $2/3$, $3/4$, $4/5$, $5/6$, а для спутниковой системы дополнительно введены значения $8/9$ и $9/10$. Использование сочетания LDPC/VCH-кодов более эффективно по сравнению с каскадами кодов Рида–Соломона и сверточными кодами. Так, например, для системы DVB-S2 при выборе вида модуляции, аналогичного системе DVB-S, и при той же спектральной эффективности требуемое отношение энергии на бит к спектральной плотности шума снижается на 2,5 дБ [4].

В спутниковой системе DVB-S2 применяются следующие методы модуляции: QPSK, 8PSK, 16APSK и 32APSK. Для вещательных приложений предусмотрены QPSK, 8PSK. Для каналов с очень низким отношением сигнал/шум рекомендуется использование QPSK модуляции в сочетании с кодовыми скоростями $1/4$, $1/3$ и $2/5$. Модуляция 16APSK в основном предназначена для организации магистральных линий при условии высокой линейности транспондера. Режим 32APSK ориентирован на профессиональное применение и требует высокого отношения сигнал/шум в сочетании с внесением предорганизации обратного канала предусмотрен вариант адаптивного кодирования и модуляции искажений нелинейности транспондера на передающей стороне.

В качестве схемы модуляции DVB-T2, как и в DVB-T, используется COFDM, но кроме принятых ранее режимов 2k, 4k, 8k, применяются также 1k, 16k, 32k. Виды первичной модуляции дополнены 256QAM. Дополнительные значения относительных защитных интервалов по сравнению с системой DVB-T составляют $19/256$, $19/128$ и

$1/128$. Кроме того, появилась возможность работы с временным перемещением кадров [2].

Использование большего числа поднесущих (режим 16k и 32k) благоприятно с точки зрения дальности распространения сигнала, так как при увеличении числа поднесущих увеличивается длительность символа, и, как следствие, происходит увеличение длительности защитного интервала. Таким образом, возрастает защита от отражённых лучей, как следствие — распространение на большие расстояния. Кроме того, использование режимов с большим числом поднесущих более эффективно с точки зрения использования частотного ресурса, так как при этом форма спектра группового сигнала максимально приближена к прямоугольной. Минусом режимов 16k и 32k является меньшая устойчивость системы к эффекту Доплера. Таким образом, можно сделать вывод о том, что вещание с параметрами 16k и 32k применимо для покрытия больших зон с вещанием на стационарные приёмники.

Значительные изменения произошли и в системе DVB-C2. В данный стандарт была введена модуляция COFDM и пилот-сигналы. Кроме того, значительно повышено число позиций квадратурной модуляции [3]. Теперь оно может быть равно 4096. При высокой кратности модуляции помимо применения помехоустойчивых кодов VCH/LDPC предъявляются повышенные требования к качеству кабеля, по которому происходит распространение программ, и к условиям его прокладки.

Если рассматривать перспективы перехода на ТВЧ, то при условии использования стандарта сжатия H.264 на скорости от 7 до 9 Мбит/с при спутниковом вещании число телеканалов может составить 5–6 при использовании системы DVB-S2, вместо 4 для DVB-S. При передаче ТВЧ с помощью системы DVB-T2, число программ возрастает до 4 вместо 2–3 для системы DVB-T.

При использовании системы DVB-T2 возможное число программ в первом мультиплексе может быть повышено с 8 до 11–12. В спутниковых системах переход на DVB-S2 означает увеличение числа телеканалов стандартной четкости с 10–11 до 14–15 на один частотный канал.

Принятое в России решение об использовании стандарта сжатия H.264 в системах первого поколения (в первом мультиплексе) позволяет заранее унифицировать оборудование для производства программ и абонентские приемные устройства как для внедряемых систем первого поколения, так и для перспективных систем второго поколения.

Тем самым переход на новые системы будет менее затратным и технологически гладким.

Литература

1. ETSI EN 302 307 V1.1.2 Digital Video Broadcasting (DVB); Second generation framing structure, channel coding and modulation systems for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications.

2. ETSI EN 302 755 V1.1.1 Digital Video Broadcasting (DVB); Frame structure channel coding and modulation for a second generation digital terrestrial television broadcasting system (DVB-T2).

3. Draft ETSI EN 302 769 V1.1.1 Digital Video Broadcasting (DVB); Frame structure channel coding and modulation for a second generation digital transmission system for cable systems (DVB-C2).

4. DVB-S2 coding & modulation standard use for high data rate TM links. Experimental specification CCSDS 131.3-O-1. June 2007. — 8 p.

УДК 621.376.9

М.А. Карпова

m.a.karpova@inbox.ru

Московский физико-технический институт
(государственный университет)
Научно-исследовательский институт радио

Моделирование схем модемов с ФМ-4 и КАМ-16

Рассмотрим структурную схему модулятора-демодулятора с алгоритмом ФМ-4 (рис. 1). Сигнал с генератора псевдослучайной последовательности поступает на демультиплексор, который делит побитово исходную последовательность на два канала: I — синфазный канал и Q — квадратурный. Потоки битов обоих каналов проходят через фильтры типа «приподнятый косинус» для уменьшения межсимвольных помех. Затем сигналы поступают на два балансных модулятора, несущие частоты которых находятся в квадратуре [1]. На выходе балансного модулятора в каждом из каналов получается фазомодулированный сигнал со скоростью 1 бит/символ. После сумматора сигнал

передается со скоростью 2 бит/символ. В демодуляторе сигнал поступает на делитель мощности, после этого каждый канал перемножается с сигналом модулирующей частоты и поступает в фильтр низких частот типа «приподнятый косинус», а затем на решающее устройство по каждому из каналов. После того как последовательности бит выделены, они мультиплексируются в один цифровой сигнал.

Моделирование проводилось с помощью Simulink Matlab. Были выбраны частота несущей, частота синтезатора частот, частота дискретизации ФНЧ типа «приподнятый косинус», групповая задержка отсчетов, сглаживающий фактор так, чтобы вероятность ошибки демодулирования была почти нулевой. В качестве решающего устройства используется ФАПЧ типа «петля Костаса».

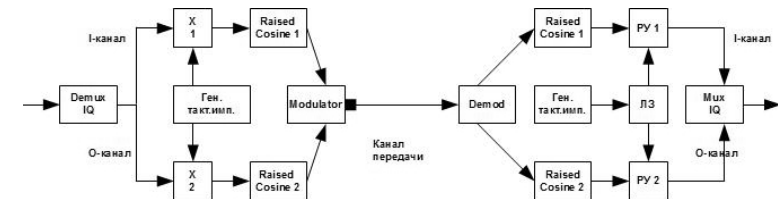


Рис. 1. Структурная схема МоДема с ФМ-4

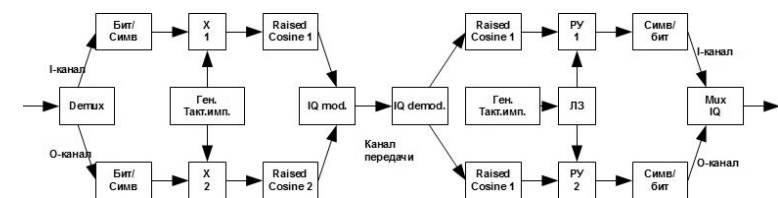


Рис. 2. Структурная схема модема с кам-16

Рассмотрим структурную схему МоДема с КАМ-16 рис. 2. Она отличается от схемы МоДема с ФМ-4 только наличием устройства «Бит/символ», в котором комбинация из двух бит преобразуется в один из уровней сигнала с логическими значениями -3 , -1 , 1 или 3 , и «Символ/бит», в котором проводится обратная процедура (уровень сигнала преобразуется в последовательность из двух бит). Таким образом, после модуляции скорость в каждом из каналов 2 бит/символ, а в канале передачи — 4 бит/символ.

Аналогично первой схеме все параметры МоДема с КАМ-16 подобраны так, чтобы на выходе демодулятора сигнал не отличался от сигнала на входе генератора ПСП. Кодирование не использовалось при исследовании свойств МоДемов.

При моделировании исследовалась вероятность ошибки демодулирования при дисбалансе амплитуд, расстройке фаз, расстройке частот в синтезаторе частот демодулятора для двух описанных Модемов. Исследования показали, что дисбаланс амплитуд и расстройка частоты синтезатора частот в демодуляторе вызывают потери примерно равные для обоих видов модуляции. Расстройка фаз синтезатора частот вызывает потери при КАМ-16 в разы большие, чем при ФМ-4, причём потери намного больше чем при дисбалансе амплитуд.

Литература

1. Зубарев Ю.Б., Кривошеев М.И., Красносельский И.Н. Цифровое телевизионное вещание. Основы, методы, системы. — М.: Научно-исследовательский институт радио (НИИР), 2001. — 568 с.

УДК 621.396.67

Д.С. Клыгач

86kds@mail.ru

Южно-Уральский государственный университет

Турникетная антенна для вещания цифрового телевидения

Концепция федеральной целевой программы «Развитие телерадиовещания в Российской Федерации на 2009–2015 гг.», утверждённая распоряжением Правительства Российской Федерации от 21 сентября 2009 г №1349-р, предусматривает модернизацию сети телерадиовещания Российской Федерации, включая переход на цифровое вещание [1]. Реализация проекта частотно-территориального плана развёртывания покрытия цифрового телевизионного вещания требует замены передающих антенн на существующих в настоящее время передающих радиотелевизионных станциях. Потребность в новых антеннах непосредственно в РФ составляет несколько тысяч штук.

Цель настоящей работы заключается в разработке оригинальной передающей антенны для диапазона частот 470 — 860 МГц, предусмотренного для цифрового телевизионного вещания, обладающей преимуществами по сравнению с известными антеннами в отношении

электрических и эксплуатационных характеристик и трудозатрат на их изготовление. На конструкцию антенны автором подана заявка на патент РФ на изобретение №2009133771 от 08.09.2009.

Для обширной территории России, естественно, требуются антенны с круговой диаграммой направленности в горизонтальной плоскости. Другим важным параметром для антенны является высокий уровень согласования антенны с фидером.

В качестве аналога предложенного технического решения рассмотрены щелевые антенны, разработанные Б.В. Брауде [2] в нашей стране и антенны, предложенные R. W. Masters [3, 4]. Указанные антенны, разработанные для метрового диапазона волн, эксплуатируются на передающих радиотелевизионных станциях в областных центрах и крупных городах РФ [5]. Известны другие варианты турникетных антенн. Их общим недостатком является узкий диапазон согласования антенны с фидером. Такие антенны настраиваются на определённый телевизионный канал.

Предложенная широкополосная щелевая антенна содержит множество пар проводящих пластин, каждая пара пластин расположена в одной из N плоскостей, при этом множество N плоскостей образует пучок плоскостей с вертикальной осью. Пластины гальванически соединены друг с другом с образованием N щелей. Щели возбуждаются таким образом, что диаграмма направленности в горизонтальной плоскости по форме близка к окружности. На рис. 1 приведены расчётные диаграммы направленности для двух и трёх пар пластин, соответственно.

Для расчёта диаграмм направленности антенны использовано строгое решение задачи дифракции плоской волны на идеально проводящей полосе и приближённое решение, реализованное путём численного решения уравнений Максвелла в интегральной форме в известной программе CST Studio. На рис. 2 приведён график реальной части входного сопротивления щелевого излучателя, когда щель согласована с фидером. На графике виден участок, где сопротивление постоянно в полосе частот.

В целом это обеспечивает широкополосное согласование с коэффициентом стоячей волны по напряжению не более 1,2. На основании приведённых результатов можно сделать вывод о том, что с помощью данной антенны можно обеспечить вещание цифрового телевидения в широком диапазоне частот. График зависимости КСВ от частоты представлен на рис. 3. Диаграмма направленности в горизонтальной плоскости представлена на рис. 4.

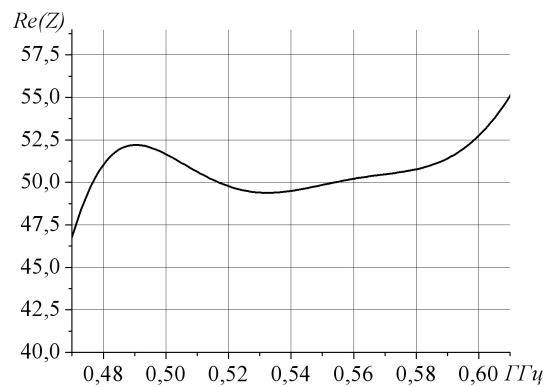


Рис. 1

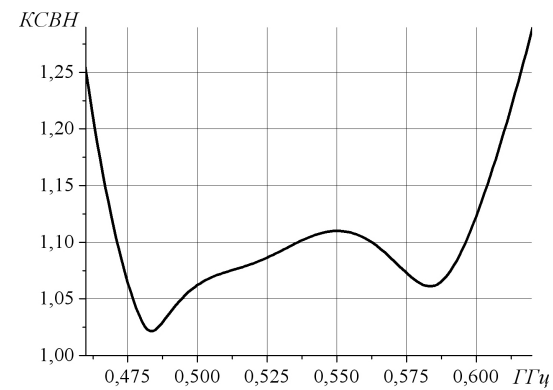


Рис. 3

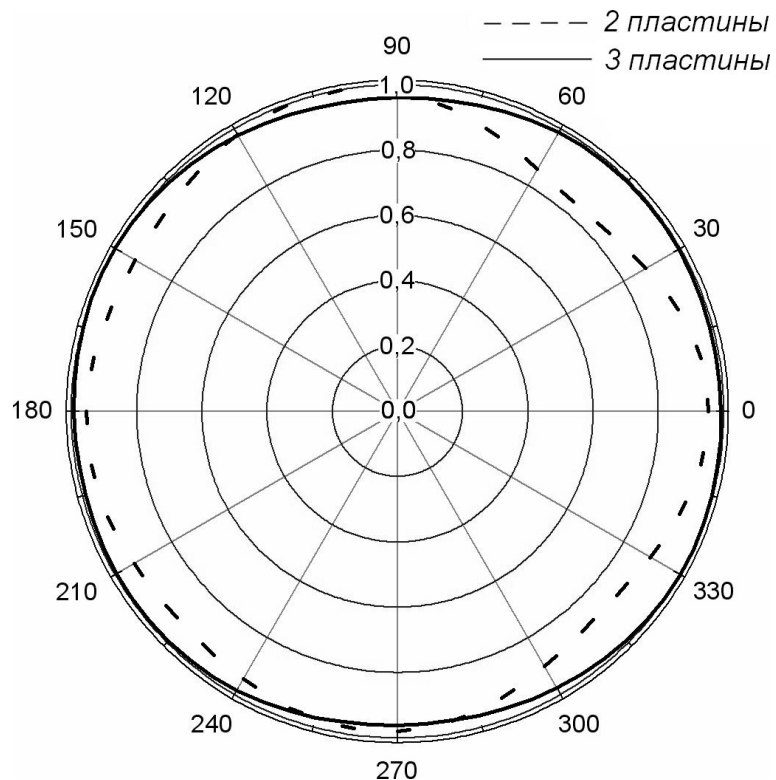


Рис. 2

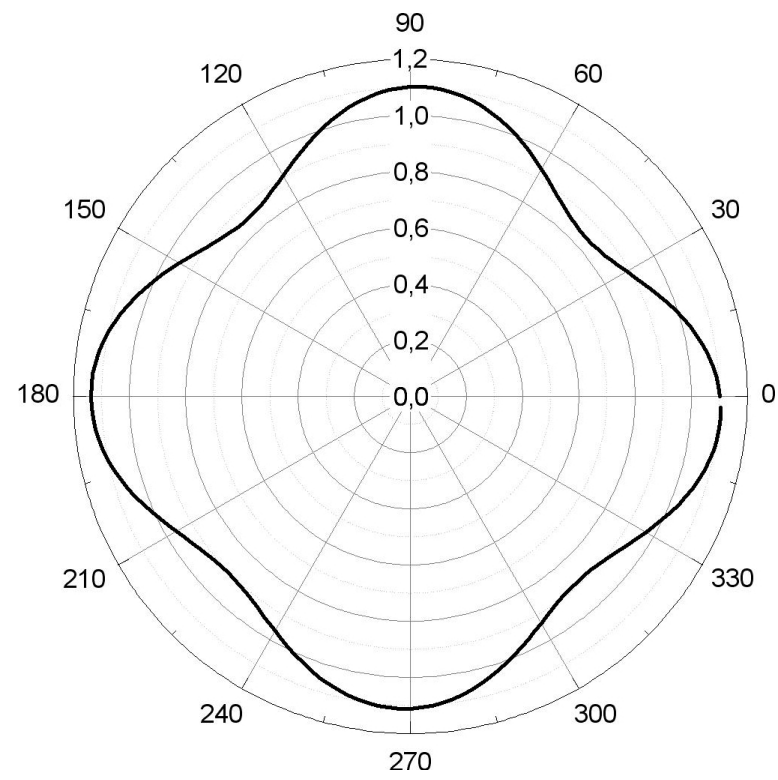


Рис. 4

Направления дальнейших работ — исследование влияния технологических и эксплуатационных факторов на параметры антенны. С этой целью автор намерен развить предложенное им приближённое решение задачи дифракции цилиндрической волны на импедансной полосе применительно к задаче проектирования щелевых антенн, разработать соответствующее программное обеспечение для расчёта параметров щелевой антенны с учётом импедансных свойств полосы. Выполнить синтез антенны с учётом влияния обтекателя по заданному коэффициенту отражения волны в диапазоне частот. Получить экспериментальное подтверждение расчётных результатов.

Литература

1. Концепция федеральной целевой программы «Развитие телерадиовещания в Российской Федерации на 2009–2015 гг.» www.minsvyaz.ru.
2. Брауде Б.В. Плоскостной вибратор. Авторское свидетельство СССР № 69974. Приоритет 12.04.1946. Оpubл. 31.12.1947.
3. Masters R.W., Erlton N.J. Antenna.US patent 2480153 США. — Заявл. 27.01.45. — Оpubл. 30.08.49.
4. Masters Robert W., Erlton N.J. Antenna. US patent. 2480154. Заявл. 27.01.45. — Оpubл. 30.08.49.
5. Трусканов Д.М. Развитие отечественной техники передающих телевизионных антенн // Сборник «Антенны». — М: Связь, 1967. — № 2. — С. 5.

УДК 621.391

А.С. Лутохин^{1,2}, Н.А. Сущенко^{1,2}

sanya-rt@rambler.ru, sckol@yandex.ru

¹ Московский физико-технический институт
(государственный университет)² Научно-исследовательский институт радио

Сенсорные сети и терминалы

Сенсорная сеть является частью единой инфокоммуникационной среды [1]. Как правило, сенсорная сеть состоит из большого количества распределённых в пространстве и взаимодействующих друг с другом устройств типа сенсор, и позволяет получать пользователю информацию об окружающей среде и обстановке. Сенсорные сети являются частным случаем беспроводных самоорганизующихся сетей (БСС), называемых в англоязычной литературе ad hoc сетями [2]. Такие сети не имеют постоянной инфраструктуры, каждый узел БСС способен функционировать как транзитное или оконечное устройство. БСС сохраняет работоспособность при выходе из строя отдельных узлов [3].

Сенсор — достаточно миниатюрные устройства, способные измерять физические параметры среды, такие, как температура, давление, влажность, концентрация различных веществ, обрабатывать полученные данные и передавать их по беспроводному интерфейсу какому-либо соседнему устройству. (В области покрытия радиосигнала каждого сенсора должен находиться как минимум ещё один сенсор, который и называется соседним.) Сенсоры могут классифицироваться по роли в сети, привязке к географическому положению, продолжительности функционирования, виду контролируемых объектов.

На архитектуру сенсорных сетей естественным образом накладываются следующие ограничения:

- устойчивость при неисправности;
- масштабируемость;
- беспроводной интерфейс;
- малое потребление энергии;
- равнозначность узлов;
- одновременное функционирование большого количества узлов на ограниченном пространстве;

— низкая стоимость узлов.

В работе предлагается способ повышения эффективности сенсорных сетей за счёт разработки сенсорных терминалов. Сенсорные сети взаимодействуют с человеком посредством сенсорных терминалов. Сенсорные терминалы получают информацию от сенсорных сетей и индивидуализируют её в зависимости от положения пользователя в пространстве, его физического состояния (возраст, состояние здоровья, усталость и т. д.) [4].

Количество приложений сенсорных сетей практически неограничено. Эти приложения условно можно разделить на несколько основных категорий:

- медицина;
- безопасность, чрезвычайные ситуации (ЧС), военные операции;
- фабрики, заводы, здания;
- сельское хозяйство;
- «умный дом»;
- транспортные системы и автомобили.

Высокий прикладной потенциал сенсорных сетей в настоящем и, как следствие, большая популярность в будущем позволяют говорить о том, что сенсорные сети и терминалы приведут к новому витку информатизации общества, получившему название и-общества.

Литература

1. *Сарьян В.К.* Входные потоки в сетях массового обслуживания // Электросвязь. — 2008. — № 4.
2. *Кучерявый Е.А., Молчан С.А., Кондратьев В.В.* Принципы построения сенсоров и сенсорных сетей // Электросвязь. — 2006. — № 6. — С. 10–15.
3. *Молчанов Д.А., Кучерявый Е.А.* Приложения беспроводных сенсорных сетей // Электросвязь. — 2006. — № 6. — С. 20–22.
4. *Кучерявый А.Е., Парамонов А.И.* Модели трафика для сенсорных сетей в и-России // Электросвязь. — 2006. — № 6. — С. 15–18.

УДК 004.032

О.С. Однороженко

olya.odn@gmail.com

Научно-исследовательский институт радио
Московский физико-технический институт
(государственный университет)

Оптимизация методов помехоустойчивого кодирования в системах подвижной спутниковой связи

Одна из самых весомых задержек связана с помехоустойчивым кодированием и декодированием данных. Для снижения этого фактора постоянно необходимо решать проблемы совершенствования существующих методов.

Помехи в спутниковой связи бывают разной природы (атмосферные, промышленные, помехи от посторонних радиостанций и каналов).

В настоящее время разрабатывается система «Арктика» для обеспечения связью удалённых территорий, находящихся в северной части Российской Федерации (включая арктические территории, её воздушное пространство). В Арктике сосредоточено 90% извлекаемых ресурсов углеводородов всего континентального шельфа России. Поэтому тема ускоренного хозяйственного освоения Арктического бассейна в настоящее время в условиях постепенного сокращения запасов природных ресурсов становится все более актуальной.

Организация круглосуточной непрерывной связи на заданной территории возможна только на базе использования наклонных высокоэллиптических орбит. Одним из главных пунктов является обеспечение подвижной связи в данном регионе. Комплекс подвижной спутниковой связи в L -диапазоне должен обеспечивать:

- президентскую и правительственную связь;
- подвижную специальную связь;
- подвижную связь в интересах управления воздушным движением.

Для проектирования бортового модулятора важно учитывать местность обслуживаемой зоны. Как известно, Арктика отличает-

ся экстремальными природно-климатическими условиями (многолетняя мерзлота, постоянные ледовый покров или дрейфующие льды в арктических морях). Растительность арктической тундры преимущественно травянистая, в северной части России деревья в основном низкорослые. Необходимо рассмотреть влияние этих лесных массивов на передачу сигнала от спутника к автомобилю. Вносимые потери очень сильно варьируются в зависимости от типа деревьев, времени года, плотности насаждений, погодных условий. По приближенным расчётным формулам можно определить, что при движении автомобиля по трассе в лесу, потери превышают 12 дБ, при насаждениях вдоль трассы до 8,5 дБ. Данные помехи можно устранить энергетическим методом, что является затратным, или помехоустойчивым кодированием. В данной задаче предпочтение отдается второму методу.

Разработка помехоустойчивого кодирования решает компромиссную задачу «сложность–эффективность». На программно-аппаратном уровне, решение этой задачи приводит к необходимости как алгоритмического, так и технического упрощения, включающего выбор наименее сложной реализации алгоритма.

УДК 004.08

Я.Г. Подкосова

yanapdk@rambler.ru

Московский физико-технический институт
(государственный университет)

Научно-исследовательский институт радио

Современные способы хранения информации на машинно-читаемых носителях

Постановка задачи. При переходе на электронный документооборот вопросы долговечности сохранности информации принимают с учётом их сильнейшего влияния на экономическую, социальную, политическую и другие сферы жизни общества, значение первостепенной важности.

В связи с этим в докладе проводится анализ возможностей использования различных типов носителей для обеспечения долговременного хранения информации и обзор их сравнительных характеристик.

Объем информации, представленной в цифровом виде, стремительно растёт. Задача сохранения информации является одной из приоритетных задач общества, и при её решении необходимо обеспечить для будущих поколений как сохранность уже накопленных знаний, так и новой информации. Существуют виды информации, в том числе научно-технической, для которой трудно указать сроки, когда эта информация теряет значимость или становится ненужной. Во многих случаях ценность информации со временем только возрастает. Объемы информации, представленной в цифровой форме, увеличиваются не только за счёт новой информации, но и за счёт перевода в цифровую форму ранее созданных информационных ресурсов.

Представление информации в цифровом виде позволяет решить ряд проблем хранения информации, создать совершенно новые возможности для доступа к информации и её обработки. Перевод документов в цифровую форму помогает решить ряд проблем длительного хранения, а именно: обеспечить возможность контролирования документов без потери качества, устранения дефектов на носителях с аналоговой формой представления (царапины на видеоматериалах, шумы на аудиозаписях и др.), мультимедийного представления информации.

Однако создание огромных архивов информационных материалов в цифровой форме привело к появлению ряда проблем, которые требуют решения в ближайшее время: создание надежных носителей для долговременного хранения цифровой информации, разработка и внедрение специальных форматов для записи информации, подлежащей долговременному хранению, разработка специальных устройств считывания информации. Несмотря на важность и сложность задач по решению проблемы быстрого морального старения программно-обеспечения, а также технического и морального старения систем считывания информации, важнейшей проблемой является создание носителей для долговременного хранения цифровой информации.

Обзор способов хранения информации. Стандартные компакт-диски. Носители, предназначенные для массового использования. Срок хранения информации на стандартных компакт-дисках составляет не более 20–30 лет. Этот срок хранения определяется особенностями технологии изготовления носителей, предназначенных для

массового использования. Срок хранения информации на стандартных компакт-дисках ограничен из-за нестабильности свойств и недостаточной механической прочности подложек из поликарбоната, а также ряда других причин.

Магнитные ленты. Для архивного хранения информации в цифровой форме в настоящее время наиболее широко используются магнитные ленты. Накоплен достаточно большой опыт сохранения аудио- и видеоинформации на магнитных носителях. Существуют образцы магнитных лент с аналоговыми аудиозаписями, на которых информация сохраняется более 50 лет. За последние десятилетия осуществлены значительные усовершенствования систем магнитной записи на ленточные носители, однако гарантированный срок хранения информации на лентах практически не изменился — (30–40) лет.

Оптические диски. Среди разработанных к настоящему времени технологий наиболее полно удовлетворяют требованиям по созданию носителей для долговременного хранения оптические методы записи и хранения информации. Оптические носители обладают рядом особенностей, которые позволяют рассматривать их как перспективные носители для долговременного хранения информации: бесконтактное считывание информации, использование физических методов защиты записанной информации, высокая плотность записи, возможность увеличения плотности и скорости записи информации, и некоторые другие.

Перспективы дальнейшего усовершенствования характеристик носителей с целью увеличения их долговечности. Существенно повысить надежность хранения информации по сравнению со стандартными компакт-дисками (CD, DVD) и носителями типа WORM (оптические носители с локальным удалением материала поглощающего слоя на участках записи сфокусированным лазерным излучением) представляется возможным за счёт изготовления носителей с более стабильными во времени характеристиками, способными выдерживать колебания температур и влажности в существенно более широком диапазоне, чем носители с поликарбонатными подложками. Эти носители изготавливаются с использованием технологий, применяемых в производстве компакт-дисков. Носители могут рассматриваться как расширение «семейства» компакт-дисков со специфической областью применения. При необходимости может производиться замена защитных слоёв (реставрация носителя). Так, медно-никелевые штампы для тиражирования грампластинок, изготовленные 60–70 лет назад, прекрасно сохранились и аудиоинформа-

ция с них воспроизводится с высоким качеством звучания. Срок хранения информации на металлических носителях может составлять сотни лет. Таким образом, сроки хранения информации на оптических носителях могут быть существенно увеличены (до 200–300) лет при использовании высокостабильных материалов для подложек носителей и однородных структур для представления информации.

Литература

1. Петров В.В., Крючин А.А., Шанойло С.М., Косско И. О., Кравець В.Г. Способи вирішення проблеми довгострокового зберігання інформації, записаної у цифровому вигляді // Доповіді Національної академії наук України. — 2003. — № 4. — С. 52–58.
2. Hedstrom M. Digital preservation: a time bomb for digital libraries // <http://www.uky/kierman/DL/hedstrom.html>.
3. Фрадкин В. Прошлое, настоящее и будущее носителей информации // <http://www.dw-world.de>.
4. Михайлов О.А. Электронные документы в архивах. — М.: Диалог. — МГУ, 2000.

УДК 004.9

В.К. Сарьян¹, А.С. Лутохин^{2,1}, Н.А. Сущенко^{2,1}
sarian@niir.ru, sanya-rt@rambler.ru, sckol@yandex.ru

¹ Научно-исследовательский институт радио

² Московский физико-технический институт
(государственный университет)

Эффективные способы расширения абонентской базы и номенклатуры инфокоммуникационных услуг

На современном этапе развития инфокоммуникационных сетей предоставления массовых услуг расширение абонентской базы и номенклатуры предоставляемых услуг является основной стратегической задачей операторов и контент-провайдеров. Наиболее развитые сети, такие, как сотовая сеть и сеть широкополосного доступа в Интернет, уже практически достигли своего предельного уровня раз-

вития в плане предоставления инфокоммуникационных услуг. Это связано с тем, что современные технологические решения имеют серьёзные ограничения, связанные с используемыми ресурсами: вычислительные способности мобильных терминалов, ресурс их батарей и ограниченность радиочастотного спектра. В связи с этими ограничениями дальнейшее расширение абонентской базы и номенклатуры предоставляемых услуг представляется невозможным без серьёзных технологических прорывов и модернизации многочисленного оборудования, что требует больших затрат. Кроме того, малые размеры мобильных устройств не позволяют реализовать удобный интерфейс пользователя для предоставления сложных инфокоммуникационных услуг по причине недостаточных размеров дисплеев терминалов [1].

Предлагается производить расширение абонентской базы и номенклатуры предоставляемых услуг за счёт адаптации уже существующих инфокоммуникационных услуг к стационарным терминалам, а также за счёт разработки новых услуг для стационарных терминалов с учётом превосходства их по техническим параметрам над мобильными терминалами. Предлагаемые способы расширения абонентской базы и номенклатуры предоставляемых операторами услуг позволяют обойти перечисленные выше ограничения, связанные с ограниченностью ресурсов, малыми вычислительными способностями и габаритами терминалов. Поскольку применение стационарно расположенных терминалов позволяет использовать проводные соединения с сетью питания, сетью инфокоммуникаций (в этом случае пропускная способность канала связи значительно выше) и дополнительными устройствами. Данный способ позволяет организовать массовый доступ к более сложным инфокоммуникационным услугам, поскольку стационарные терминалы могут значительно превосходить мобильные не только по аппаратным возможностям, но и по удобству интерфейса пользователя без ущерба простоте использования терминала.

Кроме того, применение стационарных терминалов позволяет расширить номенклатуру услуг предоставляемых мобильными терминалами, находящимися в окрестности стационарного терминала. Например, если установить короткодействующий широкополосный радиоканал связи мобильного терминала со стационарным, то это позволит обойти ограничения функциональности мобильных терминалов, связанные с малой шириной полосы передачи данных, что может быть использовано при оказании услуг.

Представленные способы расширения абонентской базы и номенклатуры предоставляемых услуг являются весьма перспективными,

поскольку уже возможна их практическая реализация. Разработанная в НИИ радиотелевизионная приставка НИИР ЦТВ [2] может быть использована в качестве стационарного терминала, о котором говорилось выше. Этот терминал имеет возможности подключения к сетям Ethernet, имеет стандартные порты расширения для подключения внешних устройств (клавиатуры, принтера флэш-накопителя и пр.) и возможности для предоставления массовых инфокоммуникационных услуг, в том числе защищённой передачи информации. Расширение абонентской базы операторов возможно в этом случае без значительных инвестиций — за счёт абонентской базы телевизионных зрителей, которые смогут использовать приставку не только для просмотра цифрового телевидения, но и для получения инфокоммуникационных услуг. Примерами таких услуг может служить: оплата коммунальных платежей, заказ товаров в интернет, IPTV, IP-телефония, обмен данными, сбор статистики предпочтений телезрителей (для телевизионных каналов) и т. д.

Таким образом, поскольку стационарные терминалы обладают как некоторыми преимуществами перед мобильными, так и некоторыми недостатками, мы предполагаем, что и для мобильных, и для стационарных терминалов найдутся свои ниши в сфере предоставления массовых инфокоммуникационных услуг. Мобильный и стационарный терминалы могут эффективно дополнять друг друга без конкуренции между собой за приоритеты пользователя. В перспективе развития технологий представляется, что каждый абонент будет иметь и мобильный и стационарный терминал с идентичными интерфейсами и наборами услуг.

Литература

1. *Сарьян В.К., Лутохин А.С., Шшишкин А.Л.* Эффективный способ расширения абонентской базы и номенклатуры предоставляемых инфокоммуникационных услуг // Материалы 3-й международной конференции «Конвергенция телекоммуникационных сетей и услуг в России». — 2009.
2. *Бутенко В.В., Назаренко А.П., Сарьян В.К.* Внедрение многофункциональной мультимедийной интеллектуальной телевизионной приставки и/или телевизионного приёмника. — залог успешного развития эфирного телевизионного вещания // Broadcasting. Телевидение и радиовещание. — 2008. — № 6.

УДК 004.05

В.К. Сарьян¹, А.С. Лутохин^{2,1}, Н.А. Сущенко^{2,1}

sarian@niir.ru, sanya-rt@rambler.ru, sckol@yandex.ru

¹ Научно-исследовательский институт радио² Московский физико-технический институт
(государственный университет)

Применение теории типовых процессов для оценки эффективности использования ограниченных ресурсов

Переход в мировом масштабе телерадиовещания от аналогового к цифровому стандарту существенно ускорил процессы конвергенции традиционных и новых сетей связи и телерадиовещания и способствовал формированию единой инфокоммуникационной среды (ИКС) как материальной составляющей информационного общества (ИО) [1, 2]. В системном плане ИО в настоящее время правомерно представлять в виде сложной открытой организационной динамической системы, включающей большое, но конечное число регулярно информационно взаимодействующих человеко-машинных объектов [3].

Однако эволюция ИО в развитых странах, как только ИКС позволила реально обеспечить доступ массового пользователя к ИК-услугам, начала сопровождаться возникновением новых ограничений, которые хотя и не могут ещё повлиять на скорость эволюции, но заметно снижают положительный эффект от её внедрения, и поэтому требуют скорейшего нахождения эффективного решения, так как стандартные методы приводят к исчерпанию природных ресурсов: частотного спектра, объёмов энергопотребления [4] и, наконец, физических и психических ресурсов (здоровья) людей.

Основная причина этих проблем состоит в том, что при обеспечении массового доступа к ИК-услугам оказался плохо проработан вопрос о порядке формирования предоставляемых услуг и контроля за качеством их предоставления. Существующие структуры и нормативно-правовая база до сих пор ориентированы только на контроль качества коммуникационной составляющей предоставляемых услуг, они не в состоянии контролировать содержательную (информацион-

ную) часть услуги, которая собственно является, как правило, основной при предоставлении ИК-услуг.

Как и в случае цепей поставок товаров, для ЦПУ можно выделить две задачи оценки эффективности:

1. Внешняя оптимизация: для каждого объекта (в данной работе мы будем называть его внешним оператором или оператором ИКС) из данного набора требуется определить эффективность его функционирования для распределения ограниченных ресурсов между внешними операторами. Примером задачи является выделение государственным регулятором лицензий на виды деятельности, связанные с использованием радиочастотного спектра.

2. Внутренняя оптимизация: для заданного множества объектов (внутренних операторов) нужно определить подмножество, требуемое для построения системы, которая функционировала бы наиболее продуктивно при заданных выделенных ресурсах. Примером задачи является выбор операторов связи, контент- и сервис-провайдеров для предоставления инфокоммуникационной услуги.

На кафедре МФТИ при ФГУП НИИР в настоящее время разрабатывается теория типовых процессов (ТП). Типовой процесс — это инвариант используемых способов удовлетворения какой-либо базовой потребности системы (технического комплекса, отдельного человека, группы людей и т. д.). Нами был разработан способ оценки эффективности осуществления типового процесса (ОЭТП) является выделение услуг, необходимых для осуществления этого типового процесса и нахождение эффективности предоставления каждой услуги по отдельности [5]. Однако оптимизация ТП может включать в себя не только модификацию услуг, но и совместную реорганизацию информационных, транспортных и производственных процессов. В связи с этим для нахождения более общего способа оценки ОЭТП в настоящее время авторами разрабатывается модель, охватывающая довольно широкий класс наиболее фундаментальных ТП. Этот подход может быть, позволит выработать методику, согласно которой будет возможно распределение ограниченных ресурсов в пользу процессов, необходимых для выживания человеческого общества, что позволит снизить темпы исчерпания как природных и материальных ресурсов, так и психических и физических ресурсов человека [1].

Литература

1. *Бутенко В.В., Назаренко А.П., Сарьян В.К.* Проблемы современного этапа и пути дальнейшего развития информационного общества // Труды НИИР. — 2009. — № 3.
2. *Бутенко В.В., Назаренко А.П., Сарьян В.К.* Создание и стандартизация инфраструктуры предоставления и системы администрирования массовых инфокоммуникационных услуг в сфере навигационной деятельности. — первоочередная задача по реализации ФЗ «О навигационной деятельности» // Третья Всероссийская конференция «Фундаментальное и прикладное координатно-временное и навигационное обеспечение». — СПб., 2009.
3. *Сарьян В.К.* Входные потоки в сетях массового обслуживания // Электросвязь. — 2008. — № 4.
4. *Назаренко А.П., Сарьян В.К.* Новый подход к развитию инфокоммуникационных технологий. Заметки с выставки СеВIT // Электросвязь. — 2008. — № 5.
5. *Сарьян В.К., Сущенко Н.А.* Использование Data Envelopment Analysis (DEA) для расчёта эффективности использования частотного спектра в инфо-коммуникационной среде (ИКС) // Труды 51-й научной конференции МФТИ. — 2008.

УДК 004

Ф.А. Слесаренко

slesarenkof@gmail.com

Московский физико-технический институт
(государственный университет)

Научно-исследовательский институт радио

Одночастотные сети как эффективный способ внедрения цифрового наземного телевизионного вещания

Как известно, Российской Федерации предстоит переход к цифровому наземному телевизионному вещанию. В соответствии с «Концепцией внедрения в России цифрового наземного телевизионного и радиовещания», утвержденной правительством, к 2015 г. все телевидение в стране должно быть только цифровым.

Среди множества систем наземного цифрового ТВ в нашей стране была выбрана европейская система DVB-T. Особенностью этой системы является возможность построения не только привычных многочастотных вещательных сетей, но и одночастотных.

Одночастотная сеть (Single Frequency Network, SFN) — сеть передающих станций, в которой вещание ведётся на одном частотном канале. Данная технология позволяет использовать множество передатчиков, покрывающих смежные области, работающих на одной частоте и передающих одинаковые программы. При этом использование всего одного частотного канала происходит без существенного взаимного влияния передатчиков друг на друга, тем самым значительно экономится частотный ресурс, проблема нехватки которого особенно остро возникает в период перехода от аналогового к цифровому вещанию.

Стандарт DVB-T предлагает наиболее эффективные средства для построения одночастотных сетей, что является одним из его важнейших преимуществ, позволивших ему получить широкое распространение. Одночастотные сети предоставляют гибкость в выборе топологии вещательной сети. Для обеспечения лучшей зоны покрытия возможно использование маломощных передатчиков в дополнение к мощным источникам сигнала, что позволяет обеспечить прием в тех

местах, где он затруднен или просто невозможен (например, низины ландшафта, местность за холмом или высотным зданием).

Одночастотные сети цифрового наземного телевидения широко распространены в мире и представлены в таких странах, как Великобритания, Германия, Испания, Франция и др.

В связи с предстоящим переходом РФ к цифровому вещанию успешный опыт использования одночастотных сетей в Германии для построения национальной вещательной сети представляет особый интерес. Внедрение цифрового телевидения в Германии началось в 2002 году, а уже к концу 2008 года им было охвачено более 90% населения. Причём с самого начала вещание велось при помощи одночастотных сетей.

Для примера, вещательная сеть региона Бавария состоит из более чем 25 передатчиков, большинство из которых ранее использовались для аналогового телевидения. Каждым источником сигнала ведётся вещание на 3–5 частотных, что позволяет транслировать от 15 до 25 телевизионных каналов. Все передатчики разделены между 10 одночастотными сетями, причём в соседних одночастотных сетях вещание ведётся на различных частотных каналах.

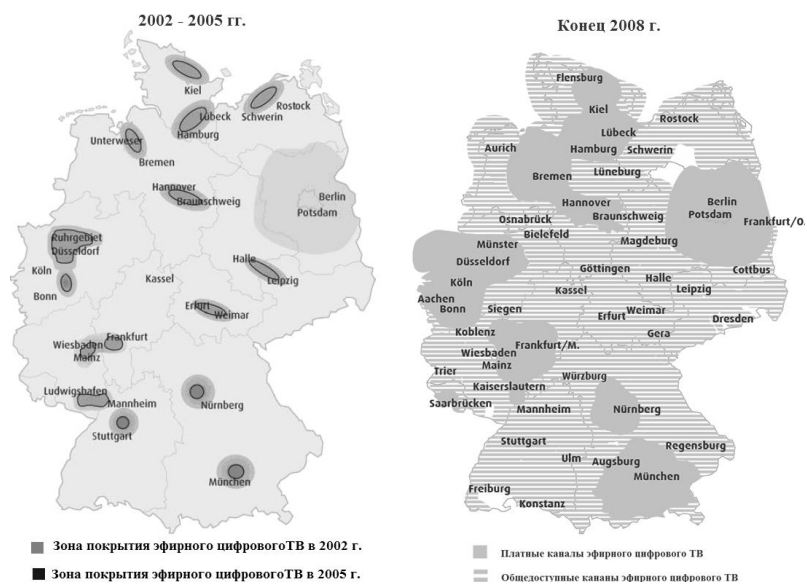


Рис. 1. Развитие эфирного цифрового ТВ в Германии

Многолетний опыт использования одночастотных сетей цифрового наземного телевидения подтверждает тот факт, что данные сети являются эффективной и надежной заменой привычных многочастотных сетей. Использование одночастотных сетей для развертывания в РФ наземного цифрового ТВ представляется предпочтительным. В таком случае для вещания можно использовать существующие передатчики, однако построение таких сетей потребует как внедрения дополнительных источников сигнала, так и тщательного планирования таких сетей (рис. 1).

Литература

1. *Зубарев Ю.Б., Кривошеев М.И., Красносельский И.Н.* Цифровое телевизионное вещание. — М.: НИИР, 2001.
2. *Fisher W.* Digital Television. A Practical Guide for Engineers. — Berlin: Springer Verlag, 2004. — 570 p.

УДК 004.032

А.С. Тимофеев

tas@niir.ru

Научно-исследовательский институт радио

Перспективы развития систем спутниковой связи и вещания

Космическая группировка спутников связи и вещания является составной частью системы спутниковой связи и вещания, предназначенной для решения задач государственного управления, обеспечения потребностей коммерческих структур, а также населения страны в современных инфотелекоммуникационных услугах.

В целях реализации поручения Правительства Российской Федерации от 13 августа 2008 года Научно-исследовательским институтом радио разработана Концепция развития российской космической группировки спутников связи и вещания гражданского назначения на период до 2020 года (далее — «Концепция...»).

Цель создания концепции: повышение эффективности функционирования российской космической группировки спутников связи и вещания и определение путей её дальнейшего развития.

Основные задачи, решённые в «Концепции...»: — определение потребностей государственных структур и рынка в услугах связи и вещания; — выработка предложений по облику космических аппаратов; — обоснование вариантов развития космической группировки; — оптимизация технико-экономических показателей развития космической группировки.

В «Концепции...» развитие космической группировки рассмотрено комплексно, совместно с земным сегментом системы спутниковой связи и вещания, с учётом мирового опыта и особенностей, характерных для России: — большие пространственные масштабы территории; — наличие регионов с низкой плотностью населения; — различие в платежеспособности населения различных регионов.

Следует отметить, что регионально-территориальные особенности Российской Федерации способствуют дальнейшему увеличению роли систем спутниковой связи и вещания в общей структуре отрасли телекоммуникаций.

Критерий эффективности, обеспечивающий интегральную оценку развития космической группировки, определён на основе двух показателей качества: полнота удовлетворения потребностей в услугах связи и вещания и экономическая эффективность при обязательном выполнении следующих необходимых условий: — преимущество обслуживания существующих абонентских сетей; — сохранение орбитально-частотного ресурса; — согласованное развитие космического и земного сегментов систем спутниковой связи и вещания; — поддержка отечественной промышленности.

Проведённые оценки показали рост требований к пропускной способности российской космической группировки: к 2015 году — в 1,5 раза, к 2020 году — в 2,5 раза.

Мировые тенденции создания космических аппаратов показывают увеличение количества создаваемых космических аппаратов тяжелого класса. Ежегодно количество выводимых на орбиту космических аппаратов тяжелого класса составляет больше 50%.

В последние годы наблюдается устойчивая тенденция уменьшения количества космических аппаратов фиксированной и вещательной спутниковых служб и ввод в эксплуатацию специализированных космических аппаратов, предназначенных для обеспечения широкополосного доступа и подвижной связи.

В составе государственной космической группировки спутников связи и вещания в настоящее время эксплуатируется 4 космических аппарата за пределами гарантийного срока активного существования

на орбите, а также один космический аппарат с ограничениями по функциональности («Экспресс-АМ2», 80€ в.д.).

До завершения 2013 года в составе космической группировки будет заменено 8 космических аппаратов, и выведен один космический аппарат в не занятую в настоящее время орбитальную позицию 145€ в.д.

Разработанная Программа развития космической группировки спутников связи и вещания обеспечивает наращивание её возможностей за счёт создания перспективных космических аппаратов, построенных на передовых технических решениях и обладающих высокими эксплуатационными показателями. Это космические аппараты «Экспресс-АМУ6», «Экспресс-АМУ7», «Экспресс-АТ3», а также космические аппараты обеспечения широкополосного доступа «Экспресс-АМУ3/4/5». Кроме того, в программный период будет создана отечественная группировка космических аппаратов на высокоэллиптической орбите «Экспресс-РВ» в рамках создания многоцелевой космической системы «Арктика».

Космические аппараты «Экспресс-АМУ3», «Экспресс-АМУ4» будут строиться на отечественной высокоэнергетической платформе, обеспечивающей мощность для полезной нагрузки не менее 18 кВт и срок активного существования не менее 17 лет. Структурное построение полезной нагрузки космического аппарата позволит адаптировать её возможности к изменяющимся потребностям рынка услуг.

Космический аппарат «Экспресс-АТ3» за счёт обработки и коммутации сигналов на борту обеспечит предоставление абонентам современных интерактивных мультимедийных сервисов.

Серия из трёх космических аппаратов «Экспресс-АМУ3/4/5» предназначена для предоставления на всей территории широкополосного доступа, причём экономические показатели спутниковой системы позволят ей эффективно конкурировать с аналогичными наземными системами.

Кроме развития космической группировки на геостационарной орбите в рамках создания многоцелевой космической системы «Арктика» будет осуществлен ввод в эксплуатацию группировки спутников связи на высокоэллиптической орбите типа «Тундра». Данная группировка позволит создать современную инфотелекоммуникационную инфраструктуру в Арктическом регионе.

Программные показатели развития государственной космической группировки позволят к 2015 году провести пропускную способность орбитальной группировки в соответствии с потребностями органов

государственного управления и социальной сферы в ресурсах спутниковых систем связи и вещания.

УДК 621.376.9

А.Л. Шишкин

sisoid@frtk.ru

Московский физико-технический институт
(государственный университет)
Научно-исследовательский институт радио

Исследование влияния ширины полосы пропускания ортогонального полосового фильтра на взаимные помехи отдельных подканалов в радиоканале с ортогональным частотным мультиплексированием

Метод ортогональной полосовой фильтрации (ОПФ) является одним из способов переноса цифрового ВЧ-сигнала, передаваемого методом ортогонального мультиплексирования, на другую несущую. Задача переноса информационного сигнала на новую несущую частоту неразрывно связана с проблемой оптимальной фильтрации переносимого сигнала.

С одной стороны, должны быть минимизированы искажения переносимого сигнала, с другой — в спектральной полосе преобразованной цифровой несущей должны быть максимально подавлены уровни на частотах соседних подканалов. Как искажения сигнала с ОПФ, так и недостаточно подавленные уровни соседних подканалов приводят к потерям при передаче. В первом случае потери наблюдаются в самом канале с ОПФ, а во втором — в соседних по отношению к нему подканалах.

Для исследования влияния ширины полосы пропускания ОПФ на взаимные помехи подканалов в системе моделирования Simulink [1] была построена модель радиоканала, имитирующая один из частот-

ных подканалов. Структурная блок-схема модели изображена на рис. 1.

Далее на основе описанной выше модели была построена схема моделирования, содержащая исследуемый подканал, а также два соседних к нему подканала с ОПФ. «Хвосты» спектров соседних подканалов попадают в спектральный интервал исследуемого подканала, ухудшая тем самым качество передачи сигнала. Было показано, что величина этого ухудшения напрямую зависит от ширины полосы пропускания ОПФ.

С помощью схемы моделирования была проведена серия экспериментов, в которых была исследована зависимость качества передачи информационного сигнала в канале с переносом частоты несущей от: полосы пропускания ОПФ, используемого типа модуляции, соотношения сигнал/шум, расстройки элементов обработки сигнала. Были найдены оптимальные значения для полосы пропускания ОПФ при фиксированных типах модуляции сигнала.

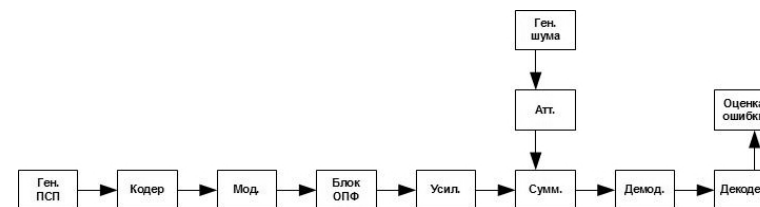


Рис. 1. Блок-схема модели радиоканала с изменением частоты несущей посредством ОПФ

Литература

1. Дьяконов В.П. Simulink 5 / 6/ 7: Самоучитель. — М.: ДМК-Пресс, 2008.

Список представленных организаций

Samsung Electronics
ЗАО «Интел А/О»
ЗАО «Московский центр SPARC-технологий»
Институт автоматизации проектирования РАН
Институт прикладной математики им. М.В. Келдыша РАН
Институт проблем передачи информации им. А.А. Харкевича РАН
Институт точной механики и вычислительной техники
им. С.А. Лебедева РАН
Институт электронных управляющих машин
Московский государственный институт электронной техники
(технический университет)
Московский государственный технический университет
им. Н.Э. Баумана
Московский государственный университет им. М.В. Ломоносова
Московский инженерно-физический институт (Национальный
исследовательский ядерный университет)
Московский технический университет связи и информатики
Московский физико-технический институт
(государственный университет)
Научно-исследовательский институт радио
ООО «Даймонд Кволити»
ООО «ИДМ»
ООО «НетКрэкер»
Особое конструкторское бюро систем автоматизированного
проектирования
Российский государственный технологический университет
им. К.Э. Циолковского

Российский университет дружбы народов
Рязанский государственный радиотехнический университет
Санкт-Петербургский государственный политехнический
университет
Санкт-Петербургский государственный университет
Южно-Уральский государственный университет
Южный федеральный университет

Именной указатель**А**

Абдрашитов В.А. 24
 Антоненко М.Н. 26
 Артамонов Д.С. 41

Б

Бабинцев А.В. 137
 Бабокин Д.Ю. 89
 Беляев А.Б. 80
 Беляев М.Г. 108
 Битнер В.А. 44
 Богданов А.Ю. 46
 Бурнаев Е.В. 108,
 111
 Бутенко В.В. 141

В

Васильев А.В. 26
 Владимиров С.М. 4
 Воробушков В.В. 49

Г

Гаврин Е.А. 83
 Грушин А.И. 71

Д

Дунаева М.А. 86

Е

Евдокимов А.В. 29

З

Золотухин М.В. 89

И

Исаев М.В. 52

К

Канев С.А. 144
 Карпова М.А. 147
 Клыгач Д.С. 149
 Кожин А.С. 55
 Королькова А.В.
 130
 Косарев Н.Е. 92
 Красносельский И.Н.
 144
 Ксендзов А.В. 113
 Кутафин А.А. 32
 Кшевецкий А.С. 13

Л

Лашин С.В. 7
 Лутохин А.С. 154,
 160, 163
 Любецкий В.А. 116,
 128

М

Макошенко Д.В. 95
 Мешков А.Н. 58
 Миллер А.Б. 118
 Москаленко И.В. 61

Н

Назаренко А.П. 141
 Некрылов Д.А. 32
 Нуждин Д.О. 99

О

Обернихин В.А. 10
 Однороженко О.С.
 156

П

Плаксина О.Н. 122
 Подкосова Я.Г. 157
 Подольская Н.А. 34
 Поляков Н.Ю. 63
 Приходько П.В. 111
 Пьянков Е.С. 125

Р

Расулов Г.М. 66
 Речистов Г.С. 101
 Рыжов М.П. 69
 Рябцев Ю.С. 49

С

Сарьян В.К. 141,
 160, 163
 Селиверстов А.В.
 116, 128
 Сенюта А.А. 13
 Слесаренко Ф.А.
 166

Смирнов С.И. 16

Смольянов П.А. 63

Сущенко Н.А. 154,
 160, 163

Сысоев И.Ю. 17

Т

Тимофеев А.С. 168
 Ткачёв С.С. 99

Х

Харичкин А.Е. 29
 Хатько Е.Е. 104
 Хропов М.С. 37

Ч

Чан Х.В. 71
 Черноиванов А.И.
 130
 Чорняк А.Д. 20

Ш

Шерстнёв А.Е. 74
 Шибаева Е.С. 134
 Шишкин А.Л. 171

Щ

Щербина Н.А. 77

Я

Ямпольский А.Д. 37

Содержание

Секция радиотехники и защиты**информации 4**

- Владимиров С.М.* Использование итеративного декодирования в сетевом кодировании 4
- Лапшин С.В.* Особенности реализации средств защиты информации от несанкционированного доступа для инфраструктуры виртуализации 7
- Обернихин В.А.* Восстановление работоспособности сетевых кластеров виртуальных серверов после вирусных атак 10
- Сенюта А.А., Кшевецкий А.С.* Оптимизация параметров интегрированной криптосистемы 13
- Смирнов С.И.* Исследование линейных регистров сдвига специального вида 16
- Сысоев И.Ю.* Одновременное использование стандартного и нормального базисов при реализации алгоритмов кодирования и декодирования рангового кода 17
- Чорняк А.Д.* Автоматизированное выделение и методы оценки сигнатур разрушающих программных воздействий 20

Секция инфокоммуникационных систем и сетей 24

- Абраштитов В.А.* Оценка эффективности систем управления работой персонала для операторов связи 24
- Васильев А.В., Антоненко М.Н.* Модели и методы оценки количественных характеристик комплекса работ в рамках проектов 26
- Евдокимов А.В., Харичкин А.Е.* Итерационные многометодные алгоритмы многокритериального ранжирования 29
- Кутафин А.А., Некрылов Д.А.* Решение задачи планирования модернизации телекоммуникационной сети с учётом технических и экономических критериев 32

- Подольская Н.А.* О задаче симуляции компьютерных сетей и методах её решения 34
- Хропов М.С., Ямпольский А.Д.* О решении задачи симуляции сетевого соединения, работающего по стандарту 802.11 37

Секция высокопроизводительных вычислительных систем 41

- Артамонов Д.С.* Методы повышения эффективности вычислительного процесса на однородных вычислительных средах 41
- Битнер В.А.* Система интерпретации промежуточного представления программы в оптимизирующем компиляторе 44
- Богданов А.Ю.* Комплексная отладка контроллера периферийных интерфейсов 46
- Воробушков В.В., Рябцев Ю.С.* Использование развязывающих конденсаторов на подложке микропроцессора «Эльбрус-S» 49
- Исаев М.В.* Объединение двух процессорных ядер с архитектурой «Эльбрус» для создания двухъядерной системы на кристалле «Эльбрус-S2» 52
- Кожин А.С.* Контроллер памяти DDR2 SDRAM и его система синхронизации в составе системы на кристалле «Эльбрус-S2» 55
- Мешков А.Н.* Разработка модели вычислительного комплекса «Эльбрус-S» 58
- Москаленко И.В.* Вопросы портирования компонент общесистемного программного обеспечения в защищенный режим микропроцессора «Эльбрус» 61
- Поляков Н.Ю., Смольянов П.А.* Разработка контроллера встроенного интерфейса AXI в составе системы на кристалле 63
- Расулов Г.М.* Конструктивно-технологические задачи создания прототипа двухъядерного микропроцессора с архитектурой «Эльбрус» 66

<i>Рыжов М.П.</i> Создание генератора тестов с самомодифицирующимися кодами для многопроцессорной системы на базе микропроцессора «Эльбрус-S»	69
<i>Чан Х.В., Грушин А.И.</i> Использование десятичной арифметики в вычислительной технике	71
<i>Шерстнёв А.Е.</i> Применение программируемых логических интегральных схем для решения задачи автоматической генерации тестовых кодов	74
<i>Щербина Н.А.</i> Разработка системного коммутатора для микропроцессора МЦСТ-4R.....	77

Секция микропроцессорных технологий 80

<i>Беляев А.Б.</i> Верификация системы поддержки транзакционной памяти для программирования многоядерных процессоров	80
<i>Гаврин Е.А.</i> Программный симулятор параллельной вычислительной системы	83
<i>Дунаева М.А.</i> Новый зарядовый усилитель считывания.....	86
<i>Золотухин М.В., Бабокин Д.Ю.</i> Преобразование графа потока управления в список инструкций	89
<i>Косарев Н.Е.</i> Исследование подходов для получения распределения потерь производительности микропроцессора с векторным счётчиком инструкций	92
<i>Макошенко Д.В.</i> Назначение переменных на регистры с помощью новых алгоритмов раскраски графа.....	95
<i>Нуждин Д.О., Ткачёв С.С.</i> Определение ориентации КА на солнце посредством солнечных батарей с использованием микроконтроллера Atmega8535.....	99
<i>Речистов Г.С.</i> Методика бинарной трансляции с динамической модификацией кода в Intel Platform Simulator	101
<i>Хатъко Е.Е.</i> Один из подходов к анализу системы тестирования сложных программных комплексов.....	104

Секция проблем передачи и обработки информации 108

<i>Беляев М.Г., Бурнаев Е.В.</i> Обобщающая способность и регуляризация в задаче аппроксимации многомерной зависимости	108
<i>Бурнаев Е.В., Приходько П.В.</i> Об одной методике бустинга для решения задач регрессии.....	111
<i>Ксендзов А.В.</i> Влияние корреляции замираний сигналов в многоантенной системе на число эффективных степеней свободы	113
<i>Любецкий В.А., Селиверстов А.В.</i> Прямые повторы в некодирующих областях хлоропластов у семенных растений	116
<i>Миллер А.Б.</i> Динамическое управление доступом к ресурсам и скоростью обслуживания при активных пользователях	118
<i>Плаксина О.Н.</i> О двух системах массового обслуживания с «прозрачными» заявками и их применении к анализу сетей мультимедиа	122
<i>Пьянков Е.С.</i> Модернизация аппаратной обработки обратной связи в сканирующем зондовом микроскопе.....	125
<i>Селиверстов А.В., Любецкий В.А.</i> Об эллипсоидах, внутри которых нет целых точек.....	128
<i>Черноиванов А.И., Королькова А.В.</i> Моделирование при помощи стохастических дифференциальных уравнений поведения TCP-трафика при взаимодействии с узлом, работающим по алгоритму RED.....	130
<i>Шибалева Е.С.</i> Анализ сетевого трафика на основе методики ARIMA	134

Секция радио и информационных технологий 137

<i>Бабинцев А.В.</i> Многолучевая зеркальная антенна для спутниковых систем связи	137
<i>Бутенко В.В., Назаренко А.П., Сарьян В.К.</i> Проблемы эволюции инфокоммуникационных сетей.....	141
<i>Канев С.А., Красносельский И.Н.</i> Системы цифрового телевизионного вещания второго поколения.....	144
<i>Карпова М.А.</i> Моделирование схем модемов с ФМ-4 и КАМ-16.....	147

<i>Клыгач Д.С.</i> Турникетная антенна для вещания цифрового телевидения.....	149
<i>Лутохин А.С., Сущенко Н.А.</i> Сенсорные сети и терминалы.....	154
<i>Однороженко О.С.</i> Оптимизация методов помехоустойчивого кодирования в системах подвижной спутниковой связи.....	156
<i>Поджосова Я.Г.</i> Современные способы хранения информации на машинно-читаемых носителях.....	157
<i>Сарьян В.К., Лутохин А.С., Сущенко Н.А.</i> Эффективные способы расширения абонентской базы и номенклатуры инфокоммуникационных услуг.....	160
<i>Сарьян В.К., Лутохин А.С., Сущенко Н.А.</i> Применение теории типовых процессов для оценки эффективности использования ограниченных ресурсов.....	163
<i>Слесаренко Ф.А.</i> Одночастотные сети как эффективный способ внедрения цифрового наземного телевизионного вещания.....	166
<i>Тимофеев А.С.</i> Перспективы развития систем спутниковой связи и вещания.....	168
<i>Шшишкин А.Л.</i> Исследование влияния ширины полосы пропускания ортогонального полосового фильтра на взаимные помехи отдельных подканалов в радиоканале с ортогональным частотным мультиплексированием.....	171
<hr/>	
Список представленных организаций.....	173
Именной указатель.....	175

ТРУДЫ 52-й НАУЧНОЙ КОНФЕРЕНЦИИ МФТИ

Современные проблемы фундаментальных и прикладных наук

Часть I Радиотехника и кибернетика

Том 1

Составитель: *Русскин С.О.*

Редакторы:

В.А. Дружинина, И.А. Волкова, О.П. Котова, Л.В. Себова

Компьютерная вёрстка: *А.В. Чудновский*

Художники: *В.Ю. Арзамасов, В.И. Голубев, А.О. Кулагина*

Издательская группа:

Э.Г. Кюн, Е.А. Аникиушкина, Г.М. Голубева, И.Н. Самохвалова

Подписано в печать 20.10.2009. Формат 60 × 84¹/₁₆.

Бумага офсетная. Печать офсетная.

Усл. печ. л. 11,4. Уч.-изд. л. 11,1. Тираж 100 экз. Заказ № 37

ГОУ ВПО «Московский физико-технический институт
(государственный университет)»