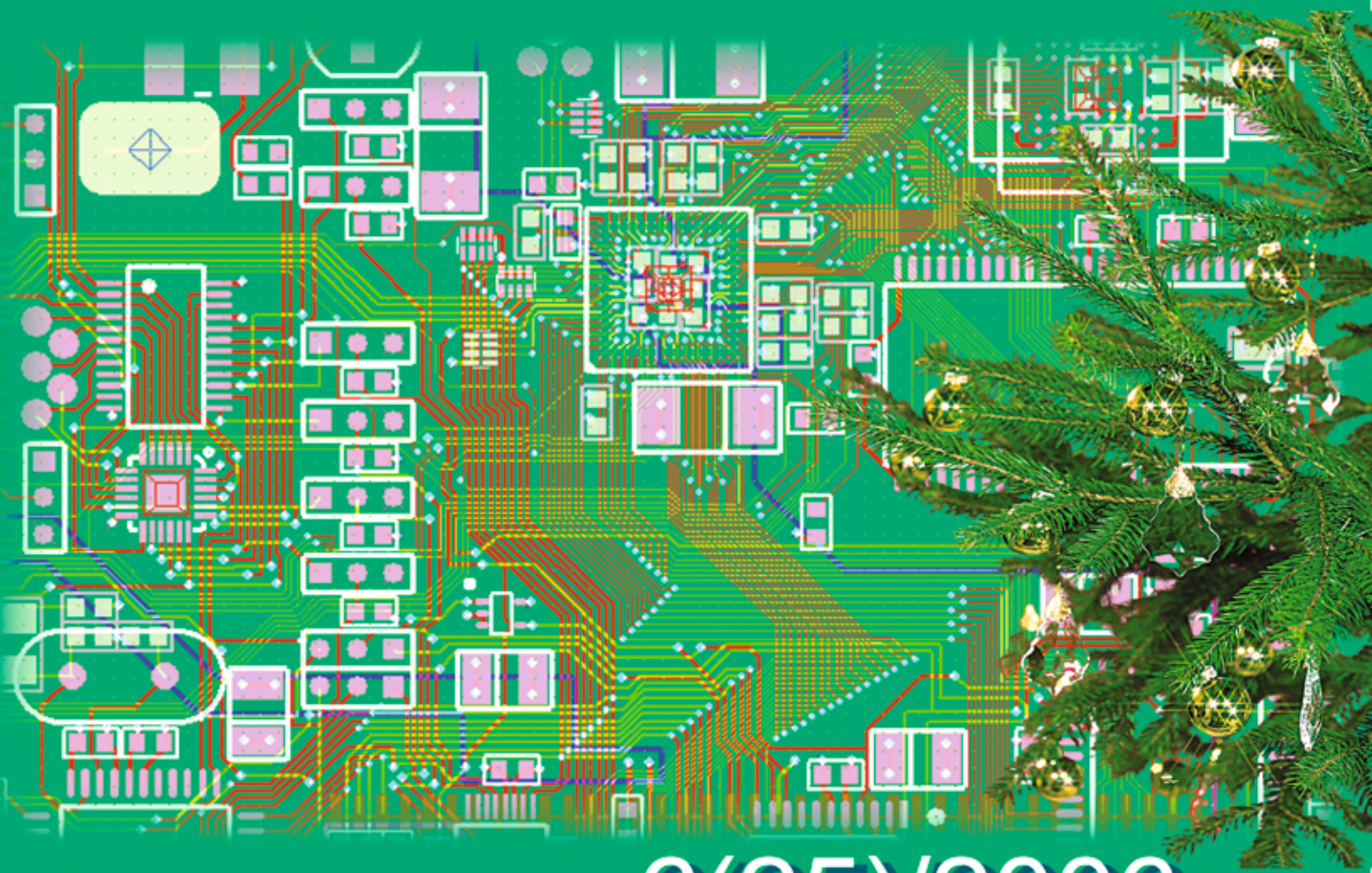


# ИНФОРМАЦИОННО- УПРАВЛЯЮЩИЕ СИСТЕМЫ

НАУЧНО-ПРАКТИЧЕСКИЙ ЖУРНАЛ



6(25)/2006

6(25)/2006

# ИНФОРМАЦИОННО-УПРАВЛЯЮЩИЕ СИСТЕМЫ

РЕЦЕНЗИРУЕМОЕ ИЗДАНИЕ

**Главный редактор**

М. Б. Сергеев,  
доктор технических наук, профессор

**Зам. главного редактора**

Г. Ф. Мощенко

**Редакционный совет:**

**Председатель** А. А. Оводенко,  
доктор технических наук, профессор  
В. Н. Васильев,  
доктор технических наук, профессор  
В. Н. Козлов,  
доктор технических наук, профессор  
Ю. Ф. Подоплекин,  
доктор технических наук, профессор  
Д. В. Пузанков,  
доктор технических наук, профессор  
В. В. Симаков,  
доктор технических наук, профессор  
А. Л. Фрадков,  
доктор технических наук, профессор  
Л. И. Чубраева,  
доктор технических наук, профессор, чл.-корр. РАН  
Р. М. Юсупов,  
доктор технических наук, профессор, чл.-корр. РАН

**Редакционная коллегия:**

В. Г. Анисимов,  
доктор технических наук, профессор  
Е. А. Крук,  
доктор технических наук, профессор  
В. Ф. Мелехин,  
доктор технических наук, профессор  
А. В. Смирнов,  
доктор технических наук, профессор  
В. И. Хименко,  
доктор технических наук, профессор  
А. А. Шалыто,  
доктор технических наук, профессор  
А. П. Шепета,  
доктор технических наук, профессор  
З. М. Юлдашев,  
доктор технических наук, профессор

**Редактор:** А. Г. Ларионова

**Корректор:** Т. В. Звертановская

**Дизайн:** М. Л. Черненко

**Компьютерная верстка:** Т. М. Каргапольцева

**Ответственный секретарь:** О. В. Муравцова

**Адрес редакции:** 190000, Санкт-Петербург,

Б. Морская ул., д. 67

Тел.: (812) 494-70-36

Факс: (812) 494-70-18

E-mail: 80x@mail.ru; ius@aanet.ru

Сайт: www.i-us.ru

Журнал зарегистрирован

в Министерстве РФ по делам печати,

телерадиовещания и средств массовых коммуникаций.

Свидетельство о регистрации ПИ № 77-12412 от 19 апреля 2002 г.

Журнал распространяется по подписке.

Подписку можно оформить через редакцию, а также

в любом отделении связи по каталогам:

«Пресса России» – № 42476;

«Роспечать» («Газеты и журналы») – № 15385

**ОБРАБОТКА ИНФОРМАЦИИ И УПРАВЛЕНИЕ**

**Беззатеев С. В., Литвинов М. Ю., Трояновский Б. К., Филатов Г. П.** Выбор алгоритма преобразования, обеспечивающего изменение структуры изображения 2

**Шолохов А. В.** Метод оценки достоверности информации при периодической коррекции наземных навигационных систем 7

**МОДЕЛИРОВАНИЕ СИСТЕМ И ПРОЦЕССОВ**

**Нестеренко В. Д.** Концепция построения архитектуры моделей процессов управления инфокоммуникационными сетями (Часть 1) 15

**ПРОГРАММНЫЕ И АППАРАТНЫЕ СРЕДСТВА**

**Ронжин А. Л., Карпов А. А., Лобанов Б. М., Цирульник Л. И., Йокиш О.** Фонетико-морфологическая разметка речевых корпусов для распознавания и синтеза русской речи 24

**Канжелев С. Ю., Шалыто А. А.** Автоматическая генерация автоматного кода 35

**ИНФОРМАЦИОННЫЕ КАНАЛЫ И СРЕДЫ**

**Рыжиков Ю. И.** Средние времена ожидания и пребывания в многоканальных приоритетных системах 43

**КРАТКИЕ СООБЩЕНИЯ**

**Голландцев Ю. А.** Сравнение механических характеристик асинхронных и вентильных индукторно-реактивных двигателей 50

**ХРОНИКА И ИНФОРМАЦИЯ**

Юсупов Рафаэль Мидхатович. Творческая биография 54

**СВЕДЕНИЯ ОБ АВТОРАХ**

57

**АННОТАЦИИ**

59

**СОДЕРЖАНИЕ ЖУРНАЛА «ИНФОРМАЦИОННО-УПРАВЛЯЮЩИЕ СИСТЕМЫ» за 2006 г. [№ 1–6]** 62

ЛР № 010292 от 18.08.98.

Сдано в набор 10.10.2006. Подписано в печать 15.12.2006. Формат 60×90/8.

Бумага офсетная. Гарнитура SchoolBookС. Печать офсетная.

Усл. печ. л. 8,0. Уч.-изд. л. 9,0. Тираж 1000 экз. Заказ 658.

Оригинал-макет изготовлен в редакционно-издательском центре ГУАП. 190000, Санкт-Петербург, Б. Морская ул., 67.

Отпечатано с готовых диапозитивов в редакционно-издательском центре ГУАП. 190000, Санкт-Петербург, Б. Морская ул., 67.



УДК 681.3

## ВЫБОР АЛГОРИТМА ПРЕОБРАЗОВАНИЯ, ОБЕСПЕЧИВАЮЩЕГО ИЗМЕНЕНИЕ СТРУКТУРЫ ИЗОБРАЖЕНИЯ

**С. В. Беззатеев,**

канд. техн. наук, доцент

**М. Ю. Литвинов,**

соискатель

**Б. К. Трояновский,**

доцент

**Г. П. Филатов,**

соискатель

Санкт-Петербургский государственный университет аэрокосмического приборостроения

*Рассматривается проблема выбора эффективного алгоритма преобразования видеoinформации в видеосистемах встраиваемого класса, обеспечивающего ее конфиденциальность при передаче и хранении. Предлагается модификация алгоритма VEA, позволяющая повысить защищенность информации в условиях атак по перехваченной преобразованной информации и выбранному исходному изображению и обеспечивающая эффективное уничтожение структуры передаваемой видеoinформации.*

*The authors study the choice of an effective algorithm for the transformation of videoinformation in embedded video systems that ensures its confidentiality on keeping and transmission. A modification of the VEA algorithm is proposed. This modification increases the protection level of the information against the interception attacks and ensures an efficient destruction of the transmitted information.*

### Введение

Для решения задачи обеспечения конфиденциальности передаваемой и хранимой информации традиционно используется специальное преобразование, устойчивое к различного типа атакам. В контексте данной задачи наиболее эффективными и опасными являются атаки по парам исходное — преобразованное изображение и по выбранному исходному изображению. Кроме того, следует отметить необходимость предотвращения атак, связанных со спецификой исходной информации, — изображения, которое изначально содержит характерные фрагменты (контуры) и тем самым уже по умолчанию предполагает возможность атаки по парам исходное — преобразованное изображение. Задача выбора алгоритма преобразования осложняется спецификой видеосистем встраиваемого класса — существенными ограничениями на свободные вычислительный ресурс и объем оперативной памяти.

### Потоковые и блочные преобразования

Все преобразования условно разделяют на два типа — потоковые и блочные. К потоковым преобразованиям относятся системы, использующие

ключевые последовательности, генерируемые регистрами сдвига, и системы хаотичного преобразования (CVEC). Такие системы обеспечивают высокую скорость обработки информации, однако они неустойчивы к атакам с использованием перехваченных образцов обработанного и исходного изображения [14]. Системы блочного преобразования, в свою очередь, можно разделить на системы полного и частичного преобразования. Системы частичной обработки видеoinформации, использующие так называемые методы селективного преобразования, делятся на системы, учитывающие структуру видеоформата и ориентированные на обработку форматов сжатого изображения [1–6, 8–12], и системы, предназначенные для обработки несжатой информации [13]. Рассмотрим особенности этих подходов на примере нескольких наиболее известных алгоритмов.

1. Video Encryption Algorithm by Okao and Nahrstedt [8]. Основной идеей алгоритма является изменение статистических свойств видеоформата MPEG. Преобразование представляет собой аналог одного раунда сети Фейстейла. Первоначально весь информационный поток разбивается на блоки одинаковой длины (например, на байты), таким



■ *Рис. 1. Пример использования алгоритма RVEA (а), алгоритма Video Encryption Algorithm (б) к исходному изображению (в)*

образом получается два списка — четные и нечетные байты. Результатом преобразования являются байты, полученные как побайтная сумма по модулю два (исключающее ИЛИ) четных и нечетных байт, и преобразованные с использованием симметричного алгоритма (AES, DES, ГОСТ 28147–89 и т. п.) нечетные или четные байты. Следует отметить недостаточную защищенность данного алгоритма при его использовании для обработки последовательности кадров и возможность применения атаки по выбранному исходному видеофрагменту. Очевидно, что данный недостаток может быть легко преодолен, если в алгоритме обработки использовать несколько раундов сети Фейстейла.

2. Одним из характерных примеров систем частичного преобразования, использующих структуру видеоформата, является семейство алгоритмов, ориентированных на стандарт MPEG [3]. Наиболее защищенным из этого семейства алгоритмов считается алгоритм RVEA, который обеспечивает защиту от атак как по перехваченному преобразованному изображению, так и по выбранному исходному изображению. В отличие от остальных алгоритмов этого семейства RVEA преобразует не только знаковые биты коэффициентов DCT, но и знаковые биты векторов движения формата MPEG. В общей сложности RVEA подвергает обработке около 10% всего объема MPEG-файла. Однако сравнение результатов обработки реального изображения алгоритмами, приведенными в работах [3, 8], наглядно демонстрирует преимущества первого алгоритма в случае, если стоит задача разрушить структуру передаваемого изображения (рис. 1).

3. Использование подхода частичного преобразования несжатого видеоизображения рассмотрено в работе [13]. Основной идеей является обработка не всего информационного блока для отдельной точки изображения, а только старших (одного или двух) бит для каждого блока. Для преобразования старших бит используются стандартные симметричные алгоритмы типа AES, DES, ГОСТ 28147–89 и т. п. Очевидно, что такой подход дает существенный выигрыш по затратам на обработку изображения. К сожалению, сами авторы отмечают дос-

точную устойчивость этого алгоритма к атакам по перехваченному изображению, подвергнутому преобразованию, лишь в случае, когда обрабатывалось не менее 50 % от объема каждого информационного блока.

Основываясь на проведенном анализе свойств существующих систем и учитывая требования, предъявляемые к системе преобразования видеопотока:

- защищенность к атаке по известному открытому изображению;
- полное разрушение структуры видеoinформации;
- защищенность от атаки по парам исходная — преобразованная видеoinформация;
- независимость от используемого видеоформата —

следует выбрать систему непрерывного, полного преобразования всего объема видеoinформации, используя потоковый или блочный алгоритм. Кроме того, учитывая условие старения информации (ее актуальности), определяемое временным периодом до нескольких дней, можно использовать упрощенный вариант известных алгоритмов (таких как AES, DES, ГОСТ 28147–89).

### Блочное преобразование и его модификация

Для того чтобы блочный алгоритм преобразования обладал свойством разрушения структуры видеoinформации, должно выполняться, по крайней мере, следующее свойство: одинаковые блоки исходного изображения должны преобразовываться в существенно отличающиеся друг от друга блоки обработанной информации, т. е. для любых  $I_j = I_k$   $E(I_j) \neq E(I_k)$ . Такое свойство для преобразования  $E(*)$  может быть достигнуто двумя способами: постоянно меняющейся функцией  $E(*)$  или использованием для преобразований функции от двух переменных —  $E(*, K)$ , где  $K$  — некоторый параметр, меняющийся для каждого нового блока исходного изображения. Рассмотрим возможные методы реализации каждого из перечисленных способов.

1. Постоянное изменение функции преобразования  $E(*)$  может быть достигнуто либо с исполь-

зованием детерминированного алгоритма изменения функции  $E(*)$ , который должен сохраняться в тайне, либо с использованием некоторого случайного преобразования. Первый вариант, очевидно, сводится к введению дополнительного секретного параметра, позволяющего менять вид функции. Во втором варианте необходимо найти такое случайное преобразование, результат применения которого можно было бы легко восстановить, не зная самого преобразования. Одним из методов такого преобразования является использование помехоустойчивого кода в качестве детерминированной части преобразования  $E(*)$ , при этом секретом будет эффективный алгоритм декодирования (исправления ошибок) для выбранного кода. Случайной частью преобразования  $E(*)$  будет генерация случайного вектора ошибки, вес Хэмминга которого не должен превышать числа ошибок, исправляемых кодом. В общем виде такое преобразование можно записать следующим образом:

$$pAGP \oplus e = c,$$

где  $p$  — информационный блок изображения длины  $k$ ;  $G$  — порождающая матрица  $(n, k, d)$ -помехоустойчивого кода (здесь  $k$  — длина информационного блока,  $n$  — длина кодового слова,  $d$  — минимальное расстояние кода);  $A$  — матрица  $(k \times k)$ , имеющая обратную;  $P$  — матрица перестановок  $(n \times n)$ ;  $e$  — случайный вектор ошибки длины  $n$ ,  $wt(e) \leq (d - 1)/2$ , являющийся в данном случае параметром  $K$  для выполняемого преобразования.

Для восстановления исходного блока изображения  $p$  первоначально используется алгоритм декодирования с исправлением случайного вектора ошибки  $e$ , а затем из полученного кодового слова восстанавливается значение информационного блока. Устойчивость такого метода к различным атакам подтверждается безопасностью алгоритма Мак Эллиса, использующего такой же подход в несимметричных системах. Этот метод обработки видеoinформации может быть модифицирован, если использовать идею неравнозначных бит в информационном блоке [13] и, соответственно, помехоустойчивые коды с неравной защитой позиций в кодовом слове [15].

2. В данной статье предлагается использовать функцию от двух переменных, в качестве которой может выступать любой симметричный алгоритм блочного преобразования типа ГОСТ. Для обеспечения быстрой синхронизации при потере пакетов в распределенных IP-системах начальное значение параметра  $K_{i0}$  для пакета с номером  $i$  может быть установлено, например, следующим образом:

$$K_{i0} = F(K_0, i),$$

где  $K_0$  — начальное значение параметра.

В качестве функции  $F(*, *)$  может быть выбрано простое арифметическое сложение 256 битных чисел с игнорированием разряда переполнения.

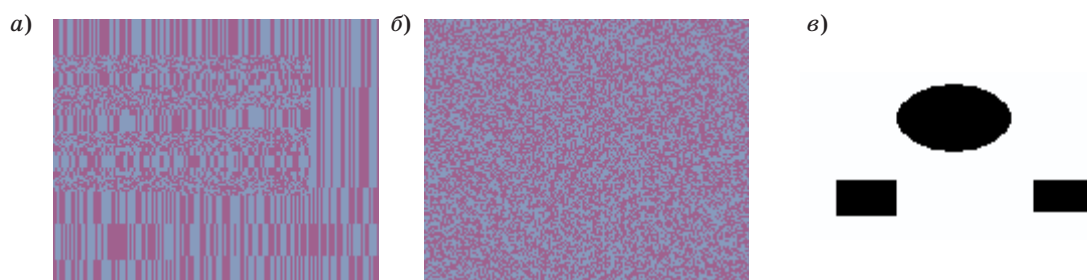
### Пример использования модифицированного блочного преобразования

Рассмотрим более подробно предлагаемый в данной работе блочный алгоритм преобразования, адаптируемый по параметру защищенность/сложность реализации. За его основу взята модификация алгоритма VEA [8] и сеть Фейстейла со стандартным размером входного информационного блока в 64 бита и длиной параметра  $K$  256 бит. На каждом раунде выполняются простейшие (аналогичные ГОСТ 28147–89) операции замены и перестановки. Число раундов можно изменять от 8 до 16 в зависимости от требуемого соотношения защищенность/сложность реализации. Для согласования алгоритма с требованиями использующего его приложения преобразованию подвергаются пакеты данных, длина которых может лежать в некотором ограниченном диапазоне, но последовательные пакеты могут иметь разные длины и не обязательно кратные длине информационного блока в 64 бита. Функция  $K_{i0} = F(K_0, i)$  быстрой синхронизации реализована в виде, не зависящем от длины пакета, и использует некоторое, заранее фиксированное, количество операций арифметического сложения с игнорированием переполнения. Алгоритм был реализован в виде приложения на языке С и тщательно тестировался на различных изображениях. Проверялась эффективность алгоритма по изменению структуры аудиоданных.



■ Рис. 2. Пример использования предложенного алгоритма при одной модификации ключа на пакет длиной 1200 байт (а), на блок длиной 64 бита (8 байт) (б) для исходного изображения (в)





■ **Рис. 3.** Пример использования предложенного алгоритма при одной модификации ключа на пакет длиной 1200 байт (а), на блок длиной 64 бита (8 байт) (б) к исходному монохромному изображению (в)

Примеры работы предложенного алгоритма по изменению структуры изображения представлены на рис. 2.

Более сложный случай — монохромные, высококонтрастные изображения (рис. 3).

Для получения достоверных оценок сложности программной реализации предлагаемого алгоритма программа была протестирована в инструментальной среде разработки Code Composer Studio ver. 3.1 фирмы Texas Instruments для 32-битного сигнального процессора семейства TMS64XX. Использовались только средства оптимизации компилятора языка С.

В качестве оценки сложности алгоритма выступает количество тактов на байт преобразуемой информации.

#### Характеристики вычислительной сложности

Количество раундов в алгоритме ГОСТ	8	16	32
Сложность (такты/байт)	25	47	91

Так, при 8 раундах преобразования процессор с тактовой частотой 600 МГц способен в реальном времени преобразовывать поток 24 Мбайт/с.

#### Заключение

Предложенная в работе модификация блочного преобразования с изменяющимся параметром позволяет обеспечить эффективное искажение структуры изображения и тем самым предотвратить наиболее распространенные методы атак. Само преобразование выполняется с использованием хорошо известной схемы сетей Фейстейла с не менее чем восьмью раундами, устойчивой ко всем известным в настоящее время атакам. Очевидным направлением повышения качества предложенного здесь метода является нахождение алгоритма случайной генерации параметра  $K$  блочного преобразования при обработке изображения, позволяющего выполнить обратное преобразование без знания значения  $K$ . Таким образом, представляется эффективным решение, которое позволило бы объединить преимущество схемы Мак Эллиса, ис-

пользующей случайный параметр — вектор ошибки  $e$ , с простотой описанного в данной работе модифицированного блочного преобразования.

#### Литература

1. Agi, Gong L. An empirical study of secure MPEG video transmissions // ISOC Symposium on Network and Distributed Systems Security. San Diego, California. 1996. P. 137–144.
2. Alattar M., Al-Regib G. I., Al-Semari S. A. Improved selective encryption techniques for secure transmission of MPEG video bit-streams // International Conference on Image Processing (ICIP'99): Proc. of the 1999 IEEE. (IEEE Signal Processing Society. 1999).
3. Bhargava B., Shi C., Wang Y. MPEG Video Encryption Algorithms. <http://raidlab.cs.purdue.edu/papers/mm.ps>
4. Cheng H., Li X. On the application of image decomposition to image compression and encryption // Communications and Multimedia Security: Second Joint Working Conference on Communications and Multimedia Security, CMS'96. Chapman & Hall. Essen, Germany. Sept. 1996. P. 116–127.
5. Cheng H., Li X. Partial encryption of compressed images and videos // IEEE Transactions on Signal Processing. 2000. 48(8):2439–2451.
6. Pommer, Uhl A. Selective encryption of wavelet packet subband structures for obscured transmission of visual data // 3<sup>rd</sup> IEEE Benelux Signal Processing Symposium (SPS 2002): Proc (IEEE Benelux Signal Processing Chapter). Leuven, Belgium. Mar. 2002. P. 25–28.
7. Qiao L., Nahrstedt K. Comparison of MPEG encryption algorithms // International Journal on Computers and Graphics (Special Issue on Data Security in Image Communication and Networks). 1998. 22(3):437–444.
8. Qiao L., Nahrstedt K. A New Algorithm for MPEG Video Encryption. Proc. of the 1<sup>st</sup> International Conference on Imaging Science (Systems and Technology (CISST '97). Las Vegas, NV. July 1997. P. 21–29.
9. Schneck P. A., Schwan K. Authenticast: An adaptive protocol for high-performance, secure network applications: Technical report / Georgia Institute of Technology. Atlanta, GA, USA. 1997.

10. Shi C., Bhargava B. A fast MPEG video encryption algorithm: Proc. of the ACM Multimedia 1998. Boston, USA. 1998. P. 81–88.
11. Skrepth C. J., Uhl A. Selective encryption of visual data: Classification of application scenarios and comparison of techniques for lossless environments // Advanced Communications and Multimedia Security: Sixth Joint Working Conference on Communications and Multimedia Security CMS'02. Kluwer Academic Publishing. Portoroz, Slovenia. Sept. 2002.
12. Tang L. Methods for encrypting and decrypting MPEG video data efficiently: Proc. of the ACM Multimedia 1996. Boston, USA. Nov. 1996. P. 219–229.
13. Podesser M., Schmidt H.-P., Uhl A. Selective bitplane encryption for secure transmission of image data in mobile environments. [http://www.cosy.sbg.ac.at/~uhl/norsig\\_slides.pdf](http://www.cosy.sbg.ac.at/~uhl/norsig_slides.pdf)
14. Li S., Li C., Chen G. Cryptanalysis of the RCES/RSES Image Encryption Scheme: Electronic preprint. IACR's Cryptology ePrint Archive: Report 2004/376. 2004.
15. Bezzateev S., Shekhunova N. Generalized Goppa codes for correcting localized errors: Proc. of ISIT-98. Boston, USA. 1998. P. 377.

**СВЯЗЬ. АВТОМАТИЗАЦИЯ. ЭЛЕКТРОНИКА.  
ИНДУСТРИЯ БЕЗОПАСНОСТИ-2007  
СПЕЦИАЛИЗИРОВАННАЯ ВЫСТАВКА  
20–23 марта 2007 г.**

Место проведения: 644033, Россия, г. Омск, ул. Красный Путь, 155, корп.1

**Организаторы**

Международный выставочный центр «ИНТЕРСИБ»  
Правительство Омской области  
Администрация города Омска

**Направления работы выставки**

*Связьинфо. Интернет-техком*

Специализированный салон системных решений информационных технологий, сети, сетевые телекоммуникационные и Интернет/интранет-технологии  
Автоматизированные системы связи и системы управления связью  
Системы и аппаратура радиосвязи, спутниковой и космической связи  
Средства телевидения и радиовещания  
Системы и аппаратура передачи данных, коммутационное оборудование  
Системы и оборудование для обеспечения контроля и безопасности систем и сетей связи и т. д.  
Радиоизмерительная техника  
Периферийное оборудование для обработки данных  
Рабочие и прикладные программы любого назначения  
Сети  
Сетевое оборудование и комплексные системы обеспечения телекоммуникационных систем связи  
Оборудование и комплексные решения для внешних и локальных сетей связи  
Комплексные решения по коммуникационной поддержке деятельности предприятий малого и среднего бизнеса

*Автоматизация*

Системы и технологии автоматизации производства, проектирования и управления  
Технические средства, системы и компоненты автоматического контроля и управления  
Периферийное оборудование  
Лазерная техника

Обработка изображений в промышленном процессе  
Обеспечение и контроль качества  
Информационные технологии и программное обеспечение

*Электроника*

Электронные компоненты  
Полупроводники  
Встроенные системы  
Микропроцессоры  
Сенсоры и микросистемы  
Печатные платы  
Компоненты и subsystemы  
Технологии, оборудование и материалы для электронных производств  
Оптическое, лазерное и радиоэлектронное оборудование  
Электромеханические компоненты и соединительные технологии  
Электронная техника

*Технологии безопасности. Средства МЧС*

Системы контроля и управления доступом; считыватели и идентификаторы; домофоны; устройства для персонализации и защиты карт СКУД; антикражные системы  
Системы охранной сигнализации; периметральные средства обнаружения; приемно-контрольная аппаратура; средства оповещения; вспомогательное оборудование  
Системы телевизионного наблюдения; устройства усилительные и аккумуляторные; аппаратура приема, регистрации, обработки и передачи изображений; вспомогательные аксессуары и др.

**Дополнительная информация**

<http://www.exponet.ru/exhibitions/by-id/industrysecurityom/industrysecurityom2007/index.ru.html>

УДК 621.391

# МЕТОД ОЦЕНКИ ДОСТОВЕРНОСТИ ИНФОРМАЦИИ ПРИ ПЕРИОДИЧЕСКОЙ КОРРЕКЦИИ НАЗЕМНЫХ НАВИГАЦИОННЫХ СИСТЕМ

**А. В. Шолохов**

канд. техн. наук, доцент

Серпуховский военный институт ракетных войск

*Разработан способ определения достоверности информации навигационных систем, корректируемых эпизодически с привлечением различных внешних источников, на основе методов теории марковских случайных процессов. Рассмотрены отличительные особенности задачи, приводящие к значительным погрешностям при оценке достоверности известными методами, показаны пути учета рекуррентного характера событий и непостоянства интенсивностей их потоков.*

*A method to determine the information reliability of navigating systems with periodical correction is developed. The study is based on the theory of Markov stochastic processes. We emphasize some special features of this problem which result in significant errors in the estimation of reliability by known methods, and show some ways of taking into account the recurrent character of events and the inconstancy of intensity of their streams.*

## Введение

Анализ качества алгоритмов интегрированных навигационных систем традиционно предполагает нахождение потенциальной точности оцениваемых параметров [1, 2]. Зная ее, можно определять и характеристики достоверности [3]. Однако при нахождении апостериорной плотности распределения навигационных параметров известными методами не учитываются вероятности «аномальных» событий, которые возникают в связи с привлечением внешней информации и, как правило, снижают достоверность.

Примерами таких событий служат различные типы информационных нарушений: пропадание сигнала спутниковой навигационной системы, непрогнозируемое изменение псевдодальности одного из спутников в созвездии и пр. В настоящее время известны различные методы их идентификации [4, 5], но вероятностные и временные характеристики процедур на их основе пока не учитываются в оценках достоверности выходных параметров. Перечень «аномальных» событий не ограничивается информационными нарушениями. К ним можно отнести события, связанные с работой обслуживающего персонала. Для навигационных систем военного назначения характерны события, обусловленные информационным противоборством сторон.

В зависимости от конкретных условий вероятностями указанных событий можно пренебречь без

значимых потерь в оценке качества навигационной информации. При оценке достоверности информации автономных наземных навигационных систем (ННС) необходим учет распределения расстояний между контрольными пунктами (КП), расположенными на маршрутах движения и используемыми для обнуления возрастающих во времени погрешностей [6]. Кроме этого, привлечение информации цифровых карт дорог (ЦКД) для уточнения выработанного ННС местоположения сопряжено с ошибками распознавания КП [7, 8] или выбора дороги, по которой фактически движется объект, например на перекрестках [9–11]. Получение точных оценок достоверности усложняется вследствие рекуррентного характера событий коррекции, обусловленного необходимостью принудительной коррекции ННС на КП. В связи с этим возникает необходимость разработки нового подхода к оценке достоверности информации, позволяющего учитывать различные события в интегрированных системах навигации, в первую очередь, ошибки, связанные с привлечением внешней информации.

## Постановка задачи

Процессы изменения во времени погрешностей ННС с достаточной для практических приложений точностью описываются случайными марковскими моделями. Поэтому используем теорию марковских случайных процессов [3] и при решении за-



дачи оценки достоверности. Рассмотрим состояния процесса функционирования интегрированной системы, ее граф состояний, начальные и переходные вероятности состояний.

При формировании состояний процесса функционирования интегрированной системы (далее — состояния системы) учтем, что достоверность информации зависит не только от погрешностей, накопленных ННС к заданному моменту времени, но и от периодичности ее коррекции с привлечением внешних источников [6], а также вероятностей ошибок при их использовании. Исходя из этого введем два альтернативных признака функционирования системы: режим работы (автономный или с привлечением внешней информации) и состояние погрешностей (в пределах установленного допуска или за пределами). Сочетания возможных реализаций этих признаков позволяют сформировать состояния  $S_0, S_1, S_2$  системы (табл. 1). Можно различать состояния при выходе погрешностей за допуски, но в решаемой задаче необходимости в этом нет.

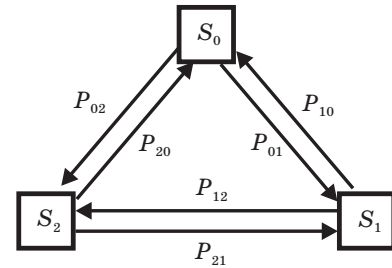
Сформированные состояния  $S_X, X = 0, 1, 2$ , позволяют определить достоверность информации интегрированной системы как вероятность  $P$  ее нахождения во множестве состояний, характеризующихся набором признаков удовлетворительного качества работы. Таковыми здесь являются состояния  $S_0, S_1$ , поэтому  $P = P_0 + P_1$ .

В общем случае вероятности  $P_X(L)$  нахождения системы в состояниях (далее — вероятности состояний) зависят от пройденного пути  $L$  (или времени), своих начальных значений  $P_X(0)$ , переходных вероятностей  $P_{XX}$  или интенсивностей. Сумма вероятностей  $P_X(L)$  равна единице, поскольку система может находиться только в одном из состояний  $S_X$ . Начальные значения  $P_X(0)$  вероятностей предположим известными. Как правило, автономные ННС требуют привлечения внешней информации при начальной настройке [1, 6], поэтому начальная вероятность состояния  $S_0$  равна единице, остальные — нулю.

Все возможные переходы системы из одного состояния  $S_0...S_2$  в другое показаны на рис. 1 стрелками. Они характеризуются переходными вероятностями  $P_{XX}$ , которые зависят от характеристик ННС и внешних источников информации. Обоснование формул нахождения  $P_{XX}$  довольно многогранно, поэтому оно вынесено за рамки постановки задачи и рассматривается ниже.

■ Таблица 1. Признаки работы и состояния интегрированной системы

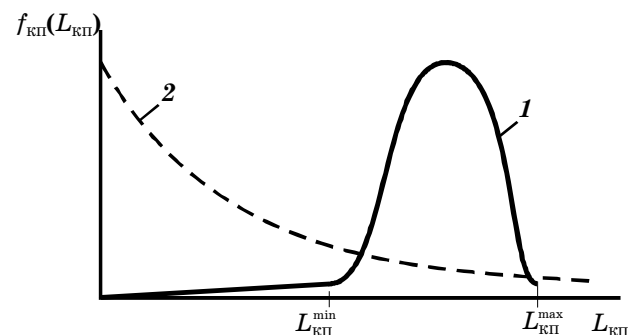
Режим работы навигационной системы	Состояние погрешностей	
	в допуске	за допуском
Работа с привлечением внешней информации	$S_0$	$S_2$
Автономная работа	$S_1$	$S_2$



■ Рис. 1. Размеченный граф состояний интегрированной системы

Исходные данные  $P_X(0), P_{XX}$  являются достаточными для нахождения зависимостей  $P_X(L)$  «классическими» способами [3], но при этом могут возникнуть значительные погрешности в искомой оценке достоверности  $P$ . Причина кроется в существующем подходе к обеспечению заданной точности и достоверности информации автономных навигационных систем, предусматривающем их периодическую коррекцию с привлечением внешних источников [6]. Рассмотрим это более подробно.

Одними из основных внешних источников в рассматриваемой задаче являются КП, где точно известны координаты и другие геодезические параметры. В них обнуляются погрешности ННС, т. е. реализуются переходы системы в состояние  $S_0$ . Поток событий, вызывающих данные переходы, характеризуется непостоянной интенсивностью, зависящей от величины пройденного пути  $L$ , что значительно усложняет нахождение вероятностей состояний  $P_X(L)$ . Физически это объясняется тем, что расстояние  $L_{КП}$  между КП является случайной величиной, фактический закон распределения которой существенно отличается от показательного. Контрольные пункты располагаются через заданные промежутки пройденного пути, поэтому функция плотности распределения вероятностей (далее — плотность)  $f_{КП}(L_{КП})$  расстояния между КП имеет вид, показанный схематично на рис. 2 (линия 1). Аналитически она может быть задана, например, посредством усеченного гауссова распределения



■ Рис. 2. Плотность распределения вероятностей расстояния между КП

$$\left. \begin{aligned}
 f_{\text{КП}}(L_{\text{КП}}) &= \begin{cases} f_N(L_{\text{КП}})G & \text{при } 0 \leq L_{\text{КП}} \leq L_{\text{КП}}^{\text{max}} \\ 0 & \text{иначе} \end{cases} \\
 f_N(L) &= \exp\left(-\frac{1}{2D_{\text{КП}}}\left[L - \frac{1}{2}(L_{\text{КП}}^{\text{min}} + L_{\text{КП}}^{\text{max}})\right]^2\right) \\
 \frac{1}{G} &= \int_{-\infty}^0 f_N(L)dL + \int_{L_{\text{КП}}^{\text{min}}}^{\infty} f_N(L)dL, \quad D_{\text{КП}} \approx \sqrt{\frac{1}{6}(L_{\text{КП}}^{\text{max}} - L_{\text{КП}}^{\text{min}})}.
 \end{aligned} \right\} (1)$$

Для сравнения приведена плотность показательного закона распределения (линия 2), которая, как видно, не является хорошей аппроксимацией распределения вида (1). Из рисунка ясно, что коррекция ННС на КП наиболее вероятна, когда величина пути  $L_{\text{КП}}$ , пройденного объектом от предыдущего КП, больше минимального рекомендуемого расстояния  $L_{\text{КП}}^{\text{min}}$  между КП, но не превышает максимально допустимого  $L_{\text{КП}}^{\text{max}}$ . Как следствие, момент «встречи» КП в будущем в значительной степени зависит от пути, пройденного с момента нахождения объекта на КП в прошлом. Следовательно, в рассматриваемой задаче переходные вероятности  $P_{XX}(L)$  являются функциями пройденного пути, которые находятся в соответствии с заданной плотностью  $f_{\text{КП}}(L_{\text{КП}})$  и другими характеристиками конкретной ННС. Пренебрежение указанным последствием (т. е. принятие допущения  $P_{XX}(L) = \text{const}$ ) приводит к значительным погрешностям в оценке достоверности.

Кроме последствия в решаемой задаче необходимо учитывать также так называемые рекуррентные события — моменты прохождения объектом КП. Их рекуррентный характер состоит в том, что вероятность перехода из любого состояния в  $S_0$  стремится к единице, если величина пути, отсчитываемого от последнего пройденного КП, стремится к  $L_{\text{КП}}^{\text{max}}$ :

$$P_{X0}(L_{\text{КП}}^{\text{max}}) = 1. \quad (2)$$

Особенность решаемой задачи, по сравнению с известными задачами, в которых рассматриваются рекуррентные события «омоложения» [3, с. 396], определяется наличием не одного состояния, интерпретируемого обычно как «возраст» системы, а целого их набора ( $S_0, S_1, S_2$ ) в каждый момент работы. Причем в интегрированной системе в произвольные моменты необходимо рассматривать не

только процессы «омоложения», определяемые  $P_{X0}$ , но и прочие переходы между различными состояниями, вероятности которых связаны с  $P_{X0}$ :

$$P_{X0}(L) + \sum P_{X0}(L) = 1, \quad (3)$$

где суммирование проводится по всем индексам, содержащим ноль, кроме 0.

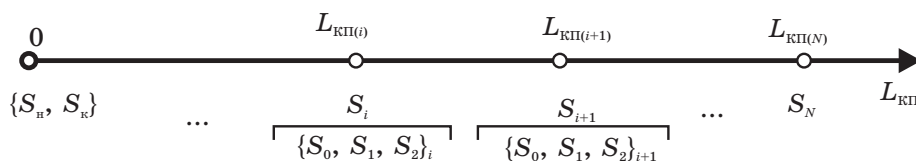
Таким образом, наличие последствия и рекуррентных событий определяет необходимость разработки нового подхода к решению задачи оценки достоверности эпизодически корректируемой интегрированной системы. Задача сводится к нахождению достоверности  $P$  на основе вероятностей  $P_X(L)$  соответствующих состояний  $S_X$  интегрированной системы. Их начальные значения  $P_X(0)$  известны, переходные вероятности  $P_{XX}(L)$  находятся в соответствии с известной плотностью  $f_{\text{КП}}(L_{\text{КП}})$  (1) и с учетом ограничений (2) и (3), характеризующих рекуррентные события прохождения объектом КП.

### Общее решение задачи оценки достоверности

Решение поставленной задачи целесообразно искать, основываясь на рассмотрении работы системы на отрезке  $[0; L_{\text{КП}}^{\text{max}}]$  между двумя последовательно расположенными КП. Это позволяет, зная  $f_{\text{КП}}(L_{\text{КП}})$ , сравнительно просто определять переходные вероятности  $P_{XX}(L_{\text{КП}})$  как функции расстояния  $L_{\text{КП}}$  от предыдущего КП. С другой стороны, такой подход не накладывает ограничений на длину  $L$  пройденного пути, которая совпадает с  $L_{\text{КП}}$  только до первой коррекции навигационной системы.

Поскольку переходные вероятности зависят от положения объекта на отрезке  $[0; L_{\text{КП}}^{\text{max}}]$ , будем рассматривать тройку состояний  $S_i = \{S_0, S_1, S_2\}_i$  в каждой  $i$ -й точке на указанном отрезке (рис. 3). Все эти  $N$  точек предположим для простоты узлами регулярной сетки с известным шагом по  $L_{\text{КП}}$ . Значения  $L, L_{\text{КП}}$  и  $N$  могут быть определены по известной скорости  $V$  объекта, периоду дискретности  $\Delta t$  и такту работы  $j$  системы навигации.

Началу отсчета на оси  $L_{\text{КП}}$  соответствует нахождение системы на КП. Однако, учитывая затраты времени на проведение технологических операций коррекции навигационной системы, вместо состояния  $S_0$  на КП введем два состояния:  $S_{\text{н}}$  — начала и  $S_{\text{к}}$  — окончания процедуры коррекции. При этом необходимость рассмотрения состояний  $S_1, S_2$  в КП отпадает. В произвольных точках на



■ Рис. 3. Состояния интегрированной системы

оси  $L_{КП}$  состояния  $S_0$  соответствуют привлечению внешних источников информации, кроме КП (ЦКД [6–11], спутниковых навигационных систем [1, 2], рельефа [12] и пр.). Таким образом, в произвольный момент функционирования навигационной системы начало работы на отрезке  $[0; L_{КП}^{max}]$  характеризуется двумя состояниями  $S_H, S_K$ , а каждая  $i$ -я точка – тройкой состояний  $S_0, S_1, S_2$ . Объединим все указанные состояния в векторе

$$S = [S_H \ S_K \ \underbrace{S_0 \ S_1 \ S_2}_{i=1} \ \underbrace{S_0 \ S_1 \ S_2}_{i=2} \ \dots \ \underbrace{S_0 \ S_1 \ S_2}_{i=N}]^T, \quad (4)$$

для которого сформируем вектор  $P_S(j)$  вероятностей в  $j$ -й момент функционирования интегрированной системы. В начальный момент ( $L = 0, j = 0$ ) целесообразно предположить, что система находится в готовности на КП, т. е. вероятность состояния  $S_K$  равна единице, а остальных состояний — нулю:

$$P_S(0) = [0 \ 1 \ \underbrace{0 \ 0 \ 0}_{i=1} \ \underbrace{0 \ 0 \ 0}_{i=2} \ \dots \ \underbrace{0 \ 0 \ 0}_{i=N}]^T. \quad (5)$$

Располагая начальными вероятностями  $P_S(0)$  (5) состояний и соответствующей матрицей  $P_{XП}$  переходных вероятностей (она будет рассмотрена ниже), можно рассчитать вероятности состояний  $S$  (4) в  $j$ -е моменты процесса функционирования интегрированной системы по известной рекуррентной формуле

$$P_S(j+1) = (P_{XП})^T P_S(j), \quad j = 0, \dots, N_j, \quad (6)$$

где  $P_S(j) = [P_H(j) \ P_K(j) \ \underbrace{P_0(j) \ P_1(j) \ P_2(j)}_{i=1} \ \dots \ \underbrace{P_0(j) \ P_1(j) \ P_2(j)}_{i=N}]$  — вектор вероятностей состояний  $S$  (4);  $N_j$  — общее число моментов процесса функционирования системы, определяемое интервалом времени или пройденного пути, на котором необходимо оценить достоверность информации.

Формула (6) позволяет найти вероятности состояний  $S_0, S_1, S_2$  в  $i$ -х точках на оси  $L_{КП}$  (см. рис. 3) в  $j$ -е моменты работы навигационной системы. Следовательно, для определения вероятностей указанных состояний в  $j$ -й момент независимо от расположения объекта между КП, т. е. независимо от  $i$ , необходимо в векторе  $P_S(j)$  выполнить суммирование по  $i$  соответствующих вероятностей. Для получения вероятности  $P_{SX}(j)$  пребывания процесса функционирования интегрированной системы в интересующем состоянии  $S_X$  в  $j$ -й момент необходимо в векторе  $P_S(j)$  выполнить суммирование по  $i$  вероятностей  $P_X(j)$  соответствующих состояний  $S_X$ :

$$P_{SX}(j) = \sum_{i=0}^N \underbrace{P_X(j)}_i. \quad (7)$$

Окончательно оценки достоверности  $P(j)$ , т. е. вероятности того, что погрешности не будут выходить за установленные допуски, находятся в  $j$ -й момент на основании вероятностей  $P_{SX}(j)$

$$P(j) = P_{S_K}(j) + P_{S_0}(j) + P_{S_1}(j) = 1 - P_{S_H}(j) - P_{S_2}(j) \quad (8)$$

для известных (связанных с  $j$ ) значений пройденного пути  $P(L) = P(jV\Delta t)$ .

### Определение переходных вероятностей

*Потоки событий в алгоритме интегрированной навигационной системы.* Общий подход к определению матрицы  $P_{XП}$  переходных вероятностей рассмотрим для двух последовательных моментов процесса функционирования интегрированной системы. В  $(i+1)$ -й момент используем обозначения  $S_{0*}, S_{1*}$  и  $S_{2*}$  (рис. 4) для состояний системы, аналогичных состояниям  $S_0, S_1$  и  $S_2$  в  $i$ -й момент работы соответственно. Состояния  $S_H$  и  $S_K$  присутствуют во всех моментах работы навигационной системы, поэтому для них новых обозначений не требуется.

Возможные переходы интегрированной системы в  $i$ -й момент показаны на рис. 4 стрелками. Они характеризуются переходными вероятностями  $P_{XX}$ , являющимися элементами переходной матрицы изображенного на рис. 4 графа состояний:

$$P_{Пi} = \begin{bmatrix} P_{HH} & P_{HK} & 0 & 0 & 0 \\ 0 & 0 & P_{K0} & P_{K1} & P_{K2} \\ P_{0H} & 0 & P_{00} & P_{01} & P_{02} \\ P_{1H} & 0 & P_{10} & P_{11} & P_{12} \\ P_{2H} & 0 & P_{20} & 0 & P_{22} \end{bmatrix}.$$

Нулевые элементы матрицы  $P_{Пi}$  соответствуют невозможным переходам. В этой связи отметим невозможность самопроизвольного восстановления готовности навигационной системы без привлечения внешней информации, т. е. перехода из состояния  $S_2$  в  $S_{1*}$ . Ненулевые вероятности  $P_{XX}$  зависят от характеристик потоков случайных событий в интегрированной системе, среди которых выделим наиболее значимые:

$E_{КП}$  — событие прохождения объектом КП и коррекции навигационной системы;

$E_{в.и}$  — событие привлечения внешнего источника информации, например отрезка ЦКД для коррекции навигационной системы;

$E_{в.о}$  — событие ошибки, возникшей в процессе использования внешнего источника информации;

$E_{с.н}$  — событие — момент выхода погрешностей за установленные допуски вследствие их накопления в автономно функционирующей системе навигации.

Указанные события позволяют определять переходные вероятности в общем — как вероятности их реализации



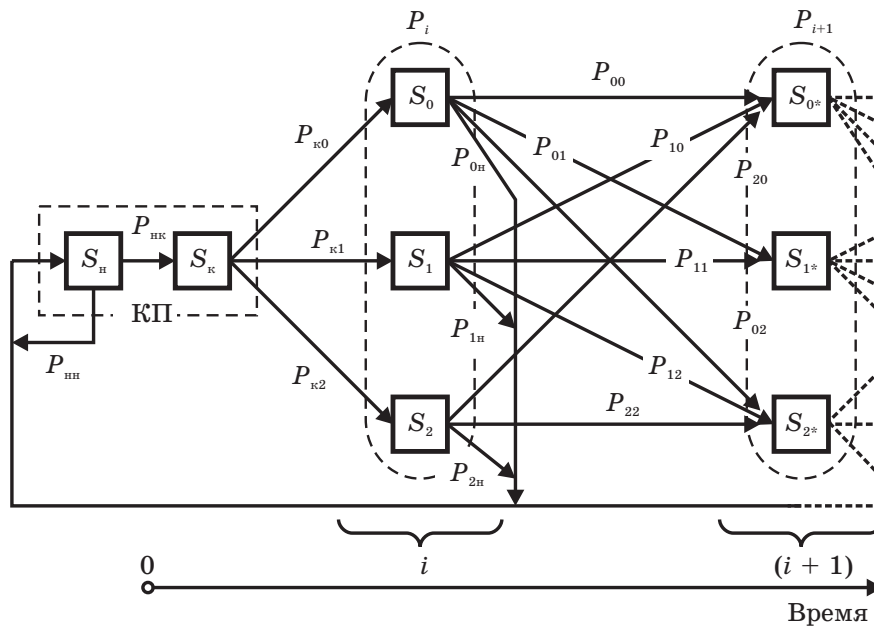


Рис. 4. Граф переходов процесса функционирования интегрированной навигационной системы

$$P_{XX} = \pi(E_{с.н}, E_{в.и}, E_{в.о}, E_{КП}). \quad (9)$$

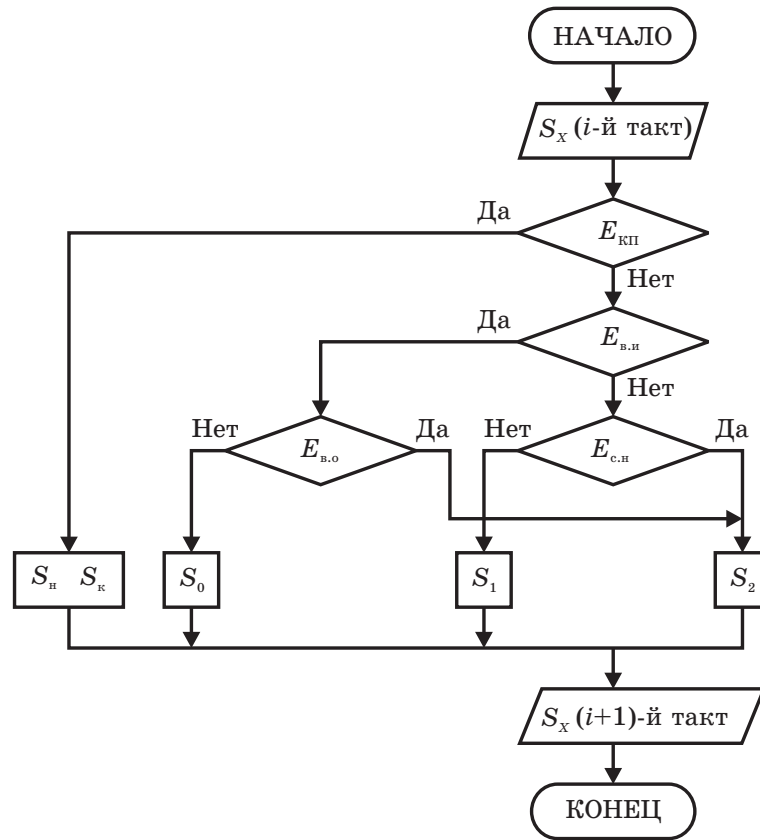
Обоснование выражений для нахождения переходных вероятностей. Конкретизируем  $P_{XX}$  применительно к рассматриваемой системе. Для этого схематически представим алгоритм функционирования интегрированной системы в произвольном такте (рис. 5), рассматривая только указанные события. На рисунке результат проверки условий положителен, если случайное событие реализуется в данном такте. Из рисунка видно, что в данной системе наивысшим приоритетом обладает информация КП, поскольку в случае ее доступности другие внешние источники информации не востребуются, и состояние погрешностей навигационной системы не имеет значения. Вероятность  $\pi_{КП} = \pi(E_{КП})$  того, что в данном такте навигационной системы объект окажется на КП, достаточно мала. Поэтому в отсутствие КП могут привлекаться внешние источники, такие как рельеф, ЦКД и пр. В среднем, вероятность  $\pi_{в.и} = \pi(E_{в.и})$  этого события в такте превосходит  $\pi_{КП}$ , поскольку КП располагаются на довольно больших расстояниях друг от друга. Здесь имеет место, пусть достаточно малая, вероятность  $\pi_{в.о} = \pi(E_{в.о})$  ошибок, приводящих к снижению достоверности информации. Примером может служить ошибка выбора дороги на перекрестке в известных алгоритмах коррекции систем навигации по ЦКД [9]. Наконец, если по объективным причинам внешняя информация отсутствует, система навигации функционирует в автономном режиме, характеризующемся увеличением погрешностей и, как следствие, снижением достоверности  $\pi_{с.н} = \pi(E_{с.н})$ . Значе-

ния вероятностей  $\pi_{с.н}$  могут быть найдены как функции от пройденного пути  $L$  известными способами [12], основанными на решении уравнения Фоккера—Планка—Колмогорова, которому удовлетворяет функция  $F_{с.н}(L)$  распределения вероятностей нахождения погрешностей ННС в заданных допусках.

Таким образом, в рассматриваемой интегрированной системе можно выделить события  $E_{КП}$ ,  $E_{в.и}$ , повышающие достоверности информации, и события  $E_{в.о}$ ,  $E_{с.н}$ , приводящие к ее снижению. Эти случайные события в каждом такте работы системы характеризуются соответствующими безусловными вероятностями  $\pi_{КП}$ ,  $\pi_{в.и}$ ,  $\pi_{в.о}$ ,  $\pi_{с.н}$  и могут реализовываться в строго определенной последовательности:  $E_{КП}$ ,  $E_{в.и}$ ,  $E_{в.о}$ ,  $E_{с.н}$ , причем возможность реализации последующих событий зависит от исходов предыдущих. Поэтому искомые переходные вероятности могут быть найдены на основе формулы (9) как произведения соответствующих условных вероятностей

$$P_{XX} = \pi(E_{КП}) \pi(E_{в.и} | E_{КП}) \times \pi(E_{в.о} | E_{КП}, E_{в.и}) \pi(E_{с.н} | E_{КП}, E_{в.и}, E_{в.о}). \quad (10)$$

Условные вероятности определяются в соответствии с алгоритмом функционирования навигационной системы (см. рис. 5). Условная вероятность  $\pi(E_{в.и} | E_{КП})$  численно равна  $\pi_{в.и}$ , если не реализуется событие  $E_{КП}$ , т. е. в данном такте работы навигационной системы объект не оказался на КП. При нахождении объекта на КП (событие  $\bar{E}_{КП}$ , вероятность которого  $\pi(\bar{E}_{КП}) = 1 - \pi(E_{КП})$ ) значение условной вероятности равно единице:



■ Рис. 5. Алгоритм функционирования интегрированной системы в произвольном такте

$$\pi(E_{в.и} | E_{кп}) = \begin{cases} \pi_{кп} & \text{при } \bar{E}_{кп}, \\ 1 & \text{при } E_{кп}. \end{cases}$$

Аналогично определяются и остальные условные вероятности в формуле (10), как функции от исходов событий  $E_{кп}$ ,  $E_{в.и}$  и  $E_{в.о}$ . Они позволяют находить вероятности  $P_{XX}$ , зная последовательности исходов событий  $E_{кп}$ ,  $E_{в.и}$ ,  $E_{в.о}$ ,  $E_{с.н}$ , приводящих интегрированную систему к одному из состояний  $S_0, S_1, S_2$  или  $S_n, S_k$  (см. рис. 5). Последовательности исходов событий для соответствующих переходных вероятностей и выражения по их нахождению приведены в табл. 2. Знаками «+», «-» и «~» отмечены положительные, отрицательные и любые исходы соответственно. Рассмотрим полученные выражения. Сумма вероятностей переходов из какого-либо состояния в другие равна единице:  $\sum P_{0X} = \sum P_{1X} = \sum P_{2X} = 1$ , поскольку все переходы образуют полную группу событий в каждом такте работы интегрированной системы.

Одинаковые выражения для нахождения некоторых переходных вероятностей не означают равенства их числовых значений в конкретных навигационных алгоритмах. Так, например, вероятность  $\pi_{в.и}$  использования внешнего источника в виде отрезка ЦКД зависит от состояния системы.

При движении наземного объекта по дороге (состояние  $S_0$  в начале  $i$ -го такта — см. рис. 5) значение  $\pi_{в.и}$  больше, чем при движении по бездорожью (состояние  $S_1$ ). Это объясняется тем, что до момента выхода на какую-либо дорогу и коррекции навигационной системы объект некоторое время движется по бездорожью, где использование данного источника информации невозможно. Следовательно, значения вероятностей  $P_{00}, P_{10}$  различаются. В общем, значения безусловных вероятностей зависят от состояний системы, из которых реализуются переходы. Обращаясь к рис. 5, можно сказать, что результаты проверки всех условий (вероятности исходов) зависят от входного состояния  $S_x$  в начале  $i$ -го такта. Как видно из таблицы, переходные вероятности  $P_{2X}$  не зависят от исхода события  $E_{с.н}$ , что отражает факт невозможности восстановления готовности навигационной системы без привлечения внешних источников информации. Переходные вероятности  $P_{2X}$  являются частными случаями переходных вероятностей  $P_{0X}$  при  $\pi_{кп} \rightarrow 0$ . Физически это означает, что вероятность коррекции на двух различных КП в двух последовательных тактах работы навигационной системы исчезающе мала.

Таким образом, искомые переходные вероятности могут быть рассчитаны в соответствии с фор-

Таблица 2. Выражения для нахождения переходных вероятностей

Переходные вероятности	Случайные события				Выражение для нахождения переходной вероятности
	$E_{\text{КП}}$	$E_{\text{в. и}}$	$E_{\text{в. о}}$	$E_{\text{с. н}}$	
$P_{0\text{н}}, P_{1\text{н}}$	+	~	~	~	$\pi_{\text{КП}}$
$P_{00}, P_{10}$	-	+	-	~	$(1 - \pi_{\text{КП}}) \pi_{\text{в. и}} (1 - \pi_{\text{в. о}})$
$P_{01}, P_{11}$	-	-	~	-	$(1 - \pi_{\text{КП}}) (1 - \pi_{\text{в. и}}) (1 - \pi_{\text{с. н}})$
$P_{02}, P_{12}$	-	+	+	~	$(1 - \pi_{\text{КП}}) \pi_{\text{в. и}} \pi_{\text{в. о}}$
	-	-	~	+	$(1 - \pi_{\text{КП}}) (1 - \pi_{\text{в. и}}) \pi_{\text{с. н}}$
$P_{2\text{н}}$	+	~	~	~	$\pi_{\text{КП}}$
$P_{20}$	-	+	-	~	$(1 - \pi_{\text{КП}}) \pi_{\text{в. и}} (1 - \pi_{\text{в. о}})$
$P_{22}$	-	+	+	~	$(1 - \pi_{\text{КП}}) \pi_{\text{в. и}} \pi_{\text{в. о}}$
	-	-	~	~	$(1 - \pi_{\text{КП}}) (1 - \pi_{\text{в. и}})$
$P_{\text{к}0}$	-	+	-	~	$\pi_{\text{в. и}} (1 - \pi_{\text{в. о}})$
$P_{\text{к}1}$	-	-	~	-	$(1 - \pi_{\text{в. и}}) (1 - \pi_{\text{с. н}})$
$P_{\text{к}2}$	-	+	+	~	$\pi_{\text{в. и}} \pi_{\text{в. о}}$
	-	-	~	+	$(1 - \pi_{\text{в. и}}) \pi_{\text{с. н}}$

мулами табл. 2 по известным безусловным вероятностям. Вероятности  $P_{\text{нн}}, P_{\text{нк}}$  определяются в соответствии с известными формулами [3] на основании временных характеристик коррекции навигационных систем на КП.

**Безусловные вероятности событий интегрированной системы.** Наиболее просто безусловные вероятности  $\pi_{\text{КП}}, \pi_{\text{в. и}}, \pi_{\text{в. о}}, \pi_{\text{с. н}}$  могут быть рассчитаны при предположении о постоянстве интенсивностей потоков соответствующих событий. Однако в решаемой задаче интенсивности зависят от пройденного пути  $L$  объекта. Общий для определения всех вероятностей подход рассмотрим на примере вероятности  $\pi_{\text{КП}}$ , зависящей от положения объекта между КП в  $i$ -й точке на оси  $L_{\text{КП}}$  (см. рис. 3) в соответствии с известной плотностью  $f_{\text{КП}}(L_{\text{КП}})$  (см. рис. 2).

Рассматриваемая система будет находиться в  $(i + 1)$ -й точке (см. рис. 3) в любом из состояний при условиях, что она находилась в  $i$ -й точке и реализовался переход из любого состояния  $i$ -й точки в какое-либо состояние  $(i + 1)$ -й точки. Данный переход осуществится, если в  $(i + 1)$ -й точке не реализуется событие  $E_{\text{КП}}$ , т. е. не «встретится» КП. Поэтому его вероятность равна  $(1 - \pi_{\text{КП}})$ . Далее, система будет находиться в  $(i + 1)$ -й точке в любом состоянии, если на отрезке  $[0; i\Delta tV]$  оси  $L_{\text{КП}}$  не «встретился». Следовательно, вероятность  $P_i$  нахождения системы в любом состоянии в  $i$ -й точке определяется в соответствии с плотностью  $f_{\text{КП}}(L_{\text{КП}})$

$$P_i = 1 - \int_0^{i\Delta tV} f_{\text{КП}}(L_{\text{КП}}) dL_{\text{КП}} = 1 - F_{\text{КП}}(i\Delta tV),$$

где  $F_{\text{КП}}$  — соответствующая  $f_{\text{КП}}(L_{\text{КП}})$  функция распределения вероятностей. Аналогично,  $P_{i+1} = 1 - F_{\text{КП}}((i + 1)\Delta tV)$ . С другой стороны,  $P_{i+1} = P_i(1 - \pi_{\text{КП}})$ , откуда

$$\pi_{\text{КП}} = 1 - \frac{P_{i+1}}{P_i} = 1 - \frac{1 - F_{\text{КП}}((i + 1)\Delta tV)}{1 - F_{\text{КП}}(i\Delta tV)}.$$

Аналогично, с использованием функции  $F_{\text{с. н}}(L)$ , находятся и вероятности  $\pi_{\text{с. н}}$ .

**Определение финальных вероятностей и оценок достоверности**

В общем случае значения  $P(j)$  не являются постоянными при изменении  $j$  (пройденного пути или времени). Если предположить значения элементов переходных матриц  $\mathbf{P}_{\text{Хн}}$  близкими к действительным, то на начальном этапе движения объекта изменчивость  $P(j)$  может быть обусловлена, в основном, начальными значениями вероятностей  $P(0)$  (5). С учетом этого определен интерес представляет нахождение установившихся (финальных) вероятностей состояний, не зависящих от начальных условий.

Финальные вероятности в общем случае найдем по определению [3], т. е. из условия  $\mathbf{P}_S(j) \cong \mathbf{P}_S(j + 1)$ . В соответствии с рекуррентным соотношением (6)



финальные вероятности должны удовлетворять равенству

$$\left(\mathbf{I} - (\mathbf{P}_{ХП})^T\right) \mathbf{P}_S^* = \mathbf{0}, \quad (11)$$

где  $\mathbf{I}$  — единичная матрица соответствующей размерности;  $\mathbf{P}_S^*$  имеет структуру вектора  $P_S(j)$  (6). Чтобы вычислить  $\mathbf{P}_S^*$ , необходимо учесть нормировочное условие, согласно которому сумма финальных вероятностей всех состояний равна единице. В соответствии с ним элементы любой (например, последней) строки матрицы  $\left(\mathbf{I} - (\mathbf{P}_{ХП})^T\right)$  и соответствующий элемент нулевого вектора в правой части (11) необходимо заменить единицами. Решение

полученной после замены системы линейных уравнений даст финальные вероятности  $\mathbf{P}_S^*$  всех состояний системы, по которым, воспользовавшись формулами (7) и (8), можно вычислить финальную вероятность  $\mathbf{P}^*$  того, что погрешности координат не выйдут за установленные допуски.

Заметим, что размерность системы линейных уравнений (11) может быть достаточно большой. Тем не менее, большинство элементов матрицы  $\mathbf{P}_{ХП}$  равно нулю, и ее число обусловленности не является критичным. Как следствие, при численном решении системы (11) (с учетом нормировочного условия) традиционными методами на современных ЭВМ не возникает проблемы потери точности и не требуется значительных вычислительных затрат.

### Литература

1. **Интегрированные** инерциально-спутниковые системы навигации / Под общ. ред. В. Г. Пешехонова; ГНЦ РФ-ЦНИИ «Электроприбор». СПб., 2001. 235 с.
2. **Степанов О. А.** Применение теории нелинейной фильтрации в задачах обработки навигационной информации / ГНЦ РФ-ЦНИИ «Электроприбор». СПб., 1998. 370 с.
3. **Феллер В.** Введение в теорию вероятностей и ее инженерные приложения. В 2 т. Т.1: Пер. с англ. М.: Мир, 1984. 528 с.
4. **Дмитриев С. П., Колесов Н. В., Осипов А. В.** Информационная надежность, контроль и диагностика навигационных систем / ГНЦ РФ-ЦНИИ «Электроприбор». СПб., 2003. 207 с.
5. **Гришин Ю. П., Казаринов Ю. М.** Динамические системы, устойчивые к отказам. М.: Радио и связь, 1985. 176 с.
6. Пат. 2165075 РФ, G 01 C 21/00. Наземная навигационная система.
7. **Беркович С. Б. и др.** Коррекция автономных наземных навигационных систем в движении по отдельным точкам цифровой карты дороги // XIII Санкт-Петербургская Междунар. конф. по интегрированным навигационным системам: Тез. докл. / ГНЦ РФ-ЦНИИ «Электроприбор». СПб., 2006. С. 65.
8. Пат. 2184349 РФ, G 01 C 21/12. Аппаратурачисления координат с автоматической коррекцией показаний.
9. **Дмитриев С. П. и др.** Оптимальное решение задачи автомобильной навигации с использованием карты дорог // Гироскопия и навигация. 2000. № 2(29). С. 57.
10. **Mezentsev O., Collin J., Lachapelle G.** Vehicular navigation in urban canyons using a high sensitivity GPS receiver augmented with a medium-grade IMU // X St.-Petersburg International Conference on Integrated Navigation Systems / CSRI «Elektroprigor». SPb., 2003. P. 50.
11. **Шолохов А. В.** Коррекция наземных навигационных систем по цифровой карте дорог с учетом ее погрешностей // V науч.-техн. конф. молодых ученых «Навигация и управление движением»: Сб. докл. / ГНЦ РФ-ЦНИИ «Электроприбор». СПб., 2003. С. 227–233.
12. **Левин Б. Р.** Теоретические основы статистической радиотехники. Кн. 1. М.: Сов. радио, 1969. 752 с.
13. **Кулак А. В.** Уточнение координат местоположения наземных объектов по измерениям абсолютной высоты, с привлечением данных цифровой модели рельефа // Гироскопия и навигация. 2005. № 50. С. 95.

УДК 681.327.8

## КОНЦЕПЦИЯ ПОСТРОЕНИЯ АРХИТЕКТУРЫ МОДЕЛЕЙ ПРОЦЕССОВ УПРАВЛЕНИЯ ИНФОКОММУНИКАЦИОННЫМИ СЕТЯМИ (Часть 1)

**В. Д. Нестеренко,**

канд. техн. наук, начальник отдела новых технологий  
ОАО «Северо-Западный Телеком»

*Предложена архитектура инфокоммуникационной сети как объекта управления, основанная на модели мультисреды, охватывающей среды взаимодействия, генерации, распространения инфокоммуникационных сетей и позволяющая провести структуризацию моделей и алгоритмов их адаптивного управления.*

*The paper proposes an architecture for the infocommunication network as an object of control which is based on a model of composed media including the media of interaction, generation and interaction of infocommunication systems that allows the structuring of both the models and the algorithms of their adaptive control.*

Необходимым условием формирования информационной инфраструктуры современного общества является создание инфокоммуникационных сетей (ИКС), которые образуются за счет конвергенции информационных и телекоммуникационных технологий на базе достижений в области микроэлектроники, фотоники и радиотехники [1–3]. Внедрение ИКС позволит создать инструментальную платформу для построения глобального информационного пространства, изменить не только способ производства продуктов и услуг, но и формы организации досуга, воспитания и образования, окажет решающее воздействие на социальную структуру общества, экономику, политику, развитие общественных институтов.

В отличие от сетей предыдущего поколения, ИКС должна обеспечивать не только передачу информации между пользователями, но и доступ пользователей к информационным услугам и контенту, что меняет концепцию сетевого управления. Соответствующие процессы реализуют сетевые мультисервисные центры обработки информации и управления нового типа, которые представляют собой систему взаимосвязанных во времени и пространстве протокольных, вычислительных, алгоритмических и коммутационных средств и являются необходимым элементом ИКС независимо от сетевой архитектуры и принципа интеллектуализации сети.

Невозможно для освещения проблемы не сказать несколько слов об истории создания мультисервисной сети ОАО «Северо-Западный Телеком» (ОАО «СЗТ»).

сервисной сети ОАО «Северо-Западный Телеком» (ОАО «СЗТ»).

1997 г. — построение сети передачи данных X25/FR/ATM филиала Петербургской телефонной сети (ПТС);

2001 г. — СПД ОАО ПТС становится системообразующим ядром для реализации проекта мультисервисной сети связи (МСС) на базе ATM-оборудования и системы управления фирмы Alcatel;

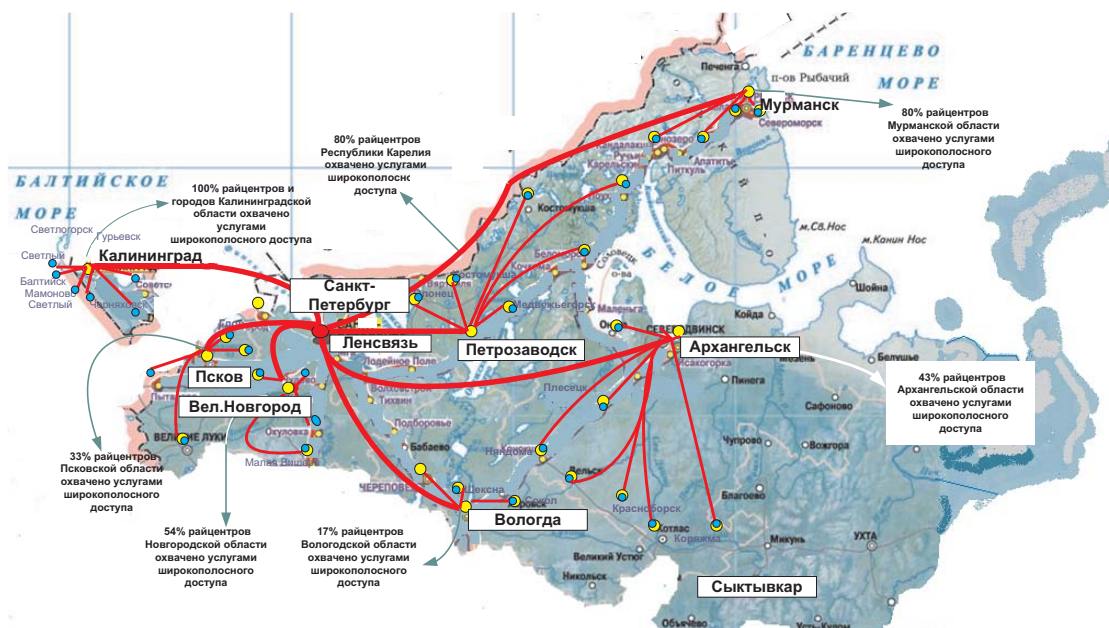
2003 г. — реализовано ядро МСС «СЗТ» и установлены региональные узлы магистрального уровня;

2004–2006 гг. — развитие региональных сетей передачи данных, построение ATM/IP MPLS сети ОАО «СЗТ».

Перспективы развития ИКС показаны на рис. 1.

Общесистемными требованиями к развитию ИКС являются:

- соответствие всего оборудования требованиям, предъявляемым к оборудованию операторского класса;
- обеспечение гарантированного качества для различных видов услуг (передача голоса, данных, видео);
- гарантированное обеспечение информационной безопасности пользовательских каналов;
- наличие развитых механизмов контроля выполнения соглашения о пользовательском трафике;
- универсальность базовой технологии переноса и коммутации любых информационных потоков;



■ Рис. 1. Перспективы развития ИКС

- возможность эффективно консолидировать существующие разнородные сети (X.25, Frame-Relay, выделенные каналы, IP и т. п.).

Как видно из истории и тенденций развития ИКС ОАО «Северо-Западный Телеком», наиболее характерной особенностью является усложнение самой сети и, естественно, усложнение ее формального описания для возможности прогнозирования развития и создания наиболее эффективных вариантов сети, способных адаптироваться при решении задач управления к стохастическим потокам пользователей. Это делает актуальным проработку концепции построения архитектуры моделей процессов управления ИКС, которая даст возможность увязать модели (аналитические, имитационные), полученные в различных предметных областях (вычислительной технике, технике связи, помехоустойчивом кодировании, информационной безопасности и др.), которые на данном этапе пересеклись при описании ИКС и требуют создания своего рода «математического интерфейса» по аналогии с реальными интерфейсами, принятыми в эталонной модели взаимодействия открытых систем (ЭМВОС) [1].

Проведем формальное описание архитектуры ИКС как объекта управления, рассматривая взаимодействие внешней среды и элементов сети между собой и обработку цифровой информации каждым из элементов этой сети в соответствии с принципами ЭМВОС в виде внешней среды  $E$  (от англ. environment — среда) и сетевой среды  $N$  (от англ. net — сеть). Каждую из них представим в виде двух сред: среды генерации, или  $GEN(E)$ -среды (от англ. generation — генерация), среды распространения, или  $WID(E)$ -среды (от англ. widespread —

распространение), среды взаимодействия, или  $INT(N)$ -среды — (от англ. interaction — взаимодействие) и среды обработки, или  $PRO(N)$ -среды (от англ. processing — обработка) (рис. 2).

Выделение  $GEN(E)$ -среды должно позволить описать воздействие пользователей на сеть, т. е. потоков их запросов на ресурсы ИКС;  $WID(E)$ -среда должна позволить отразить воздействие внешних мешающих факторов (помех в канале связи (КСв), сбоев и т. п.) на распространение физических сигналов (электрических, оптических и т. д.);  $INT(N)$ -среда даст возможность описать сеть в целом как единую систему, взаимодействующую с внешней средой на системном уровне ( $GEN(E)$ - $INT(N)$ -уровне);  $PRO(N)$ -среда даст возможность описать основные структурные элементы сети, обрабатывающие цифровую информацию: узлы коммутации, концентраторы, абонентские пункты, тракты передачи и т. п. — с учетом взаимодействия с внешней средой на реализационном (аппаратно-программном) уровне ( $WID(E)$ - $PRO(N)$ -уровне).

Основное внимание в статье будет уделено управлению ядром ИКС, т. е. магистральной ее частью. Соответствующие процессы реализуют сетевые мультисервисные центры обработки информации и управления (ЦОУ) нового типа, которые представляют собой систему взаимосвязанных во времени и пространстве протокольных, вычислительных, алгоритмических и коммутационных средств и являются необходимым элементом ИКС независимо от сетевой архитектуры и принципа интеллектуализации сети [1].

Рассмотрение особенностей формализации выделенных сред начнем с  $INT(N)$ -среды, так как она является связующей основой для остальных сред,



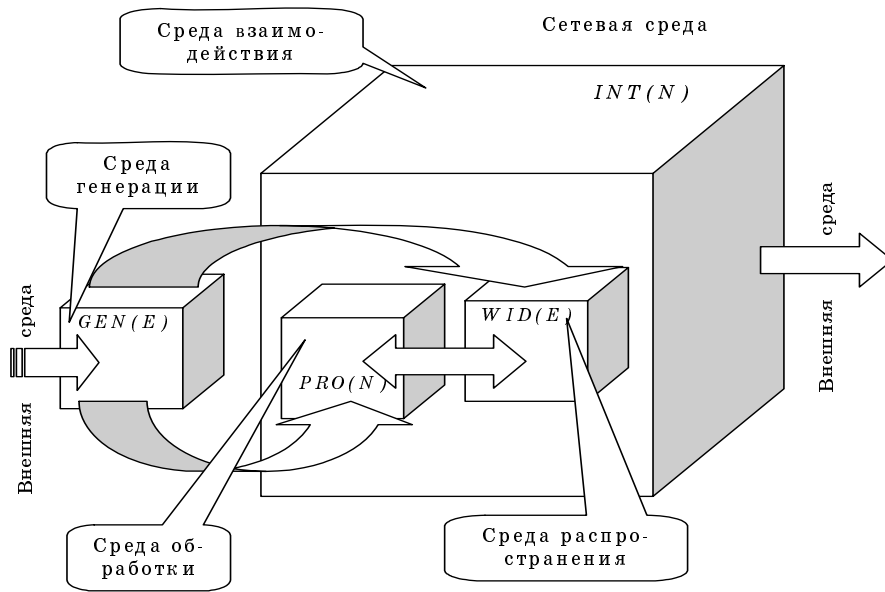


Рис. 2. Условно предлагаемая архитектура взаимодействия сред ИКС

с каждой из которых она взаимодействует. Представим  $INT(N)$ -среду как некоторую однородную или регулярную информационно-вычислительную среду [ 1 ], состоящую из ЦОУ, формализуемых в виде многополюсников  $\mu_i^n$  с тремя типами входов и выходов  $\mu_{ki}^n, \mu_{ii}^n, \mu_{ij}^n$ , у которых число входов соответствует числу выходов, так как, согласно ЭМВОС, все соединения между открытыми системами являются дуплексными.  $INT(N)$ -среда имеет сложную структуру, позволяющую изображать иерархию в ИКС, т. е. выделять особенности взаимодействия ЦОУ внутри данного уровня иерархии и между соседними уровнями. При этом

$$\mu_i^n = \{\mu_{1i}^n, \mu_{2i}^n, \dots, \mu_{i(i-1)}^n, \mu_{ii}^n, \mu_{i(i+1)}^n, \dots, \mu_{in}^n\},$$

где  $i$  — номер уровня данного многополюсника:  $i = 1, I_n, I_n$  — число уровней иерархии;  $n$  — номер ЦОУ в множестве узлов  $i$ -го уровня  $M_i$ ;  $n = 1, N_i, N_i$  — число узлов  $i$ -го уровня иерархии. Иногда удобно рассматривать только связи между соседними уровнями

$$M_i^n = \{M_{i(i-1)}^n, M_{ii}^n, M_{i(i+1)}^n\},$$

а связи с не соседними уровнями изображать с помощью фиктивных узлов

$$M_j^n = \{M_{i(j-1)}^n, 0, M_{i(j+1)}^n\}.$$

Если  $M_i^n$  — множество всех входов и выходов  $n$ -го узла уровня  $i$   $INT(N)$ -среды, то  $M_{ki}^n \in M_i^n$  является подмножеством входов и выходов  $n$ -го узла  $j$ -го уровня, соединенных с узлами следующего нижнего уровня, а  $M_{ij}^n \in M_i^n$  — подмножество входов и выходов этого узла, связанных с узлами следующего верхнего  $j$ -го уровня. Каждое

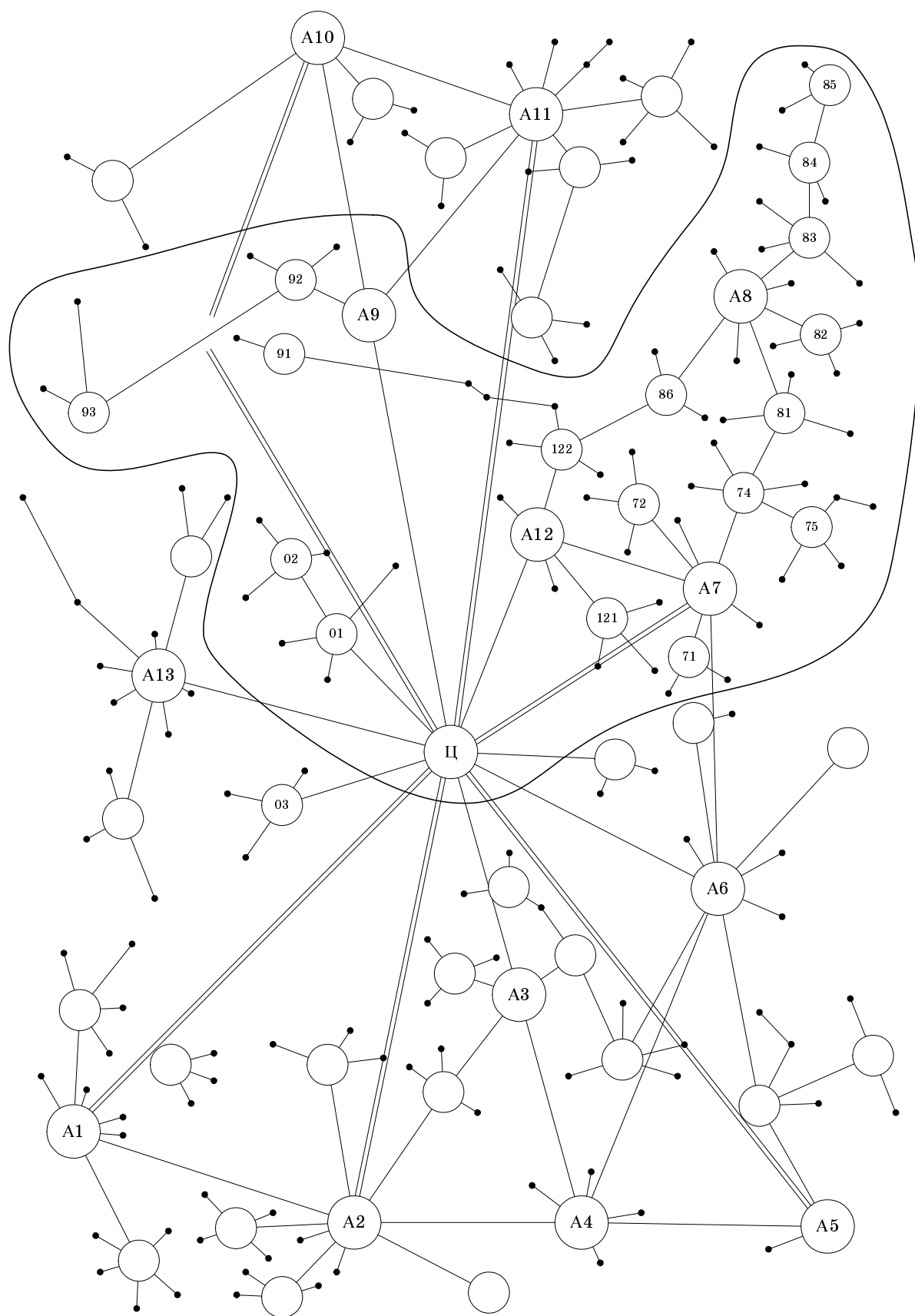
такое подмножество состоит из элементов  $M_{nk}$ :  $M_{ki}^n = \{M_{n1}, \dots, M_{nNk}\}$ .

Элементы, т. е. ЦОУ, имеют переменную структуру, а именно: при  $M_i^n = \text{const}$   $M_{(i-1)i}^n = \text{var}$ ,  $M_{ii}^n = \text{var}$ ,  $M_{i(i+1)}^n = \text{var}$  и при этом  $M_i^n = M_{(i-1)i}^n \cup M_{ii}^n \cup M_{i(i+1)}^n$  (в общем случае  $M_i^n = \bigcup_{k=1}^{i-1} M_i^n \cup \bigcup_{j=i+1}^{I_n} M_{ij}^n$ , каждый  $M_{jk}$  может принимать значение от 0 до  $M_i^n$ ;  $M_i^n$  тоже может быть переменным).

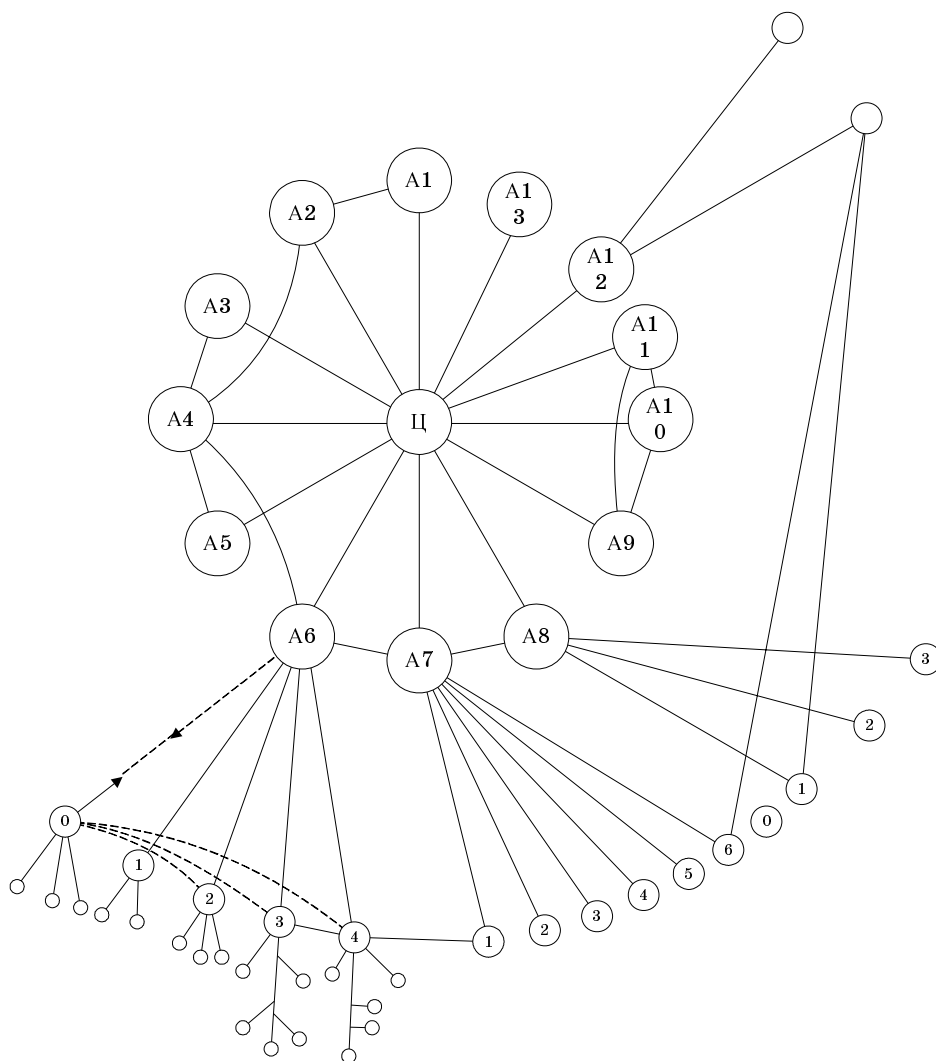
В терминах ЭМВОС рассматриваемые узлы  $INT(N)$ -среды являются совокупностями открытых систем, которые подчиняются следующим правилам:

- связь между объектами одного и того же уровня двух или более открытых систем в рамках узла  $INT(N)$ -среды может осуществляться через объекты смежных уровней, исключая физический;
- объекты открытых систем, находящиеся в разных узлах  $INT(N)$ -среды, соединяются между собой обязательно через физические уровни этих систем.

Для описания и анализа особенностей  $INT(N)$ -среды предлагается аппарат так называемых иерархических матриц связности (МС), которые будем называть  $H$ -схемами (от англ. hierarchical matrix connection). Структура любой сети может быть задана с помощью простой квадратной МС, размерность которой определяется числом узлов в сети  $N_y$ . В случае ИКС такая МС будет иметь большую размерность, к тому же применительно к каждому узлу она содержит много избыточной информации, но в явном виде в ней нельзя отразить факт наличия и особенности нескольких вторичных сетей (см. рис. 1).



■ Рис. 3. Пример произвольной радиально-кольцевой четырехуровневой ИКС: — — новая магистраль



■ Рис. 4. Фрагмент структуры ИКС

Эти недостатки можно устранить, если ввести понятие *H*-схем, представляющих собой совокупность квадратных и прямоугольных МС. Определим понятие *H*-схемы на примере произвольной радиально-кольцевой четырехуровневой ИКС (рис. 3.), на основе которой строится сеть для передачи интегральной информации с различными приоритетами. Характер доставляемых потоков информации, их направленность и приоритеты обуславливают упорядочивание ЦОУ, т. е. выделение среди множества региональных ЦОУ  $\{A_x\}$  главного ЦОУ *Ц*, для которого требуется обеспечить прямую связь со всеми региональными узлами, что осуществляется путем выделения на существующих магистралях специального некоммутируемого канала или внедрения новой магистрали.

На рис. 3 условно изображены четыре категории ЦОУ, соответствующие четырем уровням

иерархии. Для удобства дальнейшего рассмотрения фрагмент структуры (см. рис. 3) представлен без учета реальных расстояний и с выделением уровней иерархии в виде концентрических кругов около центрального узла *Ц* (рис. 4). Данное представление дает возможность увидеть, что на базе первичной сети без каких-либо дополнительных работ по созданию КСв можно построить сеть передачи информации с ярко выраженной радиальной структурой. Новые физические линии обозначены сплошными линиями, а организованные с целью создания кольцевой структуры виртуальные каналы — штриховыми линиями, т. е. пунктиром обозначены сквозные каналы не на существующих физических линиях, а на базе логических каналов через замыкающий узел верхнего уровня.

Узел 0 служит для обозначения устройства, находящегося в узле высшего уровня, через кото-



рый подключаются абоненты прямо к узлу, а не через узел нижнего уровня. Такое условное обозначение вводится для сохранения единства представления, что существенно при разработке протоколов обмена. Конфигурация узлов задается в матричной форме с использованием квадратных МС узлов данного уровня размерностью  $N_i \times N_i$ , где  $N_i$  — число узлов соответствующего  $i$ -го уровня, и прямоугольных матриц связности узлов смежных уровней размерностью  $N_i \times N_{i-1}$ , где  $N_{i-1}$  — число узлов нижнего уровня. Матрицы имеют вид

$$\|M_{ii}\| = \begin{bmatrix} m_{ii}^{11} & m_{ii}^{12} & \dots & m_{ii}^{1N_i} \\ m_{ii}^{21} & m_{ii}^{22} & \dots & m_{ii}^{2N_i} \\ \dots & \dots & \dots & \dots \\ m_{ii}^{N_i 1} & m_{ii}^{N_i 2} & \dots & m_{ii}^{N_i N_i} \end{bmatrix}.$$

Очевидно, в МС могут стоять численные значения некоторых параметров, характеризующих соответствующие связи. Матрица  $\|M_{ii}\|$  симметрична относительно главной диагонали, и ее элементами могут быть числа, которые обозначают число каналов в магистральной, прямо соединяющей узлы  $k$  и  $l$ . Если матрица относится к организации вторичных сетей, то ее можно представить в виде суммы матриц вторичных сетей, т. е.

$$\|M_{ii}\| = \|M_{ii}^I\| + \|M_{ii}^{II}\| + \dots + \|M_{ii}^{\bar{B}}\|,$$

где  $I, II, \dots, \bar{B}$  — индексы, обозначающие принадлежность к соответствующей вторичной сети. Матрицы  $\|M_{ii}^I\|, \dots, \|M_{ii}^{\bar{B}}\|$  будем называть вторичными.

Для отображения связей между узлами соседних уровней составляются прямоугольные МС вида

$$\|M_{i,i+1}\| = \begin{bmatrix} m_{i,i+1}^{11} & m_{i,i+1}^{12} & \dots & m_{i,i+1}^{N_i 1} \\ m_{i,i+1}^{12} & m_{i,i+1}^{22} & \dots & m_{i,i+1}^{N_i 2} \\ \dots & \dots & \dots & \dots \\ m_{i,i+1}^{1N_i+1} & m_{i,i+1}^{2N_i+1} & \dots & m_{i,i+1}^{N_i N_i+1} \end{bmatrix}.$$

Такие межуровневые МС могут быть первичными, отображающими физические соединения между узлами, и вторичными. В некоторых случаях МС можно рассматривать как булевы матрицы и применять к ним аппарат булевой алгебры. Также может быть применен аппарат преобразований булевых матриц к булевым определителям. Покажем, как МС формируется с помощью образующих векторов, содержащих упорядоченные символьные обозначения узлов, находящихся на одном и том же уровне, например:  $\mathbf{J} = (n_1, n_2, \dots, n_{N_i})$ , т. е. на уровне  $i$  имеется  $N_i$  узлов.

Определим вектор связности узла  $n_k$ :

$$\mathbf{J}_{n_k} = \left( \underbrace{m_{i,i+1}^{k1}, m_{i,i+1}^{k2}, \dots, m_{i,i+1}^{kN_{i+1}}}_{\text{столбец } \|M_{i,i+1}\|}, \underbrace{m_{ii}^{k1}, \dots, m_{ii}^{kN_i}}_{\text{строка } \|M_{ii}\|}, \underbrace{m_{i-1,i}^{1k}, m_{i-1,i}^{2k}, \dots, m_{i-1,i}^{N_{i-1}k}}_{\text{столбец } \|M_{i-1,i}\|} \right),$$

где  $i-1, i$  и  $i+1$  — соседние уровни иерархии сети.

На определенных таким образом МС, пользуясь некоторыми дополнительными средствами и понятиями, например понятиями ранга узла, ранга пути, сечения сети и ранга сечения сети, достаточно просто можно организовать поиск оптимального пути соединения двух абонентов, а также проводить оптимизацию структуры ИКС.

Для получения количественных оценок характеристик ИКС (и отдельных ее элементов) каждому узлу или связи между узлами приписывается вес — параметр, описывающий соответствующее свойство узла или связи, с указанием размерности. Для каждого параметра может быть составлена матрица, вхождения которой относятся либо к связям, либо к узлам (на главной диагонали). Прежде всего введем матрицы, характеризующие связи, которые еще называют матрицами морфологических характеристик (параметров): матрицы смежности, длин связей, пропускных способностей каналов, прямых каналов, надежности, кратчайших путей, пропускных способностей кратчайших путей, стоимости кратчайших путей, числа транзитов кратчайших путей и т. п.

Механизм формирования МС, т. е. формализацию процесса, иллюстрируют рис. 5, 6.

Целесообразно задавать топологию сети таблицей представительности, на основании которой легко можно строить МС, а затем составлять МС первичной сети. Связность всех узлов уровня  $i$  задается тремя основными МС:  $\|M_{i,i+1}\|, \|M_{ii}\|, \|M_{i-1,i}\|$ , а также добавочными МС (если таковые имеются). Добавочные МС отображают связи узлов данного уровня с узлами не соседних уровней  $\|M_{i+1+j,i}\|$  и  $\|M_{i-1-j,i}\|, j = 1, (I_n - i - 1)$ , где  $I_n$  — число уровней иерархии сети.

Элементы МС первичной сети соответствуют определенным элементам простой МС, в которой перечислены все узлы сети и связи между ними. Сокращение числа элементов при переходе к  $H$ -схемам происходит за счет уменьшения числа нулевых элементов простой МС. Если обозначить через  $E_1^0$  число элементов простой МС, а через  $E_2^0$  — число элементов  $H$ -схемы, то

$$E_1^0 = N^2 = (n_1 + n_2 + \dots + n_{I_n})^2, \quad n_1 \geq n_2 \geq \dots \geq n_{I_n},$$

где  $n_i$  — количество узлов на  $i$ -м уровне;

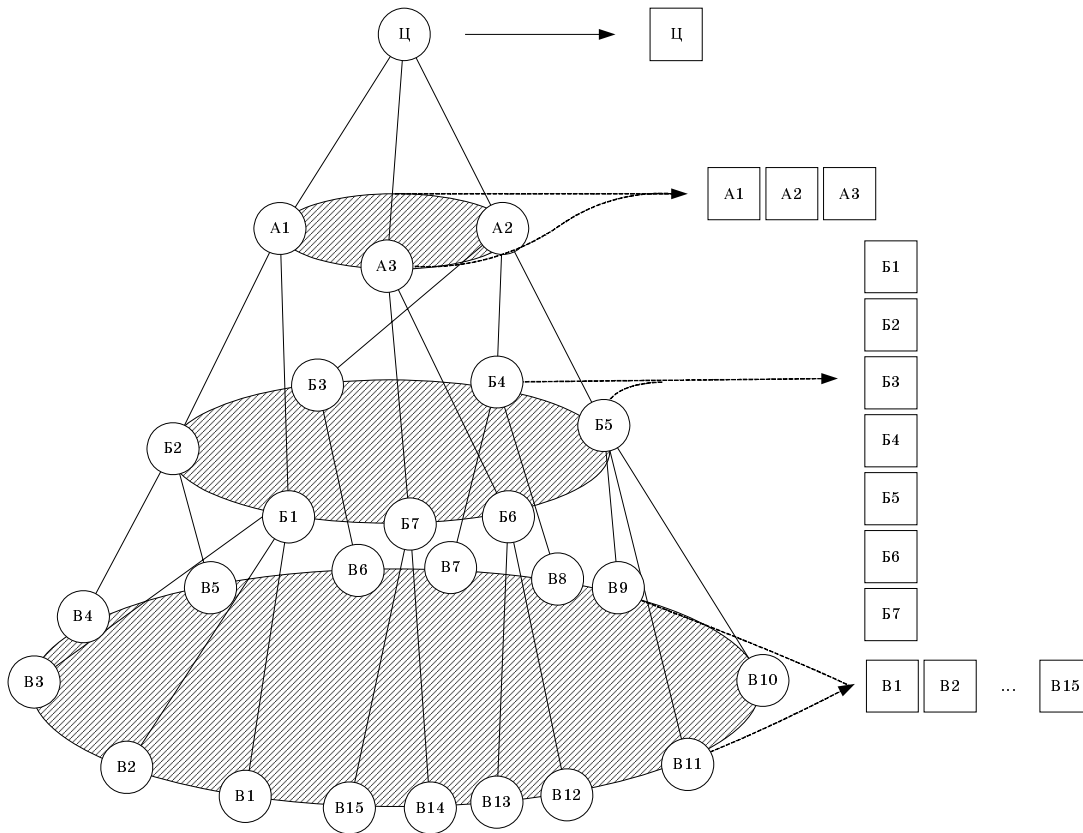


Рис. 5. Радиально-кольцевая структура ИКС

$$E_2^3 = \sum_{i=2}^{I_n} n_i^2 + \sum_{i=1}^{I_n-1} n_i n_{i+1}$$

при условии, что отсутствуют прямые соединения между абонентами;

$$E_2^{31} = \sum_{i=1}^{I_n} n_i^2 + \sum_{i=1}^{I_n-1} n_i n_{i+1}$$

в общем случае.

Очевидно,  $E_1^3 > E_2^{31} > E_2^3$  для всех  $I_n > 1$ , причем разница между  $E_1^3$  и  $E_2^3$  увеличивается с ростом  $I_n$  за счет увеличения вероятности получения нулевых добавочных МС, которые в  $E_2^3$  не присутствуют.

Полные МС образуются присоединением к основному МС добавочных МС следующим образом. На уровнях, через которые проходит прямое соединение без коммутации, создаются фиктивные узлы со связностью  $v = 2$  (см. рис. 4). Под связностью узла  $v$  понимаем число узлов, с которыми данный узел имеет прямое физическое соединение. Затем производятся операции добавления этих фиктивных узлов к соответствующим основным МС. Операция добавления определяется ниже.

С помощью  $H$ -схем можно описывать также организацию и взаимодействие между вторичными (виртуальными) сетями, созданными на базе

одной первичной сети. Отметим, что ИКС для пользователей является вторичной сетью.

Каждой виртуальной сети, организованной на базе определенной первичной сети, заданной вышеуказанными МС, сопоставляют МС виртуальной сети — виртуальные МС (ВМС). Виртуальные сети могут охватывать только определенную часть (регион) первичной сети. Для получения ВМС над полными МС первичной сети предлагается производить следующие операции.

1. Исключение узла  $k$  из уровня  $i$   $\tilde{E}_x^{ik}$ :

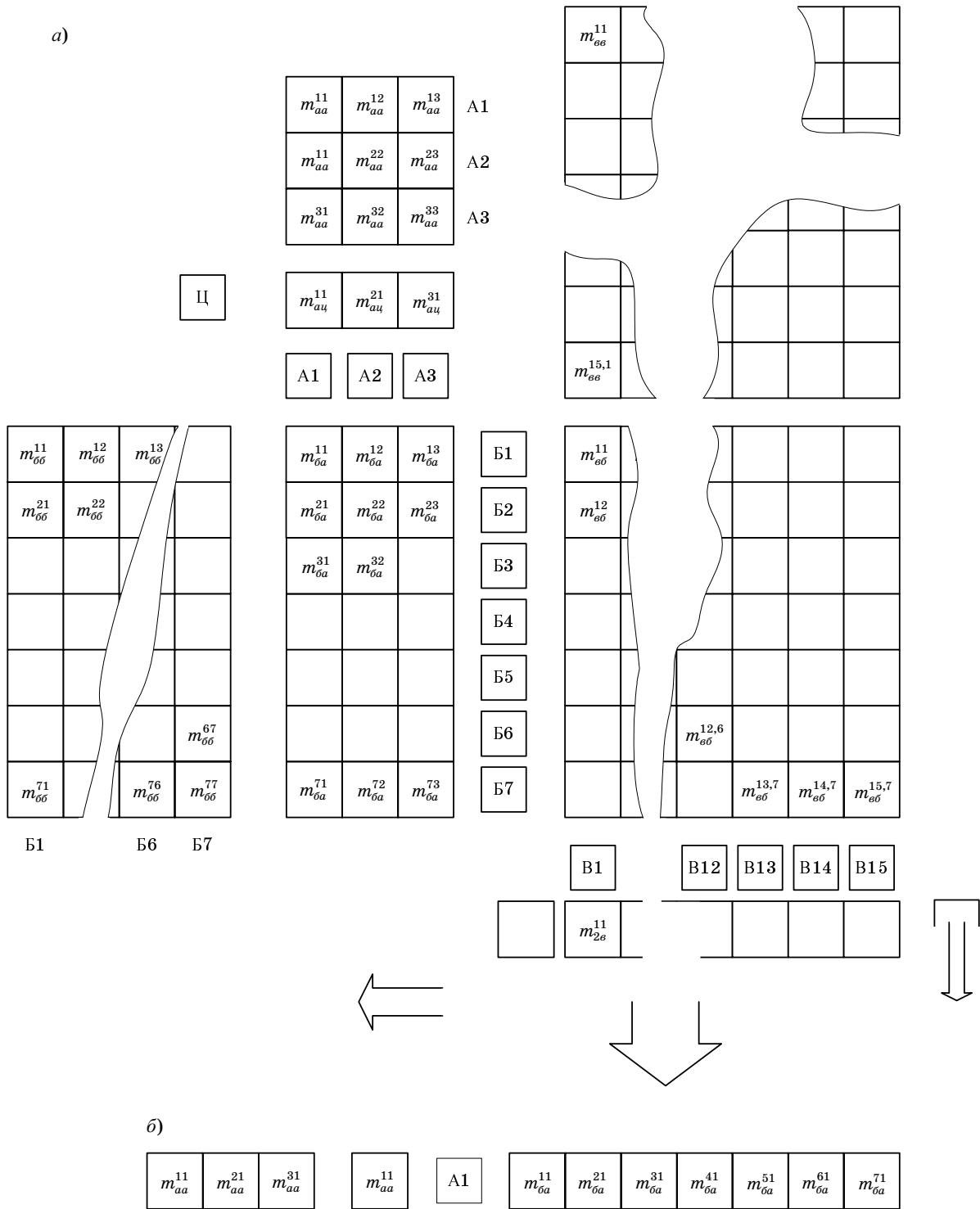
$\|M'_i\| = \tilde{E}_x^{ik} \|M_i\|$ ,  $N'_i = N_i - 1$ ,  $i = \overline{1, I_n}$ ,  $I_n = \text{const}$ , которое заключается в вычеркивании всех строк и столбцов МС, для которых узел  $k$  является образующим.

2. Добавление узла  $k$  к уровню  $i$   $\tilde{A}_d^{ik}$ :

$\|M'_i\| = \tilde{A}_d^{ik} \|M_i\|$ ,  $N'_i = N_i + 1$ ,  $i = \overline{1, I_n}$ ,  $I_n = \text{const}$ , которое заключается в добавлении новых строк и столбцов, организованных узлом  $k$ .

3. Понижение уровня  $k$ -го узла уровня  $i$   $\tilde{D}_n^{ik}$ :

$$\|M'_i\| = \tilde{D}_n^{ik} \|M_i\|, N'_i = N_i, N'_{i-1} = N_{i-1} + 1, i = \overline{1, I_n}, I_n = \text{var},$$



■ Рис. 6. Матрицы связности радиально-кольцевой структуры ИКС: а — исходная МС; б — вектор связности

когда узел верхнего уровня переводят на нижний уровень.

4. Повышение уровня  $k$ -го узла уровня  $i$   $\tilde{U}_p^{ik}$ :

$$\|M'_i\| = \tilde{U}_p^{ik} \|M_i\|, N'_i = N_i, N'_{i+1} = N_{i+1} + 1,$$

$$i = \overline{1, I_n}, I_n = \text{var},$$

когда узел с нижнего уровня переводится на высший уровень.

Одноместные операции исключения и добавления узла будем называть операциями модификации  $H$ -схем, а одноместные операции понижения и повышения уровня узла — операциями рекон-



фигурации  $H$ -схем. На  $H$ -схемах можно определить также и двухместные операции реорганизации, наложения и расслоения. В простейшем случае операции реорганизации сводятся к сложению и вычитанию матриц.

Учитывая ограниченный объем статьи, лишь кратко отметим возможности формализации остальных сред архитектуры ИКС.

Дадим формальное представление отдельных структурных элементов  $PRO(N)$ -среды, т. е. объектов открытых систем, из которых формируются ЦОУ сети. Эта задача достаточно сложна из-за многочисленности и разнообразия функциональных и структурных связей между объектами открытых систем, что не дает возможности воспользоваться предложением о построении однородной  $PRO(N)$ -среды на базе простых элементов.

Для управления использованием ресурсов ИКС в ЦОУ в соответствии с требуемыми направлениями доставки цифровой информации применяются маршрутные таблицы (МТ), сформированные на основе вектора связности ЦОУ и соответствующих ВМС, описывающих  $PRO(N)$ -среду. Столбцы этой таблицы заполняются на базе соответствующих

значений вектора связности, строки же соответствуют трем категориям ЦОУ, с которыми данный ЦОУ связывается, — верхнего, данного и нижнего уровня иерархии. Элементами МТ являются матрицы, строки которых относятся к ЦОУ соответствующего уровня, а столбцы — к КСв, ведущим к этим уровням.

Матрицы  $\|M_{11}\|$  и  $\|M_{12}\|$  формируются на базе строки (столбца)  $H$ -схемы  $\|M_{i,i+1}\|$ . Матрицы  $\|M_{21}\|$ ,  $\|M_{22}\|$  и  $\|M_{23}\|$  формируются на базе столбца (строки)  $H$ -схемы  $\|M_{i,i}\|$ , а матрицы  $\|M_{32}\|$  и  $\|M_{33}\|$  — на базе столбцов (строк)  $H$ -схемы  $\|M_{i-1,i}\|$ .

Рассмотренный подход применительно к  $N$ -среде, т. е. среде взаимодействия, позволит на этой основе комплексно рассмотреть взаимодействие и остальных сред, т. е. среды генерации, или  $GEN(E)$ -среды, среды распространения, или  $WID(E)$ -среды, и среды обработки, или  $PRO(N)$ -среды.

Таким образом, предложенный подход, базирующийся на представлении архитектуры процессов в ИКС в виде взаимосвязанных сред и математических схем их описания, позволит строить обобщенные модели ИКС как объекта управления.

## Литература

1. Колбанёв М. О., Яковлев С. А. Модели и методы оценки характеристик обработки информации в интеллектуальных сетях связи. СПб.: СПбГУ, 2002. 230 с.

2. Концептуальные положения по построению мультисервисных сетей на ВСС России. <http://www.minsvyaz.ru/img/uploaded/2002020610512757.pdf>.

3. Крупнов А. Е. Современные телекоммуникационные технологии и услуги в России на рубеже XXI века // CONNECT. 1997. № 3.

УДК 004.93

## ФОНЕТИКО-МОРФОЛОГИЧЕСКАЯ РАЗМЕТКА РЕЧЕВЫХ КОРПУСОВ ДЛЯ РАСПОЗНАВАНИЯ И СИНТЕЗА РУССКОЙ РЕЧИ

**А. Л. Ронжин,**

канд. техн. наук, старший научный сотрудник

**А. А. Карпов,**

аспирант

Санкт-Петербургский институт информатики и автоматизации РАН

**Б. М. Лобанов,**

доктор техн. наук, заведующий лабораторией

**Л. И. Цирульник,**

научный сотрудник

Объединенный институт проблем информатики НАН Беларуси

**О. Йокиш,**

менеджер проектов

Дрезденский технологический университет

Показаны особенности подготовки и обработки текстовых данных для разметки речевых корпусов, обсуждается процедура анализа и сопоставления речевых сигналов в процессе разметки, описаны процедуры транскрибирования русских текстов, морфемного представления языка и речи. Описываются автоматизированные системы создания фонетико-акустических речевых баз данных для персонализированного синтеза речи по тексту и распознавания русской речи SIRIUS.

*The particularities of preparation and processing of text data for speech corpora labeling are presented. The problem of analysis and comparison of speech signals during the labeling is discussed. The procedures of Russian text transcription and morphemic description of language and speech are presented. SIRIUS, an automated system for phonetic-acoustical voice database creation for personalized speech synthesis and Russian speech recognition, is described.*

### Введение

Речевые технологии в России и за рубежом стремительно развиваются благодаря таким преимуществам перед типовыми средствами общения человека с машиной, как естественность, оперативность, освобождение рук и зрения пользователя, возможность управления в экстремальных условиях эксплуатации. Системы распознавания и синтеза речи уже используются в реальных приложениях в США и Европе, но в России, к сожалению, пока существуют лишь отдельные разработки. Основная проблема связана со спецификой русского языка, в частности, со сложным механизмом словообразования и фонетической интерпретации.

При построении систем распознавания и синтеза речи одной из наиболее важных задач является

сегментация и маркировка баз данных (БД) речевых сигналов на минимальные семантически и фонетически значимые единицы речи. В качестве таких единиц при распознавании слитной речи целесообразно использовать морфемы, а при синтезе речи и распознавании фонем — аллофоны. Полученные сегменты сохраняются в БД и служат для обучения акустических моделей в системе распознавания речи, а также для генерации голоса клонируемого диктора в системе синтеза речи по тексту. При этом посредством ограниченного количества морфем создаются всевозможные словоформы (до нескольких миллионов), а посредством аллофонов генерируется необходимое звуковое разнообразие речевого сигнала. Одним из важнейших требований к сегментации и маркировке БД является точность разметки, от нее зависит каче-

ство синтеза и точность распознавания в конечном итоге. Обычно подготовка речевого корпуса осуществляется вручную экспертом-фонетистом с использованием полуавтоматических средств просмотра осциллограмм и спектрограмм сигнала. При высокой квалификации эксперта «ручной» метод обеспечивает достаточно точную разметку речевого корпуса, но требует много времени и усилий.

В данной работе предложена система автоматизированной обработки речевых корпусов, в основу которой положена идея «анализа через синтез» [1]. Система создана в рамках исследований по клонированию голоса и дикции личности [2, 3], созданию многоязычного и многоголосого синтезатора речи по тексту [4] и дикторнезависимой системы распознавания русской слитной речи [5].

### Выбор лингвистических единиц для задач распознавания и синтеза речи

Языки славянской группы относят к числу синтетических языков, которые характеризуются тенденцией к объединению в рамках одной словоформы лексической морфемы (или нескольких лексических морфем) и одной или нескольких грамматических морфем. Более сложная структура словообразования ведет к росту размера распознаваемого словаря, что значительно уменьшает точность и скорость распознавания. В коммерческих системах распознавания речи для английского языка (от фирм Microsoft, Dragon Systems) используется словарь свыше 100 тыс. слов. Для русского языка за счет наличия приставок, суффиксов и окончаний этот словарь возрастает более чем на порядок. При сравнении языковых моделей часто используют две характеристики: коэффициент сложности (perplexity) и число пропущенных слов или число слов, не вошедших в словарь распознавания (Out Of Vocabulary). Коэффициент сложности модели языка равен усредненному числу слов, которые могут быть связаны с предыдущим словом во фразе. В табл. 1 представлена сравнитель-

ная характеристика русского и английского языков по этим параметрам [6].

Кроме того, большинство словоформ одного и того же слова отличаются только в окончаниях, которые произносятся обычно не так четко, как начала слов. В результате большинство ошибок при автоматическом распознавании речи возникает именно в конце слов, что приводит к неточному пониманию всей фразы из-за несогласованности слов в ней. Поэтому основной проблемой автоматического распознавания речи для русского языка является сложный механизм словообразования, из-за которого резко возрастает размер распознаваемого словаря и падает точность. Для решения этой проблемы был введен дополнительный уровень представления речи — морфемный. На основе правил словообразования русского языка были разработаны методы автоматической обработки текстов. За счет разделения словоформы на морфемы словарь распознаваемых лексических единиц значительно сократился, так как в процессе словообразования часто используются одни и те же морфемы.

Далее рассмотрим фонетический уровень представления речи. В русском языке насчитывается 42 фонемы, из них 6 гласных и 36 согласных. В потоке речи фонемы в зависимости от их окружения могут изменять свои артикуляторно-акустические характеристики, что приводит к появлению их модификаций, называемых *аллофонами*, или оттенками фонем. Позиционные аллофоны определяются позицией данной фонемы относительно полноударного гласного. Появление комбинаторных аллофонов фонемы  $\Phi_i$  связано с ее ближайшим окружением, т. е. предшествующей в потоке речи фонемой (левым контекстом)  $\Phi_{i-1}$ , а также последующей в потоке речи фонемой (правым контекстом)  $\Phi_{i+1}$ . Аллофоны фонем обозначаются тремя индексами, указывающими группу позиционных аллофонов (первый индекс), группу левого (второй индекс) и правого (третий индекс) контекстов.

При обработке речи выделяются следующие позиционные аллофоны гласных: полноударный (0), частично ударный (1), первый предударный (2), не первый предударный (3), заударный (4). Здесь в скобках указан первый индекс аллофона. С учетом левого контекста генерируются следующие комбинаторные аллофоны гласных: после синтагматической паузы (0), после переднеязычных (1), губных (2) и заднеязычных (3) твердых, после /Л/ (4), после /Р/ (5), после большинства мягких (6), после /R'/ (7), после /M'/ (8), после /H'/ (9), после гласных (У), (О), (А), (Э), (Ы), (И). Всего — 16 левых контекстов. Здесь в скобках указан второй индекс аллофона. С учетом правого контекста генерируются следующие комбинаторные аллофоны гласных: перед синтагматической паузой (0); перед переднеязычными и заднеязычными (1); перед губными твердыми (2); мягкими (4).

■ Таблица 1. Сравнение статистических моделей английского и русского языков

Размер словаря, тыс. слов	Английский язык		Русский язык	
	Коэффициент сложности	Пропущенные слова, %	Коэффициент сложности	Пропущенные слова, %
65	216,1	1,10	413,3	7,60
100	224,5	0,65	481,0	5,31
200	232,4	0,31	586,8	2,65
400	236,8	0,17	670,9	1,19
500	—	—	689,9	0,93
800	—	—	713,8	0,64
1000	—	—	718,8	0,53



Здесь в скобках указан третий индекс аллофона. Итого, при этих условиях обеспечивается генерация  $N_v = N_p \cdot N_1 \cdot N_r \cdot N_{vph} = 5 \cdot 16 \cdot 5 \cdot 6 = 2400$  гласных аллофонов, где  $N_p$  — количество групп позиционных аллофонов;  $N_1$  — количество групп левых контекстов;  $N_r$  — количество групп правых контекстов;  $N_{vph}$  — количество гласных фонем. С учетом известных закономерностей число аллофонов гласных можно сократить до 1700 без заметного ухудшения качества синтеза речи.

В слове «*G'EN'ER'IRUJ'ESA*» («ГЕНЕРИРУЕТСЯ»), например, индексы ударного *I* определяются следующим образом: первый индекс (позиционный) равен 0, так как гласный полноударный, второй индекс (группа левого контекста) равен 7, так как предшествующая фонема — *R'*, третий индекс (группа правого контекста) равен 1, так как последующая фонема — *R* — принадлежит группе переднеязычных и заднеязычных твердых. Аллофоны всех гласных в слове «*G'EN'ER'IRUJ'ESA*» представлены следующим образом: «*G'E<sub>364</sub>N'E<sub>294</sub>R'I<sub>071</sub>RU<sub>414</sub>J'E<sub>461</sub>CA<sub>410</sub>*».

Аллофоны согласных генерируются только с учетом левого (первый индекс) и правого (второй индекс) контекстов. Левый контекст: после паузы (0), после глухих (1) и звонких (2) согласных, после гласных (3). Здесь в скобках указан первый индекс аллофона. Правый контекст: перед паузой (0), перед глухими (1) и звонкими (2) согласными, перед безударными (3) и ударными (4) гласными. Здесь в скобках указан второй индекс аллофона. Итого, при этих условиях обеспечивается генерация  $N_c = N_1 \cdot N_r \cdot N_{cph} = 4 \cdot 5 \cdot 36 = 720$  согласных аллофонов, где  $N_{cph}$  — количество согласных фонем. На практике при синтезе речи используется порядка 500 аллофонов согласных.

Например, в слове «*G'EN'ER'IRUJ'ESA*» индексы фонемы *G'* определяются следующим образом: первый индекс (группа левого контекста) равен 0, так как фонеме *G'* предшествует пауза, второй индекс (группа правого контекста) равен 3, так как последующая фонема — *E* — безударная гласная. Аллофоны всех согласных в слове «*G'EN'ER'IRUJ'ESA*» представлены следующим образом: «*G'<sub>03</sub>EN'<sub>33</sub>ER'<sub>34</sub>IR<sub>33</sub>UJ'<sub>33</sub>ES<sub>33A</sub>*».

### Общая структура системы фонетико-морфологической разметки речевых корпусов

Основная идея автоматизации процессов сегментации и аллофонно-морфемной маркировки заключается в реализации алгоритмов переноса меток начала и конца аллофонов с синтезированного сигнала на естественный речевой сигнал, взятый из речевых корпусов для распознавания и синтеза русской речи. Алгоритм переноса меток с одного сигнала на другой реализуется путем непрерывного во времени сопоставления естественного и синтезированного речевых сигналов с использованием алгоритмов динамического программирования (НДП-метод) [1]. Для автоматического пе-

реноса меток выбирается один из синтезированных голосов, наиболее близкий к размечаемому естественному голосу.

Общая структурная схема автоматизированной системы сегментации и аллофонно-морфемной маркировки представлена на рис. 1. На вход системы подается речевой корпус — запись естественного речевого сигнала — и орфографический текст стенограммы записи. Текст автоматически размечается на морфемы, транскрибируется и поступает на вход синтезатора речи, который, используя уже существующую БД элементов речи некоторого диктора, генерирует речевой сигнал, размеченный на фонетико-морфологические сегменты. Синтезированный и естественный речевые сигналы поступают на вход модуля переноса меток, где производится автоматическая разметка входного сигнала на фонемы и морфемы.

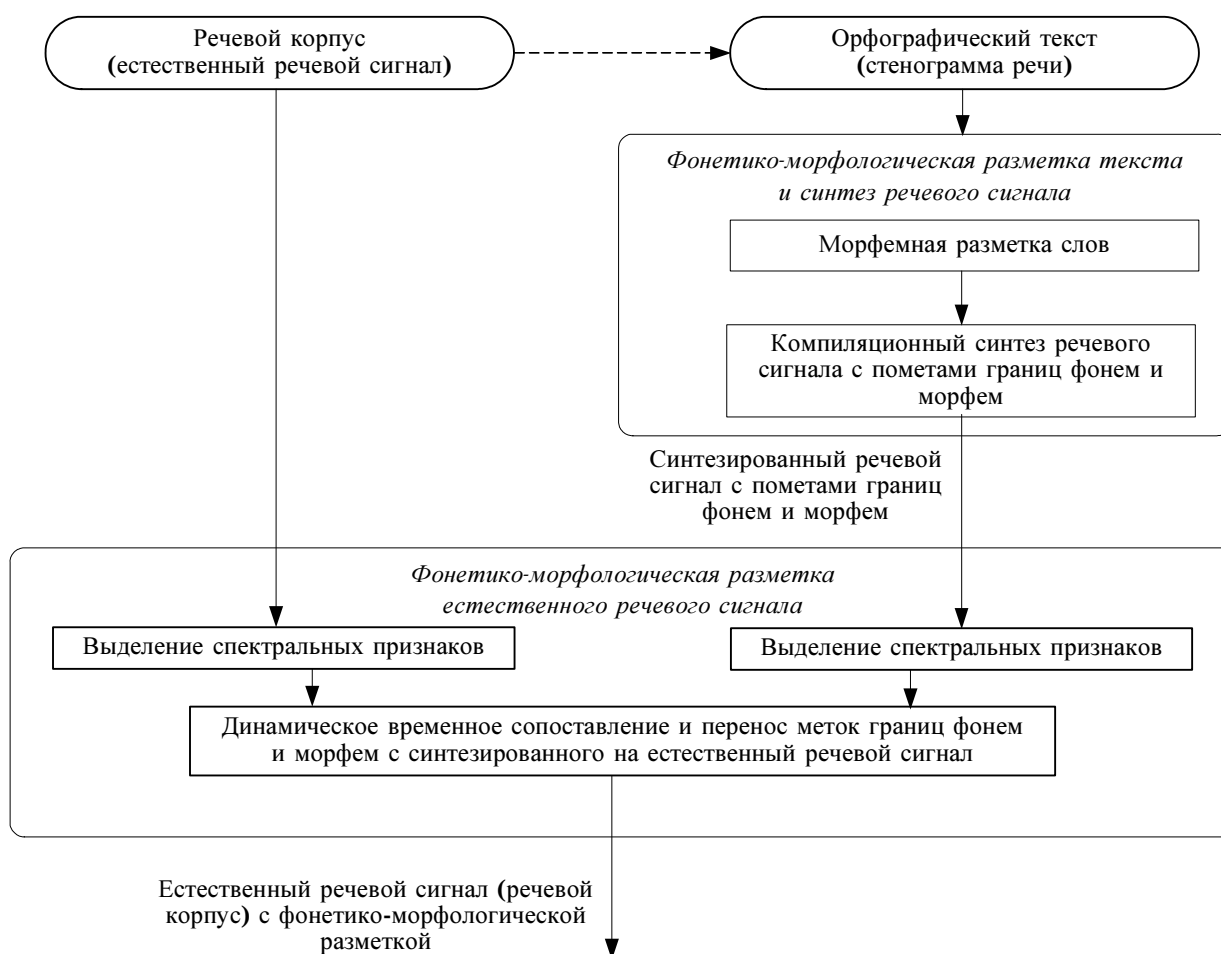
При фонетико-морфологической разметке производится автоматическая обработка текста на основе программных модулей транскрибирования и морфологического разбора. Для морфемной сегментации используются БД различных типов морфем, собранные из печатных словарей. Большая часть корневых морфем (около 4000) была взята из словаря морфем русского языка Кузнецовой [7], а различные фиксальные морфемы и флексии взяты из работы [8]. Кроме того, впоследствии, при создании ряда приложений, словарь морфем постоянно пополнялся и сейчас насчитывает около 5000 морфем.

Разбиение слова на морфемы осуществляется путем подбора различных типов морфем с учетом правил их следования в одном слове (табл. 2). Возможные пары типов морфем отмечены знаком «+». При получении недопустимой пары (отмечена в таблице знаком «-») данная гипотеза разбиения слова на морфемы откидывается и поиск продолжается дальше, пока не обнаруживается конец слова «STOP».

Разработанные БД морфем использовались для создания морфемной модели языка, строящейся на основе статистики встречаемости различных пар морфем. Для первичной оценки модели языка были использованы доступные в Интернете текстовые корпуса. Текст общим объемом около 50 Мб был предварительно обработан, а все слова в нем размечены на морфемы. В результате анализа текста были получены словарь размером около 5000 морфем и вероятности встречаемости всех пар морфем.

В результате описанного процесса автоматической обработки речевых данных получаем аннотированные корпуса речи и текста. На примере компиляционного синтезатора (рис. 2) рассмотрим их использование.

Синтезатор состоит из четырех процессоров: лингвистического, просодического, фонетического и акустического. Каждый из процессоров использует для осуществляемых им преобразований специализированные БД. В них заложены как об-



■ Рис. 1. Общая схема системы фонетико-морфологической разметки речевых корпусов

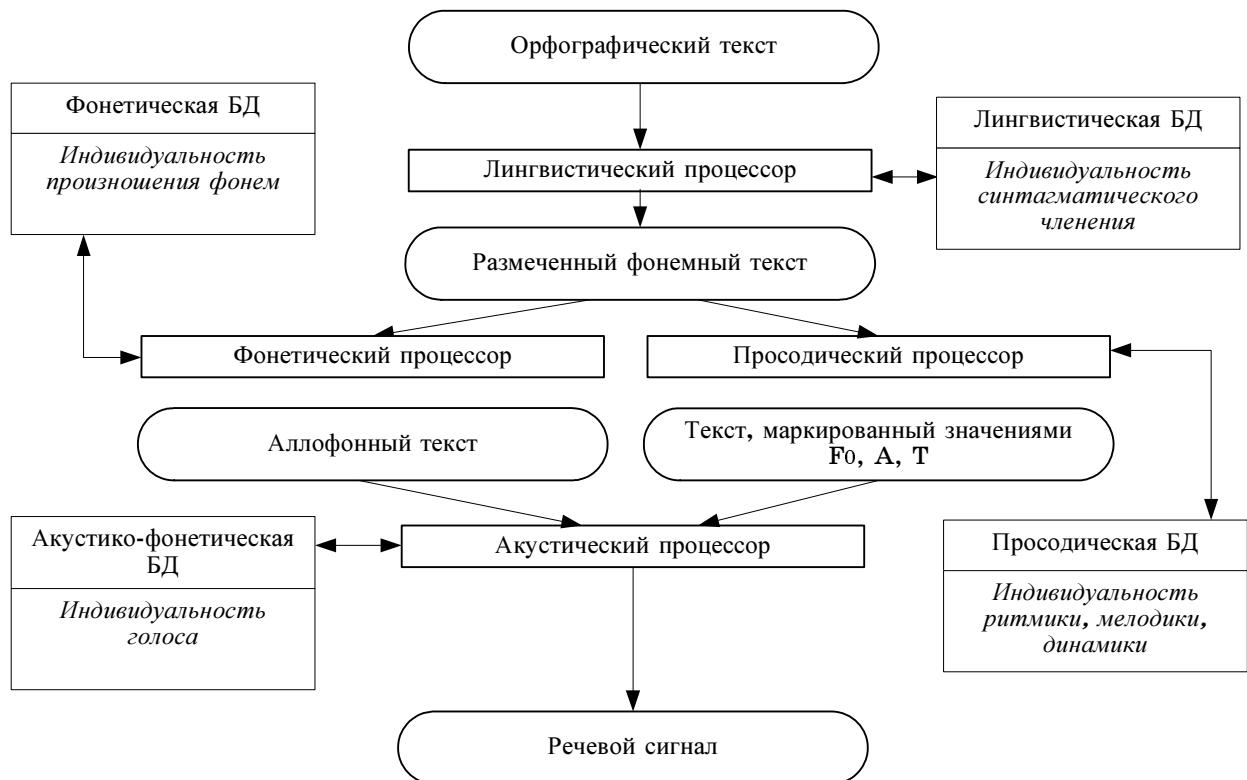
щие языковые правила (лингвистические, просодические, фонетические, акустические), так и связанные с индивидуальными особенностями голоса и речи диктора.

Лингвистический процессор предназначен для преобразования входного орфографического текста в размеченный фонемный текст. Под размет-

кой понимается разбиение текста на отдельные элементы в следующей иерархии: фонетический период, фраза, синтагма. Кроме того, процессор осуществляет расстановку словесных ударений и интонационную маркировку синтагм. Для расстановки ударений использовалась БД словоформ русского языка с информацией об ударениях в сло-

■ Таблица 2. Таблица согласования различных типов морфем в слове

Тип текущей морфемы	Тип следующей морфемы				
	Префикс	Корень	Интерфикс	Суффикс	Окончание
Нет	+	+	-	-	-
Префикс	+	+	-	-	-
Корень	STOP	STOP	+	+	+
Интерфикс	+	+	-	-	-
Суффикс	-	-	-	+	+
Окончание	STOP	STOP	STOP	STOP	-



■ Рис. 2. Общая структурная схема компиляционного синтезатора речи по тексту

воформах, насчитывающая свыше 1 млн 700 тыс. словоформ русского языка [9].

Размеченный фонемный текст поступает на вход двух процессоров — просодического и фонетического. В результате работы просодического процессора фонемный текст делится на акцентные единицы (АЕ). Далее осуществляется разметка АЕ на элементы акцентных единиц (ЭАЕ): интонационное предъядро, ядро и заядро. И, наконец, последняя функция просодического процессора — установка значений амплитуды, длительности фонем и частоты основного тона для каждого ЭАЕ. Задача фонетического процессора заключается в генерации позиционных и комбинаторных аллофонов по входному фонемному тексту. Акустический процессор на основе информации о том, какие аллофоны требуется синтезировать и какими просодическими характеристиками должен обладать каждый аллофон, генерирует синтетический речевой сигнал. При этом используется БД, в которой хранятся акустические волны аллофонов синтезируемого голоса. Полученный синтезированный сигнал используется для фонетико-морфологической разметки естественного речевого сигнала.

Модуль разметки включает в себя анализатор спектральных параметров естественного и синтезированного сигналов. Затем на основе выделенных параметров выполняется динамическое временное сопоставление естественного и синтезированного речевых сигналов. По найденному соот-

ветствию производится перенос меток с синтезированного на естественный речевой сигнал.

Анализ параметров речевого сигнала в описываемой системе заключается в вычислении нормированной сонограммы. Вычисление сонограммы проводится путем пропускания исходной записи через набор полосовых фильтров Чебышева и вычисления среднего значения сигналов на каждом участке длительностью 10 мс. Полосы пропускания фильтров выбраны в соответствии со шкалой Барка. Нормирование сонограммы осуществляется по формуле

$$S_n(n, j) = \sum_{k=n-T}^{n+T} \sum_{l=0}^C \Delta(S(n, j), S(k, l)),$$

где  $S_n(n, j)$ ,  $S(n, j)$  — нормированное и ненормированное значения точки  $n, j$  сонограммы соответственно;  $T$  — интервал нормирования;  $C$  — число каналов в сонограмме.

Функция

$$\Delta(S(n, j), S(k, l)) = \begin{cases} 1, & \text{если } S(n, j) - S(k, l) > \varepsilon \\ 0, & \text{если } -\varepsilon \leq S(n, j) - S(k, l) \leq \varepsilon, \\ -1, & \text{если } S(n, j) - S(k, l) < -\varepsilon \end{cases}$$

где  $\varepsilon$  — порог, определяемый уровнем шума во входном сигнале.

После вычисления сонограмм синтезированной и естественного речевых сигналов их сравнива-



ют НДП-методом, в основу которого положен итерационный алгоритм вычисления интегральных расстояний:

$$D(n, m) = \min \left[ \begin{array}{l} D(n-1, m) + k_h d\{S(n); E(m)\} + \\ D(n, m-1) + k_v d\{S(n); E(m)\} + \\ D(n-1, m-1) + k_d d\{S(n); E(m)\} + \\ + \frac{k}{M} |m - T(n-1, m)|; \\ + \frac{k}{M} |m-1 - T(n, m-1)|; \\ + \frac{k}{M} |m-1 - T(n-1, m-1)|. \end{array} \right]$$

При использовании классических формул НДП результаты сравнения сигналов были не всегда точными, поэтому в описываемой системе в эти формулы включены дополнительные коэффициенты: коэффициент времени  $k$  и множители  $k_h, k_v, k_d$ , причем коэффициент  $k$  нормируется по отношению к длине синтезированного сигнала  $M$ . Оптимальные значения коэффициентов  $k_h, k_v, k_d, k$  были найдены экспериментальным путем.

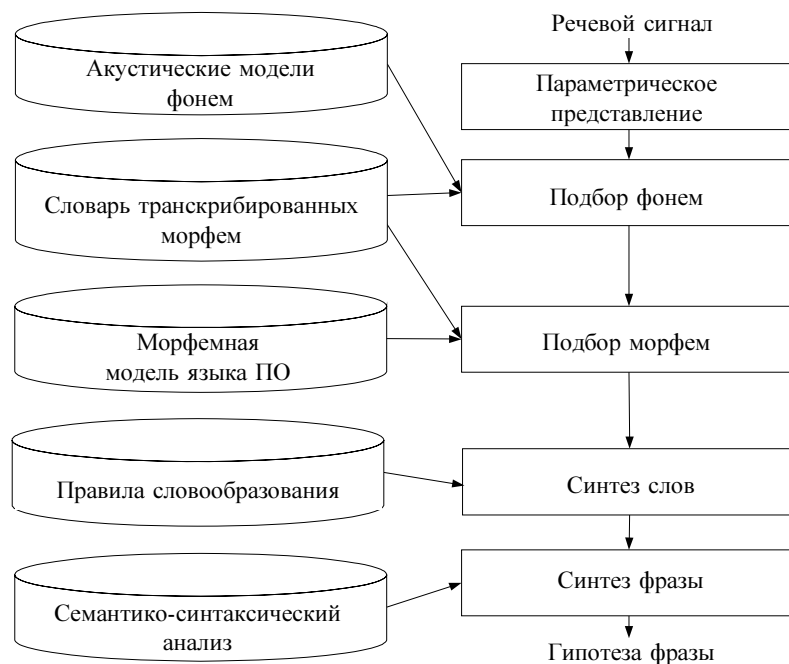
### Применение фонетико-морфологической разметки для распознавания речи

Тестирование системы распознавания и полученных БД производится в задачах голосового доступа к рубрикам электронного каталога «Жел-

тые страницы Санкт-Петербурга» (размер словаря составил 1850 слов), а также заказа авиабилетов из аэропортов Санкт-Петербурга. Далее приводятся результаты распознавания именно по первой задаче — распознавания названий рубрик. В общепринятую архитектуру распознавания речи был введен дополнительный уровень представления языка и речи — морфемный и разработана оригинальная система распознавания русской речи SIRIUS (SPIIRAS Interface for Recognition and Integral Understanding of Speech) [5] (рис. 3).

Речевой сигнал, поступающий с микрофона, в первую очередь проходит этап параметрического представления, где отрезаются начальные и конечные паузы в сигнале, а оставшийся участок кодируется в последовательность векторов признаков, которая уже следует в модуль распознавания фонем. При распознавании фонем (которые используются в форме трифонов) и формировании морфем используются методы скрытого марковского моделирования и смесей гауссовских распределений. В отличие от существующих аналогов, в нашей модели вместо слов используются морфемы. За счет этого на этапе распознавания лексических единиц было получено существенное увеличение скорости.

После распознавания фонем и подбора наиболее вероятных цепочек морфем получившийся набор гипотез далее используется для формирования цепочек слов. Синтез слов из различных типов морфем осуществляется по схеме, представленной на рис. 4. В данной модели заданы начальное и конечное состояния, а в остальных узлах присутствуют различные типы морфем. Дугами обозначены возможные переходы. Планируется сделать эту



■ Рис. 3. Структура системы распознавания русской речи SIRIUS

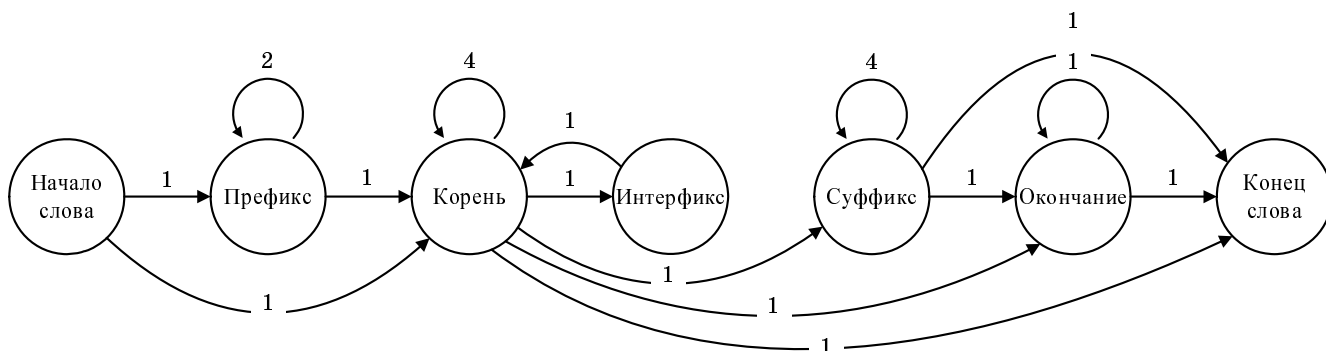


Рис. 4. Синтез слова из морфем

Таблица 3. Сравнение моделей распознавания по точности

Диктор	Точность целословного распознавания		Точность морфемно-ориентированного распознавания	
	слов	фраз	морфем	фраз
1	96	94	82	92
2	93	90	78	90
3	95	93	81	92
4	91	91	79	91
5	95	93	82	91
Среднее	94,0	92,2	80,4	91,2

модель вероятностной, а пока максимальное количество переходов из состояния в состояние заданы жестко. На этом этапе обработки на основе каждой поступившей гипотезы фразы, представленной в виде последовательности морфем, формируется еще несколько гипотез, представленных последовательностью гипотез слов.

Последним этапом обработки является синтез фразы. На входе этого уровня мы получаем цепочку слов, составляющую произнесенное высказывание. Однако процент ошибок распознавания слов здесь достаточно высок, и большинство из них, как уже отмечалось, происходит из-за ошибок в распознавании окончаний, которые производятся не так четко, как начала слов. А ошибки в окончаниях при распознавании слов приводят к тому, что происходит ошибка в распознавании всей фразы из-за несогласованности слов в предложении. Для того чтобы исправить эти ошибки, мы используем морфологический анализ предложения и грамматические правила русского языка. Таким образом, на выходе системы распознавания речи мы получаем цепочку слов, составляющих предложение.

Для тестирования системы использовались 635 фраз, записанных в офисных условиях. В экспе-

Таблица 4. Сравнение моделей распознавания по скорости обработки

Расознавание	Время, затрачиваемое на тестовый набор, с	Среднее время, с		
		на одну фразу	на одно слово	на одну морфему
Целословное	2993	4,71	1,16	–
Морфемно-ориентированное	1740	2,74	0,67	0,47

рименте участвовало 5 дикторов. Записанные файлы были пропущены через модель целословного распознавания, а затем через морфемно-ориентированную модель распознавания с последующим словообразованием (табл. 3). По сравнению с первой моделью точность распознавания морфем несколько снизилась, но за счет последующих уровней обработки точность распознавания фраз практически не изменилась.

Также был проведен тест по сравнению скорости работы морфемного и целословного распознавателей (табл. 4). Общее количество тестовых фраз, содержащихся в тестовой БД, составило 635 (состоящих из 2574 слов). Из таблицы видно, что при использовании разработанной системы скорость возросла более чем в 1,7 раза, что при незначительном падении точности позволяет говорить о создании перспективной системы распознавания речи для больших словарей.

Таким образом, разработанный модуль распознавания слитной русской речи показал достаточно высокую точность и дикторонезависимость к носителям русского языка.

### Применение фонетико-аллофонной разметки для персонализированного синтеза речи по тексту

Для автоматизированного создания БД фонетико-акустических характеристик голоса и дикции личности создана система AcousticClonator.

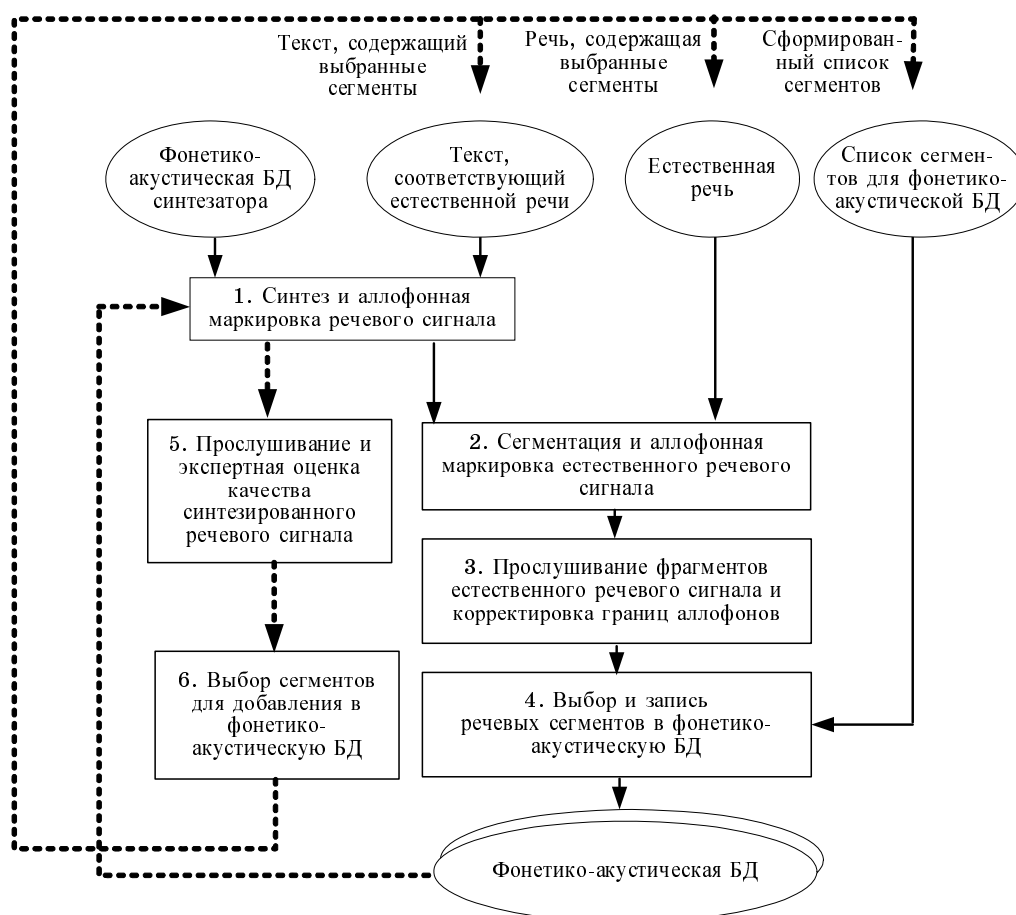
В системе реализованы этапы создания индивидуализированных БД, ответственных за синтез акустических свойств голоса и фонетических особенностей произношения.

Технология создания фонетико-акустической БД с использованием этой системы, взаимодействие блоков, входные и выходные данные показаны на рис. 5. Создание фонетико-акустической БД состоит из двух или более этапов. Последовательность действий, выполняемых на первом этапе, обозначена сплошными линиями, а на втором и последующих этапах — штриховыми.

На первом этапе в качестве списка сегментов для фонетико-акустической БД используется минимально необходимый или расширенный набор аллофонов, а в качестве речевой базы — звуковой массив, включающий набор русских слов, содержащий все необходимые аллофоны. Текст, соответствующий звуковому массиву, а также существующая фонетико-акустическая БД являются входными данными для синтеза и аллофонной маркировки речевого сигнала 1. Аллофонно-размеченный синтезированный речевой сигнал используется для сегментации и аллофонной маркировки естественного речевого сигнала 2. Следующая функция 3 — прослушивание и, при необходимости, ручная корректировка границ аллофонов — реализуется экспертом-фонетистом, который может посмотреть осциллограмму речевого сигнала, прослушать любой его участок, передвинуть метки границ аллофонов. И, наконец, пользователь может прослушать все слова, содержащие указанный речевой сегмент, и установить, из какого именно слова указанный сегмент будет помещен в БД 4. Результатом первого этапа является фонетико-акустическая БД нового диктора, содержащая минимально необходимый (или расширенный) набор звуковых волн аллофонов.

На втором этапе в синтезаторе речи используется вновь созданная фонетико-акустическая БД. Эксперт-фонетист прослушивает синтезированный по различным текстам речевой сигнал, оценивает его качество, выделяет речевые участки, качество звучания которых неудовлетворительно, и формирует список сегментов для пополнения фонетико-акустической БД. В качестве таких сегментов могут выступать аллофоны, мультифоны (сочетания аллофонов), слоги и другие сегменты речи. Сформированный список сегментов для пополнения фонетико-акустической БД, дополни-

тая функция 3 — прослушивание и, при необходимости, ручная корректировка границ аллофонов — реализуется экспертом-фонетистом, который может посмотреть осциллограмму речевого сигнала, прослушать любой его участок, передвинуть метки границ аллофонов. И, наконец, пользователь может прослушать все слова, содержащие указанный речевой сегмент, и установить, из какого именно слова указанный сегмент будет помещен в БД 4. Результатом первого этапа является фонетико-акустическая БД нового диктора, содержащая минимально необходимый (или расширенный) набор звуковых волн аллофонов.



■ Рис. 5. Технология создания фонетико-акустической БД

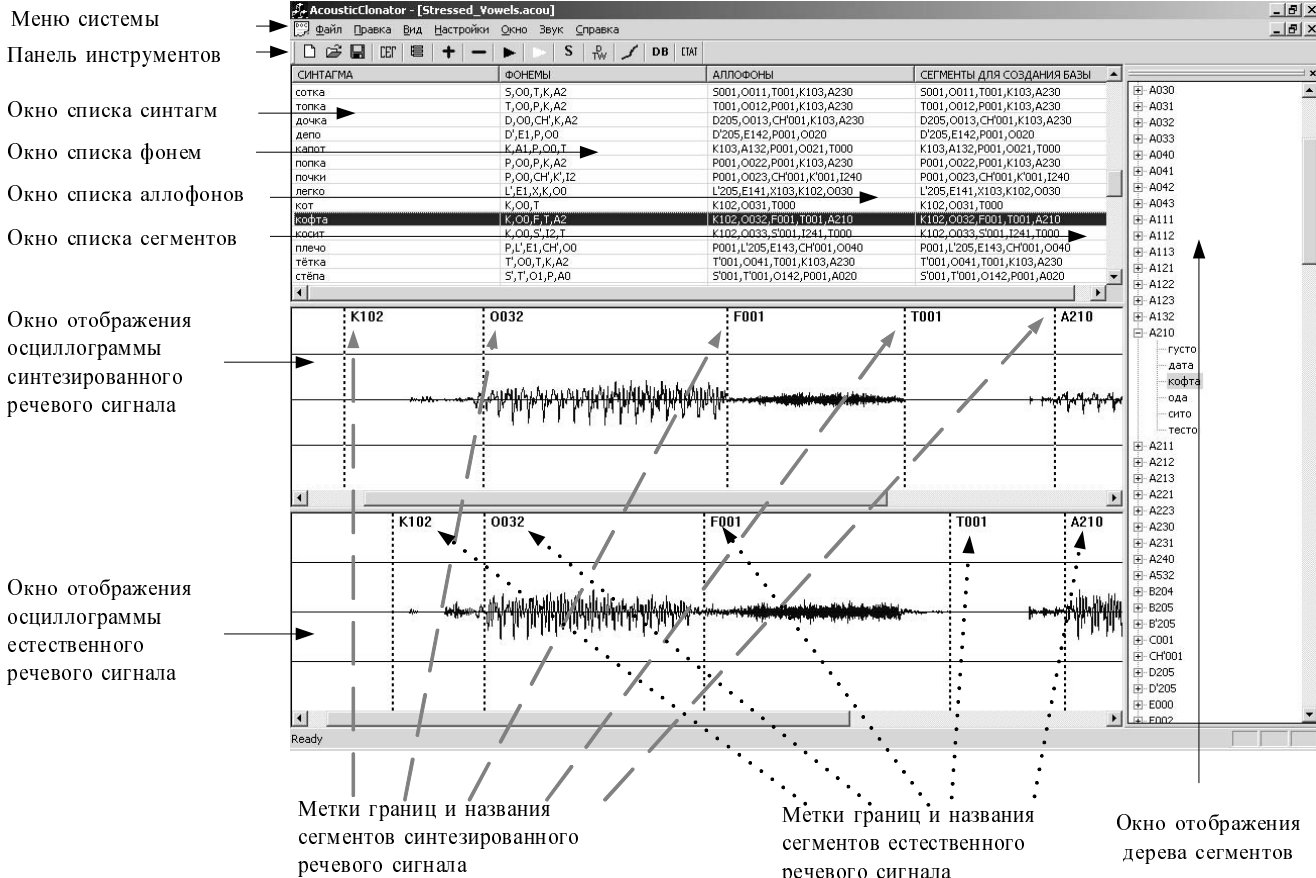


Рис. 6. Общий вид и основные блоки пользовательского интерфейса системы

тельная речевая база, содержащая эти сегменты, и соответствующий текст снова подаются на вход системы. В качестве дополнительных речевых баз используются записи двух специально подобранных фонетически репрезентативных текстов.

Второй этап создания фонетико-акустической БД может повторяться несколько раз, при этом после очередного прослушивания синтезированного речевого сигнала и оценки его качества эксперт может формировать все новые списки сегментов

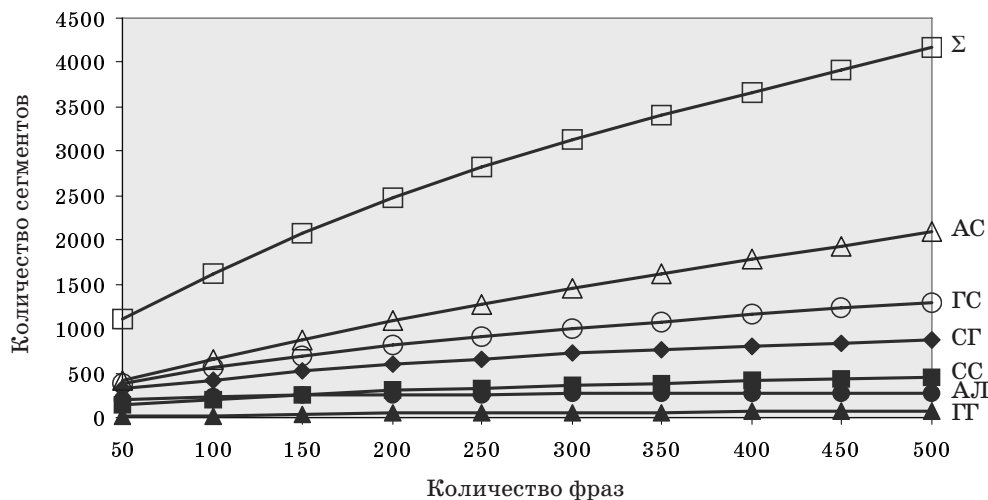


Рис. 7. Графики зависимости количества аллофонов (АЛ), диаллофонов (ГГ, СГ, СС, ГС), аллослогов (АС) и общего количества сегментов (Σ) от объема выборки



для пополнения БД, с каждым шагом повышая качество синтезируемой речи.

Пользовательский интерфейс системы (рис. 6) реализован в среде MS Visual C++ 6.0. Приложение имеет мультимедийный тип, что дает возможность пользователю варьировать наборы входных данных и делает систему более гибкой и легко настраиваемой. Основные блоки пользовательского интерфейса состоят из меню, панели инструментов, блока окон, в котором отображается список синтагм (орфографический текст), осциллограммы естественного и синтезированного речевых сигналов, синтагматического дерева сегментов.

Описанная система сегментации речевого сигнала используется для создания БД элементов речи, которая необходима для распознавания, компьютерного клонирования голоса и речи нового диктора, а также для наращивания уже существующих БД дикторов путем добавления в базу новых элементов синтеза — диаллофонов и аллослогов, что позволяет значительно улучшить качество синтезируемой речи. С использованием разработанной системы была создана акустическая БД речевых сегментов различной размерности — аллофонов, диаллофонов и аллослогов. В качестве текстовой БД использованы таблицы ГОСТ для измерения фразовой разборчивости речи [10], разработанные с учетом фонетической репрезентативности русской речи. Таблицы включают 500 фонетически сбалансированных фраз. На рис. 7 представлены графики зависимости количества различных аллофонов; диаллофонов типа ГГ (гласный-гласный), СГ (согласный-гласный), СС (согласный-согласный) и ГС (гласный-согласный), а также аллослогов, которое можно получить с использованием описанной системы в зависимости от объема выборки фраз из текстовой БД ГОСТ.

## Заключение

Исследование проблем автоматической обработки речи является важным фундаментальным направлением. Эти проблемы сдерживают развитие всевозможных систем взаимодействия человека с машиной. Представленные модели синтеза и распознавания речи в первую очередь направлены на учет особенностей русского языка. Авторами разработан и опробован новый морфемный метод представления языка и речи. Он показал высокое качество и устойчивость работы на словаре до 2000 слов конкретной предметной области. Разработаны БД различных типов морфем. В результате такой обработки обеспечивается инвариантность к грамматическим отклонениям, увеличивается скорость распознавания русской речи и других языков со сложным механизмом словообразования (в частности, славянских). Последующие работы направлены на увеличение размера распознаваемого словаря, настройку системы распознавания русской речи SIRIUS к работе с телефонным кана-

лом. При внедрении системы распознавания речи в телекоммуникационные приложения будут учтены проблемы, связанные со спецификой телефонных линий и различиями характеристик телефонных аппаратов. Накопленные речевые и лексические БД будут использованы для дальнейшего изучения механизма понимания речи и создания эффективных средств человеко-машинного взаимодействия.

Разработанные системы сегментации речевого сигнала и текста позволяют автоматизировать трудоемкий процесс создания акустических БД, а также учитывать индивидуальные фонетико-акустические особенности голоса личности. Как показал опыт, создание минимально необходимого набора аллофонов опытным фонетистом вручную занимает несколько недель рабочего времени, а расширенного набора — более двух месяцев. При использовании предложенной системы создание БД, содержащей полный набор звуковых волн аллофонов, занимает не более одного часа машинного времени. Автоматизация процесса пополнения БД мультифонами (диаллофонами и аллослогами) позволила значительно повысить качество синтезированной речи, практически приблизив к естественной. Сказанное подтверждается результатами испытания новой системы синтеза речи по тесту «МУЛЬТИФОН», акустическая БД которой содержит свыше 4 тыс. фонетических сегментов речи, полученных с помощью описанной системы фонетико-аллофонной разметки речевого сигнала. Кроме очевидного практического применения — создания индивидуализированных речевых БД для последующего высококачественного синтеза речи по тексту с манерой чтения конкретного человека и его голосом, система клонирования фонетико-акустических характеристик речи может использоваться в криминалистике для экспресс-идентификации голоса подозреваемого [11].

Данные исследования проводятся при финансовой поддержке правительства Санкт-Петербурга, Европейского Сообщества SIMILAR NoE FP6: IST-2002-507609, а также проекта INTAS № 04-77-7404.

## Литература

1. Давыдов А., Киселев В., Лобанов Б., Цирульник Л. Система сегментации речевого сигнала методом анализа через синтез // Изв. Белорусской инженерной академии. 2004. № 1 (17)/1. С. 112–114.
2. Лобанов Б. М. Компьютерное «клонирование» персонального голоса и речи // Новости искусственно-интеллекта. 2002. № 5 (55). С. 35–39.
3. Lobanov B. M., Tsiurulnik L. I. Phonetic-Acoustical Problems of Personal Voice Cloning by TTS: Proc. of the International Conference SPECOM'2004. St. Petersburg, 2004. P. 17–21.

- |  |  |
|--|--|
| <p>4. Lobanov B. M., Tsirulnik L. I. Development of multi-voice and multi-language TTS synthesizer (languages: Belarussian, Polish, Russian): Proc. of the International Conference SPECOM'2006. St. Petersburg, 2006. P. 274–283.</p> <p>5. Карпов А. А., Ронжин А. Л., Ли И. В. SIRIUS — система дикторнезависимого распознавания слитной русской речи // Изв. ТРТУ. 2005. № 10. С. 44–53.</p> <p>6. Whittaker E. W. D. Statistical Language Modelling for Automatic Speech Recognition of Russian and English. PhD thesis / Cambridge University. Cambridge, 2000. P. 141.</p> <p>7. Кузнецова А. И., Ефремова Т. Ф. Словарь морфем русского языка: Ок. 52000 слов. М.: Рус. яз., 1986. 1136 с.</p> | <p>8. Русская грамматика: В 2 т. / Редкол.: Н. Ю. Шведова (гл. ред.) и др. Т. 1: Фонетика. Фонология. Ударение. Интонации. Словообразование. Морфология / Н. С. Авилова, А. В. Бондарко, Е. А. Брызгунова и др. М.: Наука, 1980. 783 с.</p> <p>9. <a href="http://starling.rinet.ru">http://starling.rinet.ru</a></p> <p>10. ГОСТ 16600–72. Передача речи по трактам радиотелефонной связи. М.: Изд-во стандартов, 1973.</p> <p>11. Давыдов А. Г., Киселев В. В., Лобанов Б. М., Цирульник Л. И. Система экспресс-идентификации голоса личности методом клонирования акустических характеристик речи // Теория и практика речевой коммуникации: Тез. докл. Междунар. конф. М., 2004. С. 23–28.</p> |
|--|--|

**«IV МЕЖВУЗОВСКАЯ КОНФЕРЕНЦИЯ МОЛОДЫХ УЧЕНЫХ»  
апрель 2007 г.**

**Место проведения конференции:** Санкт-Петербургский государственный университет информационных технологий, механики и оптики.

**Адрес:** 197101, Санкт-Петербург, Кронверкский пр., д. 49

**Цель конференции**

Конференция проводится с целью стимулирования научно-технической деятельности молодых ученых, приобретения ими опыта публичных выступлений, подачи научных документов для публикации, а также с целью ознакомления научной общественности с результатами исследований приоритетных направлений развития науки, технологий и техники.

К участию приглашаются молодые ученые (до 35 лет).

**Программа конференции**

Планируется проведение пленарного заседания и секций с устными докладами.

В рамках конференции будут проведены научные школы: I сессия научной школы «Информационные технологии в образовании», II сессия научной школы «Информационная безопасность, проектирование, технология элементов и узлов компьютерных систем».

**Направления работы конференции**

- Информационные технологии
- Информационно-телекоммуникационные системы
- Безопасность и противодействие терроризму, защита информации
- Технологии производства программного обеспечения

Технологии приборостроения, мехатроника и робототехника

Системный анализ, математическое моделирование и управление в технических системах

Теплофизика и теоретическая теплотехника

Живые системы, биомедицинские технологии и томография

Оптотехника и оптические материалы

Фотоника и оптоинформатика

Физика твердого тела, наносистем и материалов

Гуманитарные науки (философия, социология, политология, педагогика)

Экономика, финансы и менеджмент организации

**Контрольные сроки**

Тезисы докладов и тексты статей для сборника принимаются до **2 февраля 2007 г.**

**Издание трудов конференции**

По результатам работы конференции планируется выпуск сборника тезисов докладов и сборника лучших докладов.

**Дополнительная информация**

Оргкомитет конференции,  
тел.: (812) 232-04-64,  
эл. почта: [kmu@mail.ifmo.ru](mailto:kmu@mail.ifmo.ru)  
сайт: [http://www.ifmo.ru/index.php?out=itmo\\_science\\_conference\\_07](http://www.ifmo.ru/index.php?out=itmo_science_conference_07)

УДК 681.3.06

## АВТОМАТИЧЕСКАЯ ГЕНЕРАЦИЯ АВТОМАТНОГО КОДА

**С. Ю. Канжелев,**

магистрант

**А. А. Шалыто,**

доктор техн. наук, профессор

Санкт-Петербургский государственный университет информационных технологий,  
механики и оптики

*Рассматривается подход к автоматической генерации кода автоматных программ на любом априори заданном языке программирования по графам переходов автоматов.*

*The approach to the automatic automata's program code generation is considered in the paper. The code is generated in any programming language by the automaton's transition graph.*

*Если ты ленив и упорен,  
то ты непременно чего-нибудь добьешься.*  
**Жорж Фейдо**

### Введение

В последнее время весьма актуален вопрос о визуальном конструировании программ [1], в частности, проблема автоматической генерации кода по графическим моделям. Модели можно разделить на два класса — статические и динамические. Среди статических моделей объектно-ориентированных программ основной является диаграмма классов. Многие инструментальные средства генерируют «скелет» кода по диаграммам классов.

Динамические свойства рассматриваемого класса программ наиболее эффективно описывают диаграммы состояний (графы переходов). Известны различные инструментальные средства, генерирующие код по графам переходов [2–5]. Указанные средства являются специализированными — строят код для одного-двух языков программирования.

Цель настоящей работы состоит в описании подхода к генерации кода автоматных программ (программ с явно выделенными состояниями) на любом априори заданном языке программирования.

На основе этого подхода было создано инструментальное средство MetaAuto [6], которое позволяет преобразовывать изображения графов переходов, представленных с помощью редактора MS Visio, в исходные коды программ на любых языках программирования, для которых предварительно созданы соответствующие шаблоны. В ходе работы над этим инструментальным сред-

ством в качестве примера были созданы шаблоны для трех языков программирования (C++, C# и Turbo Assembler).

### Этапы генерации кода

**Декомпозиция задачи.** Как отмечалось выше, описываемый подход предполагает построение исходного кода на различных языках программирования. Для реализации этого требования разумно разделить этап преобразования графической модели в исходный код на два: преобразование графической модели в некий общий формат представления графа переходов и преобразование из этого формата в различные языки программирования. Такое разделение приводит к большей платформенной независимости. При этом легко перейти от одного редактора графов переходов на другой, реализовав для нового редактора процедуру преобразования в общий формат. Также общий формат представления графа переходов может быть использован в валидаторах — программах проверки корректности графов переходов.

Для облегчения работы с данными, содержащимися в графе переходов, имеет смысл использовать программное представление общего формата. Под программным представлением понимается библиотека классов, изоморфная по своей структуре данным общего формата, которая предназначена для упрощения процесса чтения и сохране-



■ Рис. 1. Декомпозиция процесса генерации исходного кода

ния данных различными программами: валидаторами, генераторами кода, другими программами пользователя.

Таким образом, этап генерации кода может быть разбит на три: преобразование графа переходов в программное представление, затем изоморфное преобразование программного представления в общий формат и, наконец, преобразование из общего формата в исходный код программы. На рис. 1 изображены этапы генерации исходного кода программы. По этой схеме построено разработанное инструментальное средство MetaAuto [6].

**Генерация кода.** В рамках подхода к генерации исходного кода программы необходимо осуществлять преобразование графа переходов автомата, представленного в некотором общем формате, в исходный код программы. В общем случае граф переходов можно рассматривать как набор метаданных определенной структуры из предметной области решаемой задачи. Вопрос об автоматической генерации кода с заданными свойствами при этом является одним из вопросов, решаемых в рамках порождающего программирования (Generative Programming) [7].

В настоящее время в рамках порождающего программирования разработан ряд способов генерации кода. Среди них можно выделить: подстановки; подстановки с исполнением кода; обработчики данных регулярной структуры.

*Генерация кода, основанная на подстановках*, предполагает, что разработчик создает шаблон кода и набор данных в специальном формате, а затем, с помощью вспомогательной программы, выполняет подстановку этих данных в шаблон. Набор используемых в шаблоне подстановок может определяться как статически, так и динамически. Такой подход нагляден и прост в использовании, однако имеет весьма ограниченную область применения и требует предварительной подготовки передаваемых для подстановки данных. Классический пример подстановок (с некоторыми оговорками) — шаблоны (templates) языка C++.

Гораздо шире возможности при *генерации кода с использованием подстановки с исполнением кода*. Этот вид генерации отличается от предыдущего возможностью применять в шаблоне не только подстановки, но также и вставки исполняемого кода, оперирующего переданными в шаблон данными. Исполняемый код чаще всего использует язык, который специально создан для конкретного типа шаблонов и включает основные алгорит-

мические конструкции, такие как, например, простое ветвление (IF), выполнение итераций по переданным в качестве параметров спискам (WHILE), циклы с заданным количеством итераций (FOR).

Отметим, что в процессе усложнения конструкций, применяемых в шаблоне, и увеличения возможностей языка уменьшается наглядность самих шаблонов.

Пример использования такого способа генерации, где применяется технология ASP, первоначально разработанная для генерации HTML-страниц, приведен в работе [8]. В качестве языка исполняемого кода в указанной технологии используется язык Visual Basic.

Недостатком подходов к генерации кода, основанных на подстановках и подстановках с исполнением кода, является необходимость специальной подготовки данных для передачи в шаблон.

Третий способ генерации кода основан на *обработчиках данных регулярной структуры*; он предполагает полное разделение данных и их представления. В этом случае шаблон играет роль обработчика данных и пишется на специальном метаязыке. Примером может служить XSLT-обработка данных, представленных в XML-формате.

Вопрос генерации кода на основе XSLT-преобразования рассматривается в работах [9–11].

Основным достоинством этого способа генерации кода является возможность обработки данных сложной структуры без предварительной подготовки этих данных.

Существование перечисленных видов генерации кода обусловлено неоднородностью решаемых в рамках порождающего программирования задач и структуры метаданных. Вопрос об использовании того или иного способа генерации зависит от условий конкретной задачи и данных, используемых для такой генерации.

**Особенности генерации исходного кода для графов переходов.** Одним из вопросов, решаемых при преобразовании графов переходов автоматов в исходные коды программ, является вопрос о реализации групповых переходов [12]. Чаще всего автоматная процедура (процедура, реализующая поведение автомата), состоит из одного оператора switch. Групповые переходы в этом случае реализуются с помощью дублирования их для всех состояний, содержащихся в группе, к которой данный групповой переход относится.

Также непростой задачей является генерация логических выражений. Например, код програм-



мы, реализующей вычисление значения логических выражений для языка Turbo Assembler, сильно отличается от кода программы, реализующей вычисление логических выражений для языков высокого уровня, в которых логические выражения часто записываются одной формулой.

Большое количество рекурсивных структур и другие особенности графов переходов определяют выбор способа генерации исходного кода.

Инструментальное средство MetaAuto использует XSLT-технологии для генерации исходного кода. В терминах рис. 1 в качестве общего формата в MetaAuto для представления графов переходов выбран XML-формат. Генератор для преобразования общего формата в исходный код программы базируется на XSLT-шаблонах.

*XSLT-преобразование.* XSLT (*eXtensible Stylesheet Language Transformations* — расширяемый язык стилей для преобразований) в последнее время стал популярной XML-технологией. Спецификация определяет XSLT как язык для преобразований одних XML-документов в другие XML-документы. Однако за время своего существования XSLT стал использоваться значительно шире, например для автоматической генерации кода.

В работах [9–11] рассматривается вопрос применения языка XSLT в качестве инструмента для генерации кода программ. В работе [9] отмечены основные достоинства и недостатки генерации кода с использованием этого языка.

#### Достоинства:

- широкие возможности по изменению шаблонов без изменения любой другой функциональности;
- широкие возможности по манипулированию данными; возможность извлекать данные из дополнительных источников;
- возможность изменять язык программирования с помощью небольшого изменения шаблона;
- наглядность и широкое распространение XML-формата; нет необходимости каждый раз придумывать новый формат хранения данных.

#### Недостатки:

- трудно контролировать отступы и пробелы при генерации текста; небольшое количество функций работы со строками (спецификация XSLT 2.0 призвана исправить эти недостатки, однако на данный момент трудности существуют);
- язык XSLT не предоставляет прямого доступа к операционной системе, это несколько ограничивает возможности по генерации платформенно-специфичного кода;
- генерация нескольких исходящих документов в рамках спецификации XSLT 1.0 невозможна.

В работах, посвященных генерации кода с помощью XSLT-преобразования, код генерируется, как правило, по данным, имеющим линейную структуру, например по структуре таблиц базы данных [10] или по списку классов UML-подобных диаграмм [11]. При применении метаданных ли-

нейной структуры сложно оценить все преимущества использования XSLT-преобразования. Разработанное инструментальное средство MetaAuto демонстрирует применение XSLT-преобразований для метаданных с нелинейной структурой.

*Конфигурирование.* Инструментальное средство MetaAuto позиционируется как универсальное. Поэтому оно должно уметь «понимать» различные обозначения, применяемые в графах переходов. Необходимо иметь возможность настройки:

- синтаксиса, используемого для задания списков действий в состояниях и на переходах;
- применяемых в этих списках символов-разделителей;
- формата входных и выходных переменных;
- использующихся в логических выражениях обозначений.

Возможность конфигурирования такого рода основана на применении регулярных выражений — мощного инструмента для задания синтаксиса. Преимуществами применения регулярных выражений в качестве инструмента конфигурирования графов переходов являются:

- гибкость задания синтаксиса;
- широкое распространение регулярных выражений и наличие большого количества вспомогательных инструментов для их создания;
- большие дополнительные возможности, такие как, например, задание любого количества именованных областей.

Разработанное инструментальное средство включает в себя стандартный конфигурационный файл, описывающий обозначения графов переходов из работы [13]. Однако имеется возможность задания собственного файла конфигурации.

Рассмотрим пример создания такого файла. Создадим конфигурационный файл, поддерживающий названия входных переменных в стиле инструментального средства UniMod [5].

В графах переходов, создаваемых с помощью этого инструментального средства, входные переменные, как и выходные, имеют источник — класс, наследуемый от класса `ControlledObject` (объект управления), со следующим синтаксисом:

```
<название объекта управления>.<название входной/  
выходной переменной>
```

Например, если название объекта управления `o1`, а название входной переменной `x10`, то на графе перехода эта переменная изображается как `o1.x10`.

Такой способ наименования входных переменных не был предусмотрен в нотации графов переходов из работы [13].

Зададим формат таких входных переменных в конфигурационном файле. Предположим, что, как и в языке Java (использующемся в UniMod), названия объекта управления и входных переменных состоят из букв, цифр, знака подчеркивания и некоторых других дополнительных символов. Тогда синтаксис для каждой такой входной пере-

менной можно описать с помощью регулярных выражений вида

```
(\w+)\.(\w+)
```

Здесь `\w` означает допустимый в названии символ. Если применять в регулярном выражении именованные области, то синтаксис можно описать следующим образом:

```
(?<controlledObjectName>\w+)\.(?<inputVariableName>\w+)
```

При этом первая именованная область имеет имя `controlledObjectName` и соответствует множеству допустимых символов до точки, а вторая (с именем `inputVariableName`) — множеству допустимых символов после точки.

В процессе проверки на соответствие входной строки данному регулярному выражению, при положительном ее исходе, можно извлечь название объекта управления (именованная область `controlledObjectName`) и название входной переменной (именованная область `inputVariableName`).

Фрагмент конфигурационного файла, представленный в листинге 1, задает синтаксис входной переменной в стиле инструментального средства `UniMod`.

**Листинг 1. Фрагмент конфигурационного файла**

```
<nodeTemplate template="INPUT_VARIABLE"
  regexp="( ?<controlledObjectName>\w+)\.( ?<inputVariableName>\w+)"
  name="o${controlledObjectName}x${inputVariableName}"
  type="INPUT_VARIABLE" >
  <parameter name="object"
    value="${controlledObjectName}" />
  <parameter name="variable"
    value="${inputVariableName}" />
</nodeTemplate>
```

Этот фрагмент задает шаблон для разбора входных переменных вида `o1.x10`. Шаблон называется `INPUT_VARIABLE`. Он использует именованные области для генерации уникального имени, а также создания двух параметров `object` и `variable`.

При применении данного фрагмента в конфигурационном файле для входной переменной `o1.x10` получим объект программного представления модели (библиотеки `MetaAuto`) со значениями свойств, приведенными в листинге 2, а также изоморфное ему XML-представление, записанное в том же листинге.

**Листинг 2. Полученный объект библиотеки MetaAuto и его XML-представление**

```
Type = "INPUT_VARIABLE"
Name = "o1x10"
Parameters[0].Name = "object"
Parameters[0].Value = "o1"
Parameters[1].Name = "variable"
Parameters[1].Value= "x10"

<actionNode name="o1x10" type="INPUT_VARIABLE">
  <parameter name="object" value="o1" />
  <parameter name="variable" value="x10" />
</actionNode>
```

**Методика генерации исходного кода**

Опишем методику генерации исходного кода с применением разработанного инструментального средства. Будем преобразовывать граф переходов из работ [4, 13] в исходный код программы на языке C++.

**Шаг 1. Изображение графов переходов.** Изобразим этот граф переходов с помощью редактора `MS Visio` (рис. 2).

Для изображения графа переходов следует использовать специально созданный шаблон, содержащий все используемые на графе переходов типы элементов. Этот шаблон предоставляется вместе с инструментальным средством `MetaAuto`. Он включает в себя такие элементы, как «состояние», «переход», «описание автомата» и т. д.

**Шаг 2. Преобразование в XML-формат.** Изобразенный на первом шаге граф переходов преобразуется в XML-формат. Этот шаг включает в себя сразу два этапа (см. рис. 1): преобразование графического представления графа переходов в программное представление и изоморфное преобразование программного представления в XML-формат.

Для такого преобразования воспользуемся компонентой `Visio2Xml.exe`, являющейся частью инструментального средства. В качестве параметров передадим компоненте путь до файла, содержащего графическую модель — `analizers.vsd` и название итогового XML-файла. Для запуска из командной строки используем следующую командную строку:

```
Visio2Xml.exe analizers.vsd automatas.xml
```

Полученный XML-файл полностью описывает граф переходов.

Например, состояния и групповые состояния описываются с помощью узлов `state` XML-файла, вложенных друг в друга. Каждое состояние может включать список запускаемых в нем автоматов и выполняемых действий. На листинге 3 представлен фрагмент XML-файла.

**Листинг 3. Фрагмент полученного XML-файла. Описывает состояние «2. Предпусковые операции», вложенное в групповое состояние**

```
<state name="group_0" description="">
  Состояние 2 вложено в групповое состояние group_0
  <state name="2" description="Предпусковые операции">
    Список вложенных автоматов (A: 4,3,1)
    <stateMachineRef>
      <actionNode name="A4e0"
        type="SIMPLE_AUTOMATA_CALL_IN_STATE">
        ВЫЗОВ АВТОМАТА "4" ...
        <parameter name="automata" value="4" />
        ...с событием "0"
        <parameter name="event" value="0" />
      </actionNode>
      <actionNode name="A3e0"
        type="SIMPLE_AUTOMATA_CALL_IN_STATE">
```

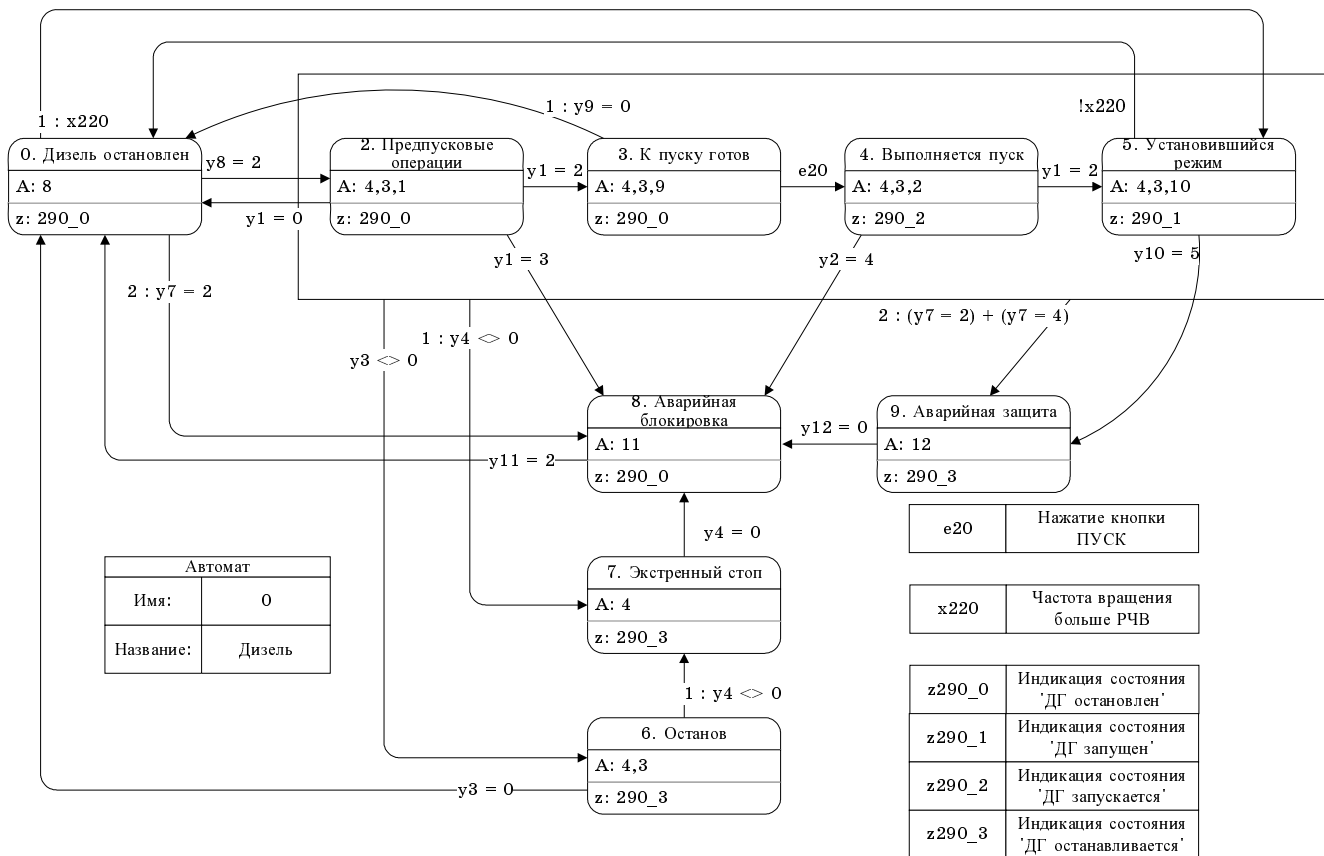


Рис. 2. Граф переходов автомата (страница A0 файла analizers.vsd)

```

<parameter name="automata" value="3" />
<parameter name="event" value="0" />
</actionNode>

<actionNode name="Ale0"
  type="SIMPLE_AUTOMATA_CALL_IN_STATE">
  <parameter name="automata" value="1" />
  <parameter name="event" value="0" />
</actionNode>

</stateMachineRef>

Список выполняемых в состоянии действий (z: 290_0)
<outputAction>
  <actionNode name="290_0"
    type="SIMPLE_OUTPUT_ACTION_IN_STATE">
    <parameter name="name" value="290_0" />
  </actionNode>
</outputAction>
</state>

...
Описание остальных состояний, вложенных в групповое, пропущено
</state>
    
```

Переходы в полученном файле описываются с помощью узлов transition XML-файла. Вложенные в них узлы описывают условие перехода, а также описывают список действий на переходе. На листинге 4 приведен фрагмент XML-файла.

Листинг 4. Фрагмент полученного XML-файла. Описывает групповой переход в состояние 9

```

Переход из группового состояния group_0 в состояние 9 с приоритетом 2 2: (y7=2)+(y7=4)
<transition sourceRef="group_0" targetRef="9"
  priority="2">
  Условие перехода (y7=2)+(y7=4)
  <condition>
    Бинарная операция OR (обозначаемая +). Использует два вложенных узла conditionNode в качестве операндов
    <binaryOperation type="OR">
      Проверка состояния автомата (y7=2)
      <conditionNode name="y7e2"
        type="OTHER_AUTOMATA_EVENT_EQUAL">
        <parameter name="automata" value="7" />
        <parameter name="event" value="2" />
      </conditionNode>
      Проверка состояния автомата (y7=4)
      <conditionNode name="y7e4"
        type="OTHER_AUTOMATA_EVENT_EQUAL">
        <parameter name="automata" value="7" />
        <parameter name="event" value="4" />
      </conditionNode>
    </binaryOperation>
  </condition>
    
```

Далее следует список действий, выполняемых на переходе. Рассматриваемый групповой переход не со-

держит никаких действий. Формат списка действий идентичен формату списка действий в состоянии

```
</transition>
```

XML-файл содержит также и другие узлы. Эти узлы соответствуют описанию действий и входных переменных автомата, названию автомата и т. д. Весь XML-файл, а также подробное описание его формата приведены в работе [6].

**Шаг 3. Создание XSLT-шаблона.** Для получения кода на языке C++ необходимо создать соответствующий шаблон. Как правило, по одному графу переходов требуется построить сразу несколько файлов. Например, в языке C++ требуется создать файл заголовков *common.h* и файл реализации *common.cpp*. В таком случае необходимо создать шаблон для каждого такого файла.

Рассмотрим шаблон для создания файла реализации *common.cpp*, как более информативный. Назовем этот шаблон *common.cpp.xslt*. В листинге 5 приведена часть этого шаблона для генерации файла, содержащего автоматную процедуру.

Другая часть шаблона, отвечающая за вывод логического выражения и различных типов входных и выходных воздействий, приведена в работе [6]. Выполняемые в ней действия весьма стандартны. Примеры создания таких шаблонов можно найти практически в любом пособии по языку XSLT.

**Листинг 5. Шаблон common.cpp.xslt**

```
<?xml version='1.0'?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:output method='text' indent="no"/>

<xsl:template match="//model">

«шапка» файла

//--- this file is machine generated ---
#include "StdAfx.h"
#include "common.h"
#include "log.h"
common_t cm;
```

Обрабатываем все автоматы в модели. Для каждого из них создаем автоматную процедуру с названием А<название автомата>

```
<xsl:for-each select="stateMachine">
void A<xsl:value-of select="@name"/>( ubyte e )
{
  ubyte y_old = cm.y<xsl:value-of select="@name"/>;
  switch( cm.y<xsl:value-of select="@name"/> )
  {
```

Рассматриваем все конечные — не групповые — состояния. Условие `count(state) = 0` показывает, что выборка будет производиться именно по таким состояниям

```
<xsl:for-each select="//state[count(state) = 0]">
<xsl:sort select="@name"/>
  Вызываем вложенные автоматы
  case <xsl:value-of select="@name"/>:
  <xsl:for-each select="stateMachineRef/actionNode">
    A<xsl:value-of select="parameter
      [ @name='automata' ]/@value"/>(e);
  </xsl:for-each>
```

Выбираем все переходы из данного состояния и переходы из групповых состояний, содержащих данное состояние. Из всех состояний текущего конечного автомата `ancestor::stateMachine/transition` выбираем те, у которых атрибут `sourceRef` совпадает с именем текущего состояния или состояния, содержащего текущее `current()/ancestor-or-self::state/@name`

```
<xsl:apply-templates
  select="ancestor::stateMachine/transition
    [current()/ancestor-or-self::state/
      @name = sourceRef]">
  <xsl:sort select="count(condition)"
    data-type="number" order="descending"/>
  <xsl:sort select="string-length(@priority) = 0"
    data-type="text" order="ascending"/>
  <xsl:sort select="@priority" data-type="number"/>
  <xsl:sort select="@targetRef" data-type="text"/>
</xsl:apply-templates>

break;
</xsl:for-each>

default:
  #ifdef A<xsl:value-of select="@name"/>_ERRORS_LOGGING
    log_write(LOG_GRAPH_ERROR, "Unknown state!");
  #else
    ;
  #endif
}

if( y_old == cm.y<xsl:value-of select="@name"/> )
  goto A<xsl:value-of select="@name"/>_end;

В случае, если состояние автомата изменилось, вызываем вложенные автоматы и выполняем действия в состоянии
switch( cm.y<xsl:value-of select="@name"/> )
{
  <xsl:for-each select="//state[count(state) = 0]" >
  <xsl:sort select="@name" />
  case <xsl:value-of select="@name"/>:
    <xsl:apply-templates
      select="stateMachineRef/actionNode" />
    <xsl:apply-templates
      select="outputAction/actionNode" />
    break;
  </xsl:for-each>
}
A<xsl:value-of select="@name"/>_end; ;
}
</xsl:for-each>
```

Далее следуют шаблоны для вывода логических выражений, действий в состоянии и на переходе. Полностью эти шаблоны приведены в работе [6]

**Шаг 4. Получение исходного кода.** Преобразуем полученный на втором шаге XML-файл в исходный код с помощью XSLT-шаблона, созданного на предыдущем шаге. Воспользуемся компонентой *XslTransform.exe* инструментального средства *MetaAuto*. Данной компоненте необходимо передать в качестве параметров путь до XML-файла (*automatas.xml*), путь до шаблона (*common.cpp.xslt*) и название итогового файла с исходным кодом (*common.cpp*).

Для запуска компоненты воспользуемся следующей командной строкой:

```
XslTransform.exe automatas.xml common.cpp.xslt common.cpp
```



В результате получим файл *common.cpp*, содержащий исходный код на языке C++, который реализует граф переходов (см. рис. 2) и представлен в листинге 6 в два столбца.

**Листинг 6.** Исходный код автоматной процедуры на языке C++. Фрагмент файла *common.cpp*, содержащего код программы

«шапка» файла

```
//- this file is machine generated
#include "StdAfx.h"
#include "common.h"
#include "log.h"
common_t cm;
```

Автоматная процедура автомата 0

```
void A0( ubyte e )
{
    ubyte y_old = cm.y0;
```

```
    switch( cm.y0 )
    {
```

Переходы из состояния 0 в состояния 0 по условию  $x220()$ , в состояние 8 по условию  $cm.y7 == 2$  и в состояние 2 по условию  $cm.y8 == 2$

```
    case 0:
        A8(e);
        if ( x220() )
        {
            cm.y0 = 5;
        }
        else if ( cm.y7 == 2 )
        {
            cm.y0 = 8;
        }
        else if ( cm.y8 == 2 )
        {
            cm.y0 = 2;
        }
        break;
```

Переходы из состояния 2 в состояния 7, 9, 0, 3, 6 и 8

```
    case 2:
        A4(e); A3(e); A1(e);
        if ( cm.y4 != 0 )
        {
            cm.y0 = 7;
        }
        else
        if ((cm.y7==2)|| (cm.y7==4))
        {
            cm.y0 = 9;
        }
        else if ( cm.y1 == 0 )
        {
            cm.y0 = 0;
        }
        else if ( cm.y1 == 2 )
        {
            cm.y0 = 3;
        }
        else if ( cm.y3 != 0 )
        {
            cm.y0 = 6;
        }
        else if ( cm.y1 == 3 )
        {
            cm.y0 = 8;
        }
        break;
```

Часть switch-блока для других состояний пропущена

```
    default:
        #ifdef A0_ERRORS_LOGGING
            log_write(LOG_GRAPH_ERROR, "Unknown state!");
        #else
            ;
        #endif
    }
```

Выполнение действий в состоянии и запуск вложенных автоматов

```
    if( y_old == cm.y0 )
        goto A0_end;
```

```
    switch( cm.y0 )
    {
```

```
        case 0:
            A8(0);
            break;
        case 2:
            A4(0);
            A3(0);
            A1(0);
            break;
```

Часть switch-блока для других состояний пропущена

```
    }
```

Далее следует остальной код

**Шаг 5. Преобразование XML-файла в документ MS Visio.** В состав инструментального средства также включена компонента обратного преобразования — файла формата XML в документ MS Visio. Для выполнения такого преобразования необходимо воспользоваться компонентой *Xml2Visio.exe*. Компоненте в качестве параметров передаются путь до файла с XML-представлением графа переходов и название итогового файла:

```
Xml2Visio.exe automatas.xml analyzers.new.vsd
```

Данная компонента по XML-файлу (*automatas.xml*) строит графическое представление графа переходов. Получаемый в результате файл отличается от исходного только расположением состояний.

**Интеграция инструментального средства со средой разработки MS Visual Studio.** Инструментальное средство MetaAuto предполагает возможность интеграции со средами разработки. Интеграцию предлагаемого инструментального средства со средой разработки MS Visual Studio рассмотрим на изложенном выше примере.

Для обеспечения интеграции можно создать *makefile* — файл, использующийся утилитой *make*. Этот файл предназначен для автоматизации процесса преобразования графа переходов в файл с исходным кодом. В рассматриваемом примере необходимо автоматизировать шаги 2 и 4. Для задания зависимостей между файлами, генерируемыми на этих шагах, используются командные строки, приведенные в описании этих шагов. Вариант *makefile* для рассматриваемого выше примера приведен в листинге 7.

**Листинг 7. makefile для автоматизации процесса генерации исходного кода**

Необходимый результат

```
result : common.h common.cpp
```

Шаг 4 для файла common.h

```
common.h : automatatas.xml common.h.xslt
XSLTransform.exe automatatas.xml common.h.xslt common.h
```

Шаг 4 для файла common.cpp

```
common.cpp : automatatas.xml common.cpp.xslt
XSLTransform.exe automatatas.xml common.cpp.xslt
common.cpp
```

Шаг 2

```
automatatas.xml : automatatas.vsd
Visio2Xml.exe automatatas.vsd automatatas.xml
```

Интеграция инструментального средства предполагает внедрение его в процесс компиляции проекта. Для этого в процесс компиляции требуется:

- добавить в проект необходимые для преобразования файлы;
- задать командную строку "\$ (ProjectDir) / nmake .exe", выполняющую преобразование в качестве события, выполняющегося до компиляции, с помощью свойства Pre-Build Event Command Line.

Программа *nmake.exe* использует *makefile* и создает файлы *common.h* *common.cpp*.

Для тестирования рассмотренного способа интеграции инструментальное средство было внедрено в процесс компиляции самого себя. С помощью инструментального средства генерируются исходные коды синтаксического и лексического анализаторов, используемых для разбора логических выражений. Исходные коды разработанного инструментального средства, в которых используется интеграция со средой разработки MS Visual Studio, размещены на сайте <http://is.ifmo.ru/projects/metaauto/>.

**Заключение**

В статье описан подход к автоматической генерации исходного кода автоматных программ. Рассмотрены различные технологии и методики, их преимущества и недостатки. Также описана реализация предложенного подхода в инструментальном средстве MetaAuto.

Это инструментальное средство разрабатывалось как генератор кода автоматных программ на любом априори заданном языке программирования. Оно было использовано для генерации кода синтаксического и лексического анализаторов, использующихся самим инструментальным средством. Таким образом, первым успешным внедрением инструментального средства стало оно само.

Однако после написания прототипа инструментального средства оказалось, что его можно использовать также и для верификации программ. Такая верификация была выполнена в работе [14]. С помощью инструментального средства процесс создания верификатора был достаточно прост, так

как граф переходов легко доступен при использовании программного представления модели.

Авторы надеются, что настраиваемость инструментального средства и легкость его использования позволят более эффективно внедрять автоматное программирование [15].

**Литература**

1. Новиков Ф. А. Визуальное конструирование программ // Информационно-управляющие системы. 2005. № 6. С. 9–22. <http://is.ifmo.ru/works/visualcons/>
2. Finite state machine. Wikipedia. The free encyclopedia. [http://en.wikipedia.org/wiki/Finite\\_automaton](http://en.wikipedia.org/wiki/Finite_automaton).
3. Сайт проекта «Finite State Machine». <http://fsme.sourceforge.net>.
4. Головешин А. Использование конвертора Visio2 SWITCH. <http://is.ifmo.ru/?i0=progeny&i1=visio2switch>.
5. Гуров В. С., Мазин М. А., Нарвский А. С., Шалыто А. А. UML SWITCH-технология. Eclipse // Информационно-управляющие системы. 2004. № 6. С. 12–17. <http://is.ifmo.ru/works/uml-switch-eclipse/>.
6. Канжелев С. Ю., Шалыто А. А. Преобразование графов переходов, представленных в формате MS Visio, в исходные коды программ для различных языков программирования (инструментальное средство MetaAuto). 102 с. <http://is.ifmo.ru/projects/metaauto/>.
7. Чарнецки К., Айзенекер У. Порождающее программирование: методы, инструменты, применение. СПб.: Питер, 2005. 736 с.
8. Селлз К. Современные способы автоматизации повторяющихся задач программирования // MSDN Magazine. 2002. N 6. P. 8–13.
9. Dodds L. Code generation using XSLT, [ibm.com/developerWorks](http://ibm.com/developerWorks), <http://www-106.ibm.com/developerworks/edu/x-dw-codexslt-i.html>.
10. Ashley P. Simplify Development and Maintenance of Microsoft .NET Projects with Code Generation Techniques, <http://msdn.microsoft.com/msdnmag/issues/03/08/CodeGeneration/default.aspx>.
11. Herrington J. Extensible Code Generation with Java, <http://today.java.net/pub/a/today/2004/05/12/generation1.html>.
12. Заякин Е. А., Шалыто А. А. Метод устранения вторичных фрагментов кода при реализации конечных автоматов. 21 с. [http://is.ifmo.ru/projects/life\\_app](http://is.ifmo.ru/projects/life_app).
13. Шалыто А. А., Туккель Н. И. SWITCH-технология — автоматный подход к созданию программного обеспечения «реактивных» систем // Программирование. 2001. № 5. С. 45–62. <http://is.ifmo.ru/works/switch/>.
14. Канжелев С. Ю., Шалыто А. А. Моделирование кнопочного телефона с использованием SWITCH-технологии. Вариант 2. 104 с. <http://is.ifmo.ru/projects/phone/>.
15. Шалыто А. А. Технология автоматного программирования // Мир ПК. 2003. № 10. С. 74–78. [http://is.ifmo.ru/works/tech\\_aut\\_prog/](http://is.ifmo.ru/works/tech_aut_prog/)

УДК 519.872.3

## СРЕДНИЕ ВРЕМЕНА ОЖИДАНИЯ И ПРЕБЫВАНИЯ В МНОГОКАНАЛЬНЫХ ПРИОРИТЕТНЫХ СИСТЕМАХ

Ю. И. Рыжиков,

доктор техн. наук, профессор

Военно-космическая академия им. А. Ф. Можайского

Предлагается способ расчета средних времен ожидания и пребывания заявок в  $n$ -канальной системе с относительным, абсолютным и смешанным приоритетами при произвольных и различных по типам заявок распределениях длительностей чистого обслуживания. Точность расчетов иллюстрируется сопоставлением с результатами имитационного моделирования и пересчетов на основе «инвариантов отношения». Обсуждается расчет «замкнутых» приоритетных систем.

A method is proposed to compute mean waiting and sojourn times for  $n$ -channel system with preemptive, non-preemptive and mixed priorities. Service time distributions are supposed to be distinct and arbitrary. The precision of the algorithm is demonstrated by a comparison with the results of imitation and calculations on the basis of "relation invariants". Closed priority systems are discussed.

### Введение

Задачи управления сложными техническими комплексами в реальном времени обычно требуют использования многомашинных или многопроцессорных систем — как из соображений производительности, так и с учетом требований надежности и организации технического обслуживания последних. Различия в важности задач, их трудоемкости и требованиях к оперативности решения приводят к введению *приоритетных* дисциплин обслуживания.

Методы анализа одноканальных систем с приоритетами разработаны достаточно хорошо (см., например, работы [1, 4, 5, 11]), хотя численная их реализация при необходимости получения преобразований Лапласа и/или высших моментов ставит ряд непростых проблем, отчасти решенных [10, 12, 17]. Однако сложность анализа приоритетных дисциплин обслуживания резко возрастает при переходе к *многоканальным* системам. Такие задачи обычно решаются лишь в простейшем (экспоненциальном) случае, причем средние длительности обслуживания предполагались различными буквально в единичных работах [7, 20, 23, 24]. К тому же решение (методом производящих функций) оказывалось весьма громоздким.

В статье А. Д. Хомоненко [21] предлагается фазовая аппроксимация системы с абсолютным приоритетом и двумя классами заявок с экспоненциальным обслуживанием, рассчитываемая итерационным методом Такахаси—Таками [13, 14, 25]. Для распределений длительности обслужива-

ния, отличных от показательных, диаграммы переходов существенно усложняются. Еще сложнее анализ многоканальных систем в случае относительных приоритетов. Здесь редуцированная схема требует учета трех потоков — данного ( $j$ -го) и двух объединенных потоков высших и низших приоритетов соответственно. Группирование типов заявок требует перехода к средневзвешенному распределению обслуживания, которое при различных экспоненциальных составляющих уже не может быть аппроксимировано показательным законом с приемлемой погрешностью. Практическую ценность могут иметь только алгоритмы, обобщенные в указанных направлениях. С другой стороны, все попытки создания реально применимых методик, учитывающих находящиеся в каналах и очередях количества заявок каждого вида, заведомо обречены на неудачу в связи с непомерным разрастанием пространства состояний.

Ниже предлагаются приближенные, но, как показали машинные эксперименты, достаточно точные способы расчета средних времен ожидания  $\{w_{j,i}\}$  заявок в  $n$ -канальной системе с относительным, абсолютным (с дообслуживанием) и смешанным статическими приоритетами при пуассоновских входящих потоках и произвольных (различных) распределениях длительностей чистого обслуживания. Режим работы системы предполагается стационарным.

Сразу же отметим, что для всех обсуждаемых дисциплин обслуживания средние времена пребывания заявки в системе получаются добавлением

к  $\{w_{j,1}\}$  первых моментов распределения длительностей обслуживания:  $v_{j,1} = w_{j,1} + b_{j,1}$ .

**Компоненты проблемы**

Поставленные задачи сводятся к расчету среднего времени недоступности системы для обслуживания «меченой» заявки типа  $j$  (типы заявок упорядочены по убыванию приоритетов). В случае приоритета с прерыванием к нему добавляется произведение среднего числа прерываний меченой  $j$ -заявки на среднюю длительность каждого прерывания. Решения этих задач для одноканальной системы элементарны. Попытаемся найти их аналогии для  $n$ -канальной.

**Среднее время ожидания**

Выпишем известные выражения для первых двух моментов распределения времени ожидания меченой  $j$ -заявки в одноканальной системе:

$$w_{j,1} = \frac{\sum_{i=1}^k \lambda_i b_{i,2}}{2(1 - R_{j-1})(1 - R_j)}; \tag{1}$$

$$w_{j,2} = \frac{\sum_{i=1}^k \lambda_i b_{i,3}}{3(1 - R_{j-1})(1 - R_j)} + \frac{\left(\sum_{i=1}^k \lambda_i b_{i,2}\right) \left(\sum_{i=1}^j \lambda_i b_{i,2}\right)}{2(1 - R_{j-1})^2 (1 - R_j)^2} + \frac{\left(\sum_{i=1}^k \lambda_i b_{i,2}\right) \left(\sum_{i=1}^{j-1} \lambda_i b_{i,2}\right)}{2(1 - R_{j-1})^3 (1 - R_j)}. \tag{2}$$

Здесь  $R_j = \sum_{i=1}^j \lambda_i b_{i,1}$  — кумулянтный коэффициент загрузки системы заявками до  $j$ -го типа включительно. Заметим, что формула (2) в наиболее доступных источниках [4, 6, 8, 11] дана с различными ошибками. Приводимый здесь вариант [3, с. 168], согласуется с результатами имитационной модели. Предполагается, что расчеты по этой системе (1), (2) будут выполняться при интенсивностях  $\{\lambda_i\}$  исходных потоков, пересчитанных на один канал.

Для  $j$ -заявки, прибывающей в многоканальную систему, период «общей» недоступности заканчивается при завершении таковой хотя бы в одном из каналов. Соответственно дополнительная функция распределения недоступности

$$\overline{W}_n(t) = \left[ \overline{W}_1(t) \right]^n. \tag{3}$$

Здесь для упрощения обозначений индекс указывает число каналов (тип заявок  $j$  подразумевается). Интересующее нас среднее значение можно найти по хорошо известной формуле из теории надежности

$$w_1(n) = \int_0^{\infty} \overline{W}_n(t) dt.$$

Простейшей реализацией этого подхода является экспоненциальная аппроксимация распределения времени ожидания в одном канале, для чего достаточно первого момента. Это допущение автоматически приводит к средней длительности ожидания, обратно пропорциональной числу каналов, что заметно расходится с экспериментальными данными (см. ниже). Для аппроксимации распределения по двум моментам удобно воспользоваться распределением Вейбулла

$$\overline{W}_1(t) = \exp(-t^k / T) \tag{4}$$

с моментами

$$w_m = T^{m/k} \Gamma(1 + m/k), \quad m = 1, 2, \dots \tag{5}$$

Определим параметры последнего приравнянием его моментов, найденными из системы (1), (2). Тогда отношение

$$\alpha = w_2 / w_1^2 = 2k\Gamma(2/k) / \Gamma^2(1/k) = 2\Gamma(2u) / (u\Gamma^2(u)), \tag{6}$$

где  $u = 1/k$ . Воспользовавшись формулой удвоения аргумента гамма-функции (см. [22], с. 55), можно переписать (6) в виде

$$\alpha = \frac{2^{2u} \Gamma(u) / \Gamma(u + 1/2)}{u\sqrt{\pi} \Gamma^2(u)} = \frac{2^{2u} \Gamma(u + 1/2)}{\sqrt{\pi} \Gamma(u + 1)},$$

откуда следует обеспечивающая быстро сходящийся итерационный процесс уточнения переменной ( $u$ ) формула

$$u_i = \frac{1}{2 \ln 2} \ln \frac{\alpha \sqrt{\pi} \Gamma(u_{i-1} + 1)}{\Gamma(u_{i-1} + 1/2)}, \quad i = 1, 2, \dots \tag{7}$$

Начальное приближение

$$u_0 = \ln 2\alpha / (2 \ln 2). \tag{8}$$

Решив уравнение (7), вычисляем  $k = 1/u$ ,  $T = (w_1 / \Gamma(u + 1))^k$ . Применяя формулу (3) к распределению Вейбулла, убеждаемся, что она приводит к распределению того же типа с пересчитанным параметром  $T_n = T/n$  и прежним  $k$ . Соответственно интересующее нас среднее время ожидания в  $n$ -канальной системе

$$w_1(n) = (T/n)^{1/k} \Gamma(1 + 1/k) = \frac{1}{k} (T/n)^{1/k} \Gamma(1/k).$$

В случае абсолютных приоритетов (с прерываниями) меченая  $j$ -заявка просто не замечает присутствия в системе заявок типов  $i > j$ . Здесь следует в уравнениях для моментов распределения ожидания заменить на  $j$  все верхние пределы суммирования, равные количеству типов заявок  $k$ .



**Период непрерывной занятости прерываниями**

Описанный выше подход естественно применить и к аппроксимации распределения периода непрерывной занятости (ПНЗ)  $n$ -канальной системы — для начала применительно к потоку однородных заявок. Тогда средняя длина ПНЗ

$$\pi(n) = \frac{b_1}{n(1 - \lambda b_1/n)}, \quad (9)$$

где  $b_1$  — средняя длительность обработки головной заявки периода занятости. В имитационном эксперименте эта зависимость подтвердилась только для показательного распределенной длительности обслуживания, а в остальных случаях могла рассматриваться лишь как грубое приближение. В связи с этим было решено искать аппроксимацию упомянутой зависимости в виде

$$\pi(n) = \frac{b_1[1 + \Delta(\rho, \theta, n)]}{n(1 - \rho)}, \quad (10)$$

где  $\rho$  — коэффициент загрузки канала;  $\theta$  — коэффициент вариации обслуживания. Необходимая поправка  $\Delta$  вычислялась через наблюдаемое в имитационном эксперименте среднее значение периода непрерывной занятости  $\pi(n)$  по формуле

$$\Delta = \frac{n(1 - \rho)}{b_1} \pi(n) - 1.$$

Отмеченное выше нулевое значение поправки для показательного распределения обслуживания

определило ее мультипликативное строение и необходимость обращения в нуль при  $\theta = 1$ . Кроме того, в широком диапазоне коэффициентов вариации зависимость от  $\theta$  при прочих равных условиях оказалась близка к линейной. Далее, модуль поправки был приблизительно пропорционален  $\rho$ . Наконец, поправка росла по числу каналов  $n$ , обнаруживая тенденцию к насыщению, и, по определению, равнялась нулю при  $n = 1$  (последнее требование — еще один аргумент за мультипликативную форму поправки). В итоге оказалось, что для использования в (10) можно принять

$$\Delta = \rho(\theta - 1) \frac{n - 1}{4n}$$

(числовой множитель «4» подобран экспериментально).

Результаты вычисления средней длины периода непрерывной занятости на имитационной модели (И) и по расчету (Р) представлены в табл. 1. Распределением заменялось гамма-распределение с коэффициентом вариации 2. Во всех случаях полагалось  $b_1 = 1$ .

**Ожидаемое число прерываний**

Вновь поровну разделим входящий поток между каналами обслуживания. Для одноканальной системы ожидаемое число прерываний  $k_j = \Lambda_{j-1} b_{j,1}$ , где  $\Lambda_{j-1} = \sum_{i=1}^{j-1} \lambda_i$  есть интенсивность потока заявок с правом прерывания  $j$ -й. В  $n$ -канальном случае заявка, прибывшая во время обслуживания  $j$ -й, может вообще ее не прерывать (если хотя бы

■ Таблица 1. Средние длительности непрерывной занятости

n	ρ	Распределение обслуживания							
		D		E <sub>3</sub>		M		H <sub>2</sub>	
		И	Р	И	Р	И	Р	И	Р
1	0,5	1,997	2,000	1,998	2,000	2,003	2,000	2,008	2,000
	0,7	3,332	3,333	3,339	3,333	3,349	3,333	3,325	3,333
	0,9	9,971	10,000	10,121	10,000	9,975	10,000	9,510	10,000
2	0,5	0,917	0,938	0,974	0,974	0,998	1,000	1,053	1,062
	0,7	1,496	1,521	1,599	1,605	1,673	1,667	1,781	1,812
	0,9	4,421	4,438	4,732	4,762	4,972	5,000	5,110	5,562
3	0,5	0,593	0,611	0,633	0,643	0,667	0,667	0,723	0,722
	0,7	0,959	0,981	1,044	1,056	1,114	1,111	1,243	1,241
	0,9	2,809	2,833	3,113	3,122	3,312	3,333	3,609	3,813
4	0,5	0,441	0,453	0,473	0,480	0,502	0,500	0,553	0,547
	0,7	0,708	0,724	0,774	0,787	0,836	0,833	0,950	0,943
	0,9	2,048	2,078	2,275	2,321	2,517	2,500	2,747	2,922
5	0,5	0,351	0,360	0,377	0,383	0,401	0,400	0,443	0,440
	0,7	0,560	0,573	0,614	0,627	0,667	0,667	0,773	0,760
	0,9	1,619	1,640	1,810	1,848	1,990	2,000	2,262	2,360

■ Таблица 2. Средние кратности прерываний

n	Тип 2		Тип 3	
	И	Р	И	Р
1	0,180	0,180	0,674	0,674
2	0,116	0,082	0,827	0,844
3	0,071	0,035	0,882	0,920
4	0,043	0,014	0,892	0,964
5	0,026	0,006	0,897	0,993

один канал свободен или занят обслуживанием менее приоритетной заявки). С другой стороны, меченую заявку могут прервать и те, которые «изначально» пришлось бы на другие каналы. Ясно, что первый эффект будет преобладать для заявок относительно высокого приоритета, а второй — для низкого. Прерывание обслуживаемой  $j$ -заявки может произойти, если все остальные каналы заняты обслуживанием не менее приоритетных заявок — вероятность  $(R_j/R_k)^{n-1}$ . Потенциальные прерывания распределяются равномерно между всеми  $j$ -заявками, одновременно находящимися на обслуживании. В среднем их будет  $1 + (n-1)\lambda_j b_{j,1}/R_j$ . Таким образом, в многоканальном случае

$$\frac{1}{k_j} = \frac{\Lambda_{j-1} b_{j,1} (R_j/R_k)^{n-1}}{1 + (n-1)\lambda_j b_{j,1}/R_j}.$$

Эта формула при  $n = 1$  является точной.

В табл. 2 результаты расчета сопоставляются с полученными на имитационной модели. Рассматривалась система с тремя типами заявок при средних длительностях обслуживания  $b_{1,1} = 0,45$ ,  $b_{2,1} = 0,90$ ,  $b_{3,1} = 1,35$  и интенсивностях потоков на канал  $\lambda_1 = 0,2$ ,  $\lambda_2 = 0,3$ ,  $\lambda_3 = 0,4$  (коэффициент загрузки  $R_3 = 0,9$ ). Поскольку обнаружилось, что типы распределений длительности обслуживания на кратность прерываний практически не влияют, объем таблицы, соответственно, сокращен.

Таблица иллюстрирует качественное соответствие результатов ожидаемым и неплохое согласие полученных кратностей для низкоприоритетных заявок, т. е. как раз там, где прерывания могут дать заметную дополнительную задержку. Для заявок высокого приоритета согласие заметно хуже, но сами кратности весьма малы, да и длительности прерываний тоже будут малыми. Здесь итоговая абсолютная погрешность суммарной средней длительности прерываний заведомо несущественна.

### Использование инвариантов отношения

Для расчета средних времен ожидания в многоканальных приоритетных системах предлагалось [1] использование инвариантов отношения — символической пропорции вида

$$\frac{M/G/1}{M/G/n} = \frac{\bar{M}/\bar{G}/1}{\bar{M}/\bar{G}/n}.$$

Таким образом, средние времена ожидания в системе  $\bar{M}/\bar{G}/n$  должны вычисляться по формуле

$$w_j(n) = w_j(1) \frac{w(n)}{w(1)}.$$

Здесь  $\{w_j(1)\}$  определяются по формуле (1). Дробь в правой части последнего равенства есть отношение средних времен ожидания в одно- и многоканальных системах с однородными потоками заявок суммарной интенсивности и средневзвешенными по интенсивностям моментами распределений длительности обслуживания. Эта дробь для наиболее интересного практически диапазона  $n < 5$  хорошо аппроксимируется формулой

$$\frac{w(n)}{w(1)} = \frac{1}{n^2} \exp[-0,7(n-1)(1-\rho)]. \quad (11)$$

С помощью инвариантов отношения можно рассчитывать и системы с абсолютным приоритетом. В этом случае в пересчете должны участвовать суммарные времена ожидания — начала обслуживания и общей длительности прерываний. К результатам пересчета следует добавить чистую длительность обслуживания.

### Сопоставление конечных результатов

Предложенные подходы были запрограммированы на Фортране 77. Необходимые для работы с инвариантами отношения расчеты беспriorитетных и одноканальных приоритетных систем выполнялись на расширенной и переведенной на Фортран версии пакета МОСТ [10]. За эталон были взяты результаты имитационного моделирования многоканальных систем при 200 тыс. наблюдений по заявкам высшего приоритета. Результаты счета выводились в точках логарифмической шкалы (1, 2, 5, 10, 20, 50, 100, 200 тыс. заявок этого типа). К указанному граничному значению они стабилизировались в трех старших знаках.

При моделировании системы с абсолютным приоритетом длительность обслуживания заявки определялась в момент ее постановки в очередь: при входе в систему — с помощью соответственно настроенного датчика случайных чисел, после прерывания — с остатком ранее сформированной длительности. Для относительного приоритета определялось более чувствительное к ошибкам время ожидания, для абсолютного — полное время пребывания.

Модели верифицировались на одноканальной задаче при исходных данных, приведенных в разделе о кратности прерываний. Достаточно большой суммарный коэффициент загрузки ( $R = 0,9$ ) обеспечивал существенную роль длительностей ожидания и прерываний низкоприоритетных зая-

■ Таблица 3. Средние времена ожидания в системе с относительным приоритетом

n	Тип	Распределение обслуживания											
		D			E <sub>3</sub>			M			H <sub>2</sub>		
		И	О	Р	И	О	Р	И	О	Р	И	О	Р
1	1	0,556	0,556	0,556	0,745	0,742	0,742	1,114	1,113	1,113	2,787	2,782	2,782
	2	0,869	0,869	0,869	1,162	1,159	1,159	1,737	1,738	1,738	4,350	4,346	4,346
	3	7,930	7,910	7,910	10,764	10,547	10,547	15,600	15,820	15,820	38,458	39,551	39,551
2	1	0,335	0,265	0,339	0,393	0,352	0,387	0,525	0,526	0,514	1,133	1,303	1,143
	2	0,499	0,415	0,525	0,620	0,551	0,631	0,872	0,822	0,868	1,970	2,037	1,987
	3	3,862	3,773	3,718	4,911	5,011	4,919	7,518	7,480	7,321	18,330	18,532	18,138
3	1	0,241	0,170	0,253	0,267	0,226	0,264	0,335	0,336	0,327	0,644	0,825	0,679
	2	0,360	0,266	0,391	0,430	0,352	0,442	0,568	0,524	0,578	1,198	1,259	1,257
	3	3,397	2,242	2,390	3,207	3,208	3,148	4,715	4,772	4,665	11,465	11,729	11,496
4	1	0,186	0,124	0,206	0,199	0,163	0,201	0,243	0,242	0,237	0,423	0,586	0,470
	2	0,280	0,193	0,317	0,325	0,255	0,344	0,421	0,379	0,433	0,828	0,916	0,908
	3	1,716	1,758	1,747	2,290	2,323	2,294	3,454	3,446	3,388	8,044	8,331	8,318
5	1	0,152	0,096	0,176	0,157	0,127	0,163	0,193	0,187	0,185	0,312	0,438	0,353
	2	0,233	0,150	0,270	0,261	0,198	0,283	0,340	0,293	0,347	0,630	0,684	0,706
	3	1,373	1,366	1,370	1,770	1,802	1,794	2,897	2,663	2,644	6,679	6,226	6,472

вок. В многоканальных случаях тот же коэффициент загрузки поддерживался умножением интенсивностей потоков на  $n$ .

Результаты расчета сведены в табл. 3 и 4. Данные для каждого распределения представлены по

схеме Имитация/инвариант Отношения/Расчет по предлагаемым методикам.

Анализ результатов показывает, что оба обсуждаемых подхода имеют право на применение. Однако погрешность инвариантов отношения для

■ Таблица 4. Средние времена пребывания в системе с абсолютным приоритетом

n	Тип	Распределение обслуживания											
		D			E <sub>3</sub>			M			H <sub>2</sub>		
		И	О	Р	И	О	Р	И	О	Р	И	О	Р
1	1	0,472	0,472	0,472	0,480	0,480	0,480	0,493	0,495	0,495	0,559	0,561	0,561
	2	1,233	1,232	1,188	1,311	1,314	1,269	1,478	1,476	1,434	2,192	2,206	2,162
	3	10,038	10,020	9,944	12,875	12,656	12,580	18,364	17,930	17,854	40,557	41,660	41,584
2	1	0,452	0,461	0,453	0,452	0,464	0,454	0,454	0,471	0,455	0,459	0,502	0,462
	2	1,000	1,059	0,998	1,020	1,097	0,997	1,064	1,172	1,034	1,223	1,512	1,203
	3	5,718	5,485	5,316	6,840	6,722	6,521	9,536	9,189	8,929	20,065	20,238	19,797
3	1	0,450	0,457	0,451	0,450	0,459	0,451	0,450	0,463	0,452	0,449	0,483	0,453
	2	0,942	1,002	0,937	0,948	1,026	0,944	0,959	1,074	0,959	1,019	1,287	1,030
	3	4,113	4,004	3,848	4,858	4,789	4,608	6,398	6,351	6,127	13,168	13,304	12,965
4	1	0,450	0,455	0,450	0,451	0,456	0,451	0,452	0,460	0,451	0,446	0,473	0,451
	2	0,919	0,974	0,922	0,923	0,991	0,925	0,926	1,025	0,933	0,951	1,175	0,972
	3	3,350	3,277	3,148	3,911	3,841	3,695	5,053	4,961	4,791	9,597	9,841	9,725
5	1	0,450	0,454	0,450	0,450	0,450	0,450	0,457	0,450	0,450	0,447	0,468	0,450
	2	0,910	0,957	0,915	0,911	0,971	0,917	0,916	0,997	0,924	0,920	1,106	0,946
	3	2,956	2,847	2,745	3,399	3,281	3,170	4,311	4,141	4,020	7,920	7,696	7,850

немарковских распределений может быть недопустимо велика, тогда как методики данной статьи приемлемую точность обеспечивают во всем исследованном диапазоне параметров. Их результаты не обнаруживают тенденций к росту относительных погрешностей при приближении к границам упомянутого диапазона. С другой стороны, в тех случаях, когда инварианты отношения лучше согласуются с имитацией, погрешность обоих подходов вполне допустима.

### Смешанный приоритет

Если в схеме абсолютных приоритетов имеется  $k$  типов заявок, то наивысшая кратность прерывания составляет  $k - 1$ . В частности, для работы вычислительной системы необходимо иметь столько же областей сохранения (запоминания состояния) прерванной заявки. При больших  $k$  имеет смысл группировать типы заявок в классы с общим абсолютным приоритетом и относительными приоритетами внутри классов [2]. Такой подход снижает суммарную частоту прерываний и количество потребных областей сохранения, оставляя по одной на каждый класс, кроме первого. В то же время он способен обеспечить большую дифференциацию оперативности обслуживания, чем схема относительного приоритета.

Рассмотрим расчет среднего времени ожидания для заявки  $j$ -го типа. Обозначим через  $j+$  наибольший индекс заявок, прерывающих  $j$ -ю, т. е. нижнюю границу предшествующего класса, а через  $j-$  нижнюю границу класса, содержащего  $j$ -заявки. Заявка  $j$ -го типа будет ждать завершения начатого обслуживания, если она не имеет права его прерывания. Следовательно, в формулах для расчета времени ожидания начала обслуживания наибольший индекс  $k$  должен быть заменен на  $j-$ .

Период недоступности канала для  $j$ -заявки может быть начат любой заявкой, имеющей право ее прерывания (т. е. индекс до  $j+$  включительно). Соответственно интенсивность потока прерываний и средневзвешенные моменты распределения длительности обслуживания считаются по этому множеству индексов. Если разрешить присоединяться к прерываниям более приоритетным заявкам одного класса с  $j$ -й, то коэффициент загрузки по прерывающим заявкам останется  $R_{j-1}$ .

Рассмотренная схема охватывает как частные случаи абсолютный приоритет (один тип заявок в каждом классе), относительный (единственный класс) и стандартный вариант смешанного (все типы заявок с абсолютным приоритетом образуют отдельные классы, заявки с относительным приоритетом объединяются в самый младший класс).

### Замкнутые системы

В приложениях часто встречаются приоритетные системы замкнутого типа — с конечными объемами источников  $\{K_j\}$ ,  $j = 1, k$ . Рассмотрим техни-

ку их приближенного расчета. Прежде всего отметим, что полный цикл оборота  $j$ -заявки составляет

$$T_j = t_j + w_j + b_{j,1},$$

где  $t_j$  — средняя задержка в источнике;  $w_j$  — среднее время ожидания обслуживания. Соответственно средняя интенсивность их потока

$$\lambda_j = K_j / (t_j + w_j + b_{j,1}), \quad (12)$$

она максимальна при нулевых  $\{w_j\}$ . Умножая обе части этого равенства на  $b_{j,1}$  и суммируя по  $j$ , получаем *достаточное* условие корректности исходных данных в форме

$$\sum_{j=1}^k K_j b_{j,1} / (t_j + w_j + b_{j,1}) < n$$

(тогда гарантируется отсутствие перегрузки «разомкнутой» системы).

Поскольку, согласно (12), интенсивности входящих потоков зависят от искомым средних длительностей ожидания, последние приходится определять методом итераций:

- 1) задать начальными значениями  $w_j = 0$ ;
- 2) для всех  $j$  вычислить  $\{\lambda_j\}$  согласно (12);
- 3) для всех  $j$ , применяя приведенные формулы в зависимости от типа приоритета, рассчитать средние времена ожидания  $\{w_j\}$  и согласно (12) — новые интенсивности  $\{\lambda'_j\}$ ;
- 4) если  $\max_j \{|\lambda'_j / \lambda_j - 1|\} > \varepsilon$ , заменить  $\{\lambda_j\}$  на  $\{\lambda'_j\}$ . Перейти к этапу 3;
- 5) конец алгоритма.

Итерации по этой схеме оказались неустойчивыми с явными признаками «слишком энергичной» пошаговой коррекции. Поэтому интенсивности потоков во всех итерациях, начиная со второй, определялись через полусумму средних времен ожидания  $j$ -заявки на двух последних шагах. Стабилизация пяти значащих цифр наступала примерно за 10 шагов практически независимо от исходных данных.

Расчет выполнялся для трех типов заявок при средних длительностях обслуживания  $\{0,45; 0,90; 1,35\}$  и задержках в источнике  $\{50, 30, 10\}$ . Численность популяций при  $n = 3$  и  $n = 1$ , удовлетворяющих условию (12), составила  $\{4, 13, 21\}$  и  $\{1, 4, 7\}$  соответственно. При обслуживании с коэффициентом вариации 2 были получены средние времена ожидания  $\{0,526; 0,647; 3,254\}$  и  $\{2,056; 2,328; 6,517\}$ , а при детерминированном обслуживании с теми же средними —  $\{0,251; 0,290; 1,506\}$  и  $\{0,510; 0,581; 2,773\}$ .

### Выводы

Предложенная методика позволяет, наконец, сравнительно просто рассчитывать на уровне средних многоканальные приоритетные системы с произвольными и различными распределениями вре-



мени обслуживания. В отличие от метода инвариантов, она не требует сложного программирования и трудоемкого расчета многоканальных систем с немарковским обслуживанием и дает достаточную точность. В сравнении с имитационным моделированием методика существенно выигрывает в

простоте программирования и объеме вычислений. Кстати, отметим, что с помощью популярной системы GPSS нужную модель не построить, поскольку названная система допускает лишь однократные прерывания, причем исключительно для одноканальных устройств.

## Литература

1. **Бронштейн О. И., Духовный И. М.** Модели приоритетного обслуживания в информационно-вычислительных системах. М.: Наука, 1976. 220 с.
2. Вычислительные системы и их программное обеспечение: модели, методы и средства исследования: Учебник / Под ред. Ю. И. Рыжикова и А. Д. Хомоненко. МОРФ, 1996. 312 с.
3. **Джейсуол Н. К.** Очереди с приоритетами: Пер. с англ. М.: Мир, 1973. 279 с.
4. **Климов Г. П.** Стохастические системы обслуживания. М.: Наука, 1966. 243 с.
5. **Конвей Р. В., Максвелл В. Л., Миллер Л. В.** Теория расписаний: Пер. с англ. М.: Наука, 1975. 359 с.
6. **Липаев В. В., Яшков С. Ф.** Эффективность методов организации вычислительного процесса в АСУ. М.: Статистика, 1975. 255 с.
7. **Лысенкова В. Т.** Исследование многолинейных систем массового обслуживания с ограниченным накопителем и приоритетами: Автореф. дис. ... канд. техн. наук / Ин-т проблем передачи информации. М., 1973. 16 с.
8. **Мова В. В., Пономаренко Л. А., Калиновский А. М.** Организация приоритетного обслуживания в АСУ. Киев: Техника, 1977. 160 с.
9. **Основы теории вычислительных систем** / Под ред. С. А. Майорова. М.: Высш. шк., 1978. 408 с.
10. **Пакет прикладных программ МОСТ** для расчета стационарных режимов в системах массового обслуживания / Эстонское НИИ ВТИ. 1988.
11. **Приоритетные системы обслуживания.** М.: МГУ, 1973. 447 с.
12. **Рыжиков Ю. И.** Комплекс программ для расчета систем массового обслуживания повышенной сложности // Программирование. 1978. № 4. С. 87–91.
13. **Рыжиков Ю. И.** Алгоритм расчета многоканальной системы с эрланговским обслуживанием // Автоматика и телемеханика. 1980. № 5. С. 30–37.
14. **Рыжиков Ю. И., Хомоненко А. Д.** Итеративный метод расчета многоканальных систем с произвольным распределением времени обслуживания // Проблемы управления и теории информации. 1980. № 3. С. 32–38.
15. **Рыжиков Ю. И.** Тестирование функционально избыточных пакетов программ // Программирование. 1986. № 1. С. 22–29.
16. **Рыжиков Ю. И.** Тестирование комплекса программ по расчету систем массового обслуживания // Управляющие системы и машины. 1986. № 2. С. 84–89.
17. **Рыжиков Ю. И.** Численная реализация преобразования Лапласа и его обращения в задачах массового обслуживания // 7-я Белорусская школа-семинар по теории массового обслуживания: Тез. докл. Минск, 1991. С. 111–112.
18. **Саати Т. Л.** Элементы теории массового обслуживания и ее приложения: Пер. с англ. М.: Сов. радио, 1965. 510 с.
19. **Сотский Н. М., Чуркин Е. А.** Стационарное распределение длины очереди в многоканальной системе массового обслуживания с приоритетами // Автоматика и телемеханика. 1985. № 1. С. 69–76.
20. **Томашевский В. Л.** Многоканальные приоритетные системы массового обслуживания: Автореф. дис. ... канд. техн. наук. М.: МГУ, 1986. 14 с.
21. **Хомоненко А. Д.** Вероятностный анализ приоритетного обслуживания с прерываниями в многопроцессорных системах // Автоматика и вычислительная техника. 1990. № 2. С. 55–61.
22. **Янке Е., Эмде Ф., Леш Ф.** Специальные функции: Пер. с нем. М.: Наука-Физмат, 1968. 344 с.
23. **Gail H. R., Hantler S. L., Taylor B. A.** Analysis of a non-preemptive priority multiserver queue // Advances in applied prob. 1988. Vol. 20. P. 852.
24. **Miller D. R.** Steady-state algorithmic analysis of  $M/M/c$  two priority queues with heterogeneous rates // Applied probability — computer science: the interface. Vol. 2. Boston: Birkhauser. 1982. P. 207–222.
25. **Takahashi Y., Takami Y.** A numerical method for the steady-state probabilities of a  $GI/G/c$  queuing system in a general class. // J. of the Operat. res. soc. of Japan. 1976. Vol. 19. N 2. P. 147–157.

УДК 621.313.13

## СРАВНЕНИЕ МЕХАНИЧЕСКИХ ХАРАКТЕРИСТИК АСИНХРОННЫХ И ВЕНТИЛЬНЫХ ИНДУКТОРНО-РЕАКТИВНЫХ ДВИГАТЕЛЕЙ

**Ю. А. Голландцев,**

доктор техн. наук,

ведущий научный сотрудник ЦНИИ «Электроприбор»

*Сравниваются механические характеристики асинхронных двигателей с короткозамкнутым ротором и вентильных индукторно-реактивных двигателей, предназначенных для применения в системах автоматического управления. Обсуждаются результаты сравнения характеристик и даются рекомендации по выбору двигателей.*

*The mechanical features of asynchronous motors are compared with short-circuit rotor and switched reluctance motors, intended for use in automatic control systems. We discuss the results of the comparison and give recommendations on the choice of the motors for specific purposes.*

В настоящее время в системах автоматического управления в основном используются асинхронные двигатели (АД) с короткозамкнутым ротором, вентильные двигатели (ВД) с возбуждением от постоянных магнитов и значительно реже шаговые и вентильные индукторно-реактивные двигатели (ВИРД).

Отлаженная автоматизированная технология производства, низкая стоимость и широкая номенклатура выпускаемых двигателей являются основными причинами широкого использования АД в регулируемом и нерегулируемом электроприводе. К недостаткам АД можно отнести относительно малый воздушный зазор, повышенные тепловые потери в обмотках ротора и статора, а также сложные алгоритмы частотного и векторного управления частотой вращения и моментом. Однако с появлением доступных микропроцессорных средств, которые позволяют реализовать практически любые алгоритмы управления АД, последний недостаток может рассматриваться с позиции надежности работы предложенных алгоритмов управления в режиме реального времени.

Номенклатура выпускаемых синхронных двигателей с возбуждением от постоянных магнитов и, соответственно, ВД, построенных на их базе, очень ограничена. К основным достоинствам ВД относятся высокие энергетические показатели и линейные алгоритмы управления, а недостатки определяются необходимостью установки датчика положения ротора, наличием постоянных магнитов и сложной технологией изготовления рото-

ра, содержащего постоянные магниты. Наилучшими удельными энергетическими показателями обладают ВД с постоянными магнитами, намагниченными радиально [1].

Во многих системах автоматики применяются безредукторные схемы электропривода, в которых используются двигатели с низкими номинальными частотами вращения. Естественные конструктивные ограничения на выполнение многополюсных синхронных двигателей объясняют появление комбинированных двигателей, у которых зубчатые пакеты на роторе чередуются с кольцевыми магнитами, намагниченными в аксиальном направлении, а статор имеет явно выраженные полюса, на которых выполнены мелкие зубцы [2]. Эти двигатели обладают повышенным коэффициентом электромагнитной редукции частоты вращения и приемлемыми энергетическими показателями, но при этом относительно большим остаточным моментом при обесточенных обмотках и наибольшей трудоемкостью изготовления из всех известных типов двигателей.

К классу ВД относятся и ВИРД, которые постепенно отвоевывают себе «место под солнцем», особенно в тех позициях, где наиболее ярко проявляются их достоинства и не играют существенной роли их недостатки. Достоинства ВИРД определяются простой технологичной конструкцией (прямые зубцы на статоре, катушечные обмотки, зубчатый безобмоточный ротор); отсутствием тепловых потерь в роторе; наличием коэффициента электромагнитной редукции; минимальным мо-

ментом инерции ротора; отсутствием «реакции якоря» и остаточного момента при обесточенных обмотках; высоконадежными схемами вентиляционных коммутаторов и низкой трудоемкостью изготовления. К недостаткам ВИРД следует отнести минимально возможный воздушный зазор, зависимость параметров двигателя от тока нагрузки и угла поворота ротора, повышенное энергопотребление и тепловыделение в пусковом режиме [3].

Сравнение характеристик двигателей, имеющих различные принципы создания момента, при разработке регулируемого электропривода представляет сложную многокритериальную задачу, приемлемое решение которой возможно только для конкретного привода. Каждый тип двигателя, очевидно, соответствует определенному классу регулируемых механизмов и условиям эксплуатации, для которых наиболее ярко проявляются его достоинства и «приглушаются» недостатки конкретного типа двигателя.

Многие известные результаты сравнений двигателей носят субъективный характер и отражают личные позиции авторов. Большинство из предлагаемых критериев сравнения оценивают показатели двигателей в некоторой расчетной точке, соответствующей либо пусковому режиму работы, либо номинальному [2]. Очень часто технические данные, приводимые в статьях, рекламных и справочных материалах, не позволяют объективно оценить возможности двигателей. При этом возникает проблема приведения сравниваемых двигателей к единому скалярному базису, в состав которого должны входить технические и эксплуатационные показатели, а также трудозатраты на изготовление двигателей в зависимости от объема выпуска. В качестве критерия сравнения двигателей, характеризующихся различными принципами действия и конструкциями, предлагается использовать естественные механические характеристики с учетом теплового режима работы.

Рассмотрим два класса самых технологичных электродвигателей: АД и ВИРД — и сравним их естественные механические характеристики.

Кроме технологичности конструкций, АД и ВИРД объединяет еще один признак — наличие одного источника электрической энергии; независимого потока возбуждения в двигателях — нет. В АД трехфазное напряжение питания создает вращающееся магнитное поле в воздушном зазоре, которое взаимодействует с током, протекающим в короткозамкнутой обмотке ротора под действием наведенной ЭДС. Вращающееся магнитное поле в ВИРД взаимодействует с ротором, имеющим явную магнитную асимметрию по расщепке статора по эквивалентным осям  $d$  и  $q$ . Следует заметить, что принцип формирования момента в АД обладает большей эффективностью по сравнению с принципом, реализованным в ВИРД. В АД момент на валу пропорционален электрическим потерям в роторе, поэтому температура нагрева обмоток ро-

тора и статора близка к максимально допустимой температуре изоляции обмотки статора. В ВИРД ротор практически холодный, тепловая нагрузка двигателя определяется электрическими потерями в обмотке статора. Для повышения энергетических показателей в АД и ВИРД воздушный зазор выбирается минимально возможным и ограничивается уровнем технологии и условиями эксплуатации.

В качестве исходного двигателя выберем трехфазный АД с короткозамкнутым ротором мощностью 13 кВт и синхронной частотой вращения  $150^{1/6}$ , расчет которого приведен в работе [4]. При сохранении внешних габаритных размеров АД спроектировано два ВИРД, различающихся числом фаз и соотношением зубцов статора / ротора. Параметры рассчитанных двигателей приведены в таблице. АД и ВИРД отличаются диаметром рашотки, числом пар полюсов, которое в ВИРД равно числу зубцов ротора. Напряжение питания обмоток АД — 220/380 В, 50 Гц, а напряжение питания обмоток ВИРД — 300 В постоянного тока. Кроме того, трехфазный ВИРД-300-6/4 рассчитан на режим одиночной коммутации фаз, когда каждая фаза проводит ток в течение 120 эл. град. В четырехфазном ВИРД 300-8/6 реализована парная коммутация, при которой ток проводят две фазы в течение 180 эл. град.

Рабочие характеристики ВИРД-300-6/4 и ВИРД-300-8/6, к которым относятся: полезная мощность на валу  $P_2 = f(M_{\text{нр}})$ , потребляемая мощность  $P_{\text{п}} = f(M_{\text{нр}})$ , электромагнитный КПД, учитывающий только электрические потери в обмотке:  $\eta = f(M_{\text{нр}})$  — представлены на рис. 1. Максимальная мощность на валу ВИРД имеет место при больших моментах нагрузки при условии сильного насыщения магнитной цепи двигателя. Максимальный КПД в ВИРД существует при малых моментах нагрузки и частотах вращения, близких к холостому ходу, когда эквивалентный, фиктивный поток возбуждения становится максимальным.

Механические характеристики ВИРД соответствуют характеристикам двигателя постоянного тока с последовательным возбуждением с ослабленным потоком. Величина фиктивного магнитного потока возбуждения в ВИРД лежит в диапазоне:  $\Phi_f = (0,25 - 0,5)\Phi_w$ , где  $\Phi_w$  — магнитный поток, созданный обмоткой. Режим работы приводного механизма определяет выбор расчетной точки на механической характеристике ВИРД, от значения которой зависит тепловой режим статора, степень насыщения магнитной системы двигателя и вид механической характеристики. Вид механической характеристики ВИРД можно сформировать за счет выбора способа соединения катушечных обмоток в фазу и режима коммутации фаз [5]. Типовые механические характеристики ВИРД в относительных единицах приведены на рис. 2. Режим парной коммутации фаз обеспечивает форсирование пускового момента по сравнению с одиночной коммутацией и

■ *Параметры двигателей АД-300, ВИРД-300-6/4 и ВИРД-300-8/6*

Параметр	Двигатель		
	АД-300	ВИРД-300-6/4	ВИРД 300-8/6
Внешний диаметр пакета статора, мм	291	291	291
Диаметр рашотки, мм	184	150	150
Длина пакета статора, мм	123	123	123
Воздушный зазор, мм	0,45	0,45	0,45
Число зубцов статора/ротора	36/46	6/4	8/6
Число фаз	3	3	4
Число пар полюсов	2	(4)*	(6)*
Шаг ВИРД, град	–	30	15
Длина лобовых частей обмотки, мм	218,5	115	87
Средняя длина витка, мм	683	477	419
Сопротивление фазной обмотки, Ом	0,36	3,19	5,46
Режим коммутации фаз	–	Одиночная	Парная
Напряжение питания, В	220/380, 50 Гц	= 300	= 300
Пусковой / максимальный момент, Н·м	125 / 202	368	334
Плотность тока при пуске, А/мм <sup>2</sup>	–	60	60
Потребляемая мощность при пуске, Вт	–	28218	32968
Установившаяся температура в режиме короткого замыкания, °С	–	340	446
Параметр повторно-кратковременного режима, %, при $T_{уст}=150$ °С	–	44	34
Номинальный момент, Н·м	<b>87</b>	<b>87</b>	<b>87</b>
Частота вращения номинальная, 1/с	146	32,5	36
Номинальная мощность на валу, Вт	12702	2827,5	3132
Потребляемая мощность в номинальном режиме, Вт	14272	3213	3559
КПД в номинальном режиме	0,89	0,88	0,88
Средняя полезная мощность, Вт	21780	11670	11970

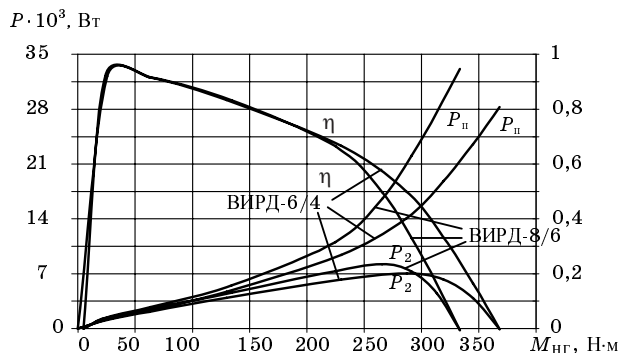
может быть реализован в двигателях, имеющих число фаз больше трех.

Различаем три возможных режима работы приводных механизмов, для которых выбираются следующие расчетные точки:

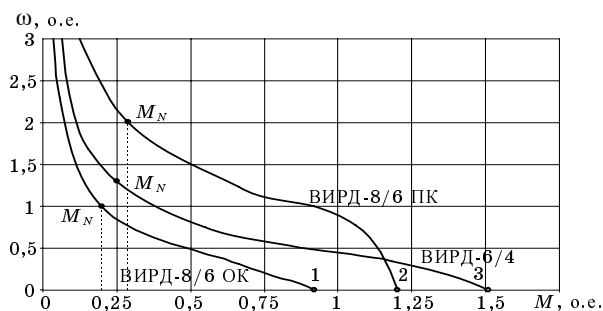
— двигатели, предназначенные для длительной работы в заторможенном режиме, рассчитываются для долговременной работы в пусковом режиме, максимальное значение пускового момента определяет-

ся допустимой температурой изоляции обмотки в установившемся режиме работы, максимальное значение индукции, близкое к индукции насыщения стали, имеет место в коронке зубца статора, КПД двигателя — низкий, механическая характеристика — «мягкая» (кривая 1 на рис. 2);

— двигатели, предназначенные для привода тяговых механизмов, должны создавать форсированный момент в повторно-кратковременном режиме

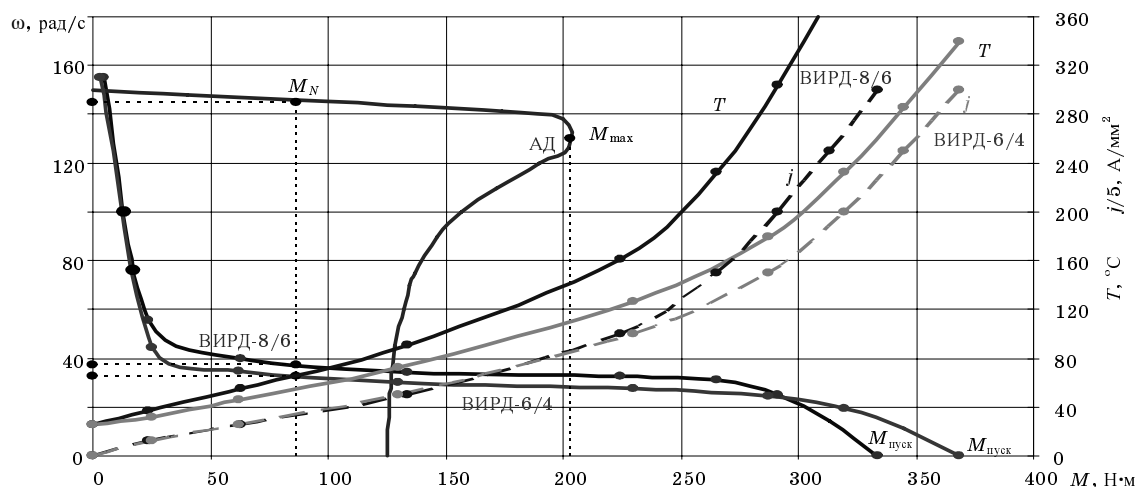


■ *Рис. 1. Рабочие характеристики ВИРД-300-6/4 и ВИРД-300-8*



■ *Рис. 2. Типовые механические характеристики ВИРД*





■ Рис. 3. Механические характеристики двигателей АД-300 и ВИРД-300-6/4, ВИРД-300-8/6

работы, величина которого определяется допустимой температурой изоляции обмотки в форсированном режиме, при этом зубцы статора полностью насыщены, КПД двигателя — средний; механическая характеристика — «жесткая» в зоне низких частот вращения и больших моментов нагрузки (кривая 2 на рис. 2);

— двигатели, предназначенные для длительной работы в фиксированном диапазоне частот вращения; рассчитываются на номинальный момент, равный  $M_N = (0,2 \div 0,3)M_{\text{пуск}}$ , прямой пуск данных двигателей осуществляется при ограничении потребляемого тока; в режиме пуска зубцы и спинка статора сильно насыщены; КПД двигателей — высокий; механическая характеристика является «жесткой» в требуемом диапазоне частот вращения (кривая 3 на рис. 2).

Механические характеристики АД-300 и ВИРД-300-6/4, ВИРД-300-8/6, а также зависимости плотности тока и температуры обмотки ВИРД от момента нагрузки  $J = f(M_{\text{нр}})$ ,  $T = f(M_{\text{нр}})$  показаны на рис. 3. Сравнение приведенных характеристик АД ( $2p = 2$ ) и ВИРД, выполненных в одних габаритах, показывает, что ВИРД может быть конкурентоспособным с АД, имеющим повышенное число пар полюсов ( $2p = 4$  и более), из-за естественного коэффициента электромагнитной редукции частоты вращения, характерного для ВИРД. В АД при  $2p = 8$  синхронная частота вращения  $\omega_c = 37,5 \text{ 1/с}$ , что соответствует основному диапазону частот спроектированных ВИРД (см. рис. 3).

Среднее значение полезной мощности на валу двигателей

$$P_{2\text{ср}} = \int_0^{\omega_{\text{х.х}}} M_{\text{нр}} d\omega$$

может быть использовано в качестве критерия сравнения их механических характеристик. Вычисленные значения полезной средней мощности, приведенные в таблице, подтверждают сказанное о том, что при снижении синхронной частоты вращения в АД в два раза среднее значение полезной мощности будет соизмеримо полезной средней мощности на валу ВИРД.

### Литература

1. Жуков В. П., Нестерин В. А. Высокомомментные вентильные электродвигатели серии 5ДВМ // Электротехника. 2000. № 6. С. 19–21.
2. Демагин А. В. Электрические машины для непосредственного привода приборных систем / ЦНИИ «Румб». Л., 1991. 80 с.
3. Голландцев Ю. А. Вентильные индукторно-реактивные двигатели / ГНЦ РФ — ЦНИИ «Электроприбор». СПб., 2003. 150 с.
4. Сергеев П. С., Виноградов Н. В., Горяинов Ф. А. Проектирование электрических машин. М.: Госэнергоиздат, 1969. 632 с.
5. Голландцев Ю. А. Механические характеристики вентильных индукторно-реактивных двигателей // Изв. вузов. Сер. Приборостроение. 2005. Т. 49. № 7. С. 51–55.

## ЮСУПОВ РАФАЭЛЬ МИДХАТОВИЧ. ТВОРЧЕСКАЯ БИОГРАФИЯ

25 мая 2006 года на общем собрании Российской академии наук (РАН) в Москве членом-корреспондентом РАН был избран член редакционного совета нашего журнала Юсупов Рафаэль Мидхатович.

Родился Рафаэль Мидхатович Юсупов 17 июля 1934 года в г. Казани. Детство мальчика было нелегким — он рано остался без отца. Его отец, Мидхат Мустафеевич, был человеком высоких нравственных правил. Несмотря на наличие брони (освобождения от призыва в армию), он в первый же день Великой Отечественной войны ушел добровольцем на фронт. В январе 1942 года М. М. Юсупов погиб под Москвой. Мать осталась одна с двумя детьми на руках — с восьмилетним сыном Рафаэлем и трехлетней дочерью Эльмирой.

После окончания в 1952 году с золотой медалью Казанской спецшколы ВВС Рафаэль Юсупов был направлен в Ленинградскую военно-воздушную инженерную академию (ныне Военно-космическая академия (ВКА) им. А. Ф. Можайского), которую окончил с отличием в 1958 году по специальности инженер-электрик. В 1964 году он окончил Ленинградский государственный университет по специальности математик.

В 1958–1985 годах Рафаэль Мидхатович проходил службу в Военной академии им. А. Ф. Можайского, занимая должности от инженера до начальника созданного с его участием уникального в системе высшего военного образования факультета сбора и обработки информации (1976–1985), а в 1985 году назначен на должность начальника направления моделирования стратегических операций Центра оперативно-стратегических исследований Генерального штаба ВС СССР; через год переведен на должность начальника кафедры оперативно-тактической подготовки Военной академии им. А. Ф. Можайского.

В 1989 году Р. М. Юсупова, оставив в кадрах Вооруженных Сил, назначают заместителем директора по научной работе Санкт-Петербургского института информатики и автоматизации Российской академии наук (СПИИРАН), а в 1991 году избирают директором этого института, каковым он и является по настоящее время.

В 1958–1976 годах Рафаэль Мидхатович проводил исследования в области теории управления полетом летательных аппаратов, теории самонастраивающихся (адаптивных) систем, теории идентификации и теории чувствительности динамических систем. Он является одним из основоположников отечественной научной школы по теории чувствительности, которая вошла в число основных разделов общего курса теории автоматического управления. Результаты научных исследова-

ний в данной области используются при проектировании, настройке и испытаниях высокоточных приборов и систем. Анализ чувствительности стал обязательным этапом создания систем управления и их элементов.

Существенное влияние на становление Р. М. Юсупова как ученого в этот период оказал выдающийся ученый в области систем управления академик Евгений Павлович Попов. В то время член-корреспондент АН СССР Е. П. Попов был начальником кафедры систем управления полетом ракет и космических аппаратов академии, а Р. М. Юсупов — старшим научным сотрудником этой кафедры. Именно по совету и при поддержке Попова Р. М. Юсупов стал заниматься теорией и практикой самонастраивающихся систем управления (по этой тематике им в 1963 году была успешно защищена кандидатская диссертация и подготовлен совместно с Е. П. Поповым доклад для Второго конгресса ИФАК — Международной федерации по автоматическому управлению, Швейцария), а затем и теорией чувствительности для анализа и синтеза систем управления ракет и космических аппаратов, по которой он защитил в 1968 году докторскую диссертацию.

С 1976 года научные интересы Р. М. Юсупова связаны с проблемами сбора и обработки информации, геофизической кибернетики (теории управления геофизическими процессами), математического моделирования, информатизации, телемедицины, конфликтологии, информационной безопасности.

В области теории моделирования Рафаэль Мидхатович развивает новое научное направление — методы оценивания качества моделей (адекватность, чувствительность, сложность и т. д.), названное им «квалиметрией моделей». Им сформулированы концептуальные основы квалиметрии моделей, разработаны методы и алгоритмы оценивания адекватности и чувствительности моделей.

Значительные результаты, полученные им при разработке концептуальных и методологических основ информатизации и информационного общества, оказали существенное влияние на эти процессы в Санкт-Петербурге и стране. Р. М. Юсупов является одним из инициаторов и организаторов работ в области информатизации Санкт-Петербурга. Им разработана универсальная структура концепции информатизации, с использованием которой под его руководством и при непосредственном участии созданы концепции информатизации Ленинградского экономического региона (1991) и типовая концепция информатизации городского района (1992). Р. М. Юсупов является соавтором концепции информатизации Санкт-Петербурга и

стратегии перехода Санкт-Петербурга в информационное общество; руководителем и разработчиком ряда проектов для города в области телекоммуникаций и телемедицины, в том числе целевой программы «Электронный Санкт-Петербург». Р. М. Юсупов — бессменный руководитель проводимых в городе при поддержке администрации Санкт-Петербурга регулярных конференций «Региональная информатика» и «Информационная безопасность регионов России», которые способствуют формированию единого информационного пространства города. Им создан постоянно действующий семинар по разработке компьютерных моделей развития города.

В более ранних работах Рафаэлем Мидхатовичем были разработаны модели подготовки научных кадров, позднее — структурная и экономикоматематическая модели информационного общества в предположении, что в последнем функционируют секторы материального и информационного (знания) производства.

С использованием этих моделей им получена новая параметрическая информационная модель развития науки, позволяющая исследовать влияние ряда параметров (объем финансирования, численность научных работников, старение знаний, «утечка умов» и технологий и т. д.) на эволюцию науки.

Под его руководством и с его участием в ВКА им. А. Ф. Можайского и в СПИИРАН выполнено, в том числе по постановлениям правительства и федеральным целевым научно-техническим программам (НТП), более 90 научно-исследовательских работ и проектов по вопросам повышения точности, эффективности, испытаний военно-технических систем (ВТС), планирования их развития, создания математического обеспечения систем обработки информации в ВТС, разработки информационно-расчетных систем оценки состояния природной среды и ее влияния на применение ВТС, создания новых информационных технологий и их использования в различных сферах развития общества.

Благодаря усилиям Р. М. Юсупова в области научно-организаторской деятельности на посту директора института, в сложных условиях социально-экономических реформ удалось не только сохранить и увеличить интеллектуальный потенциал СПИИРАН, но и закрепить за институтом роль одного из ведущих научных учреждений в области информатики и автоматизации в Российской академии наук и в Санкт-Петербурге.

Научную деятельность Р. М. Юсупов успешно сочетает с работой по подготовке высококвалифицированных инженерных и научных кадров. В течение 20 лет — в ВКА им. А. Ф. Можайского, с 1991 года — на базовой кафедре автоматизации научных исследований Санкт-Петербургского государственного электротехнического университета, а с 2003 года — и на базовой кафедре приклад-

ной информатики Санкт-Петербургского государственного университета аэрокосмического приборостроения, являясь их заведующим.

В 1996–1999 годах Р. М. Юсупов был профессором Санкт-Петербургского государственного политехнического университета, где при его непосредственном участии в 1998 году создан факультет «Безопасность», деканом которого он работал с 1998 по 2001 год. Им подготовлено 10 докторов и 45 кандидатов наук. Р. М. Юсупов — автор более 350 научных трудов, в том числе 16 монографий и 17 изобретений.

Рафаэль Мидхатович ведет большую научно-организаторскую и общественную работу в качестве председателя Объединенного научного совета этого центра по информатике, телекоммуникациям и управлению (с 1992), председателя научно-технического совета по региональным проблемам национальной безопасности (1996–1998); сопредседателя координационного совета Партнерства для развития информационного общества на Северо-Западе России (ПРИОР СЗ) (2002); заместителя председателя научного совета по информатизации Санкт-Петербурга (с 1994); члена президиума Санкт-Петербургского научного центра РАН (с 1992), научного совета при Совете Безопасности РФ (с 1999), совета РАН «Научные телекоммуникации и информационная инфраструктура» (1998–2003), совета РАН «Высокопроизводительные вычислительные системы, научные телекоммуникации и информационная инфраструктура» (с 2003), научного совета при правительстве Санкт-Петербурга (1995–1998), научных советов по государственным НТП «Перспективные информационные технологии» (1994–1998) и «Информатизация России» (с 1994), Северо-Западной секции содействия развитию экономической науки РАН (с 1998), Российского национального комитета по автоматическому управлению (с 1998), комиссии при губернаторе Санкт-Петербурга по реформированию научно-технической сферы (1998–1999), правления Санкт-Петербургского отделения Ломоносовского фонда (с 2004), Общественного совета Санкт-Петербурга (с 2002).

Р. М. Юсупов входит в состав редакционных советов журналов «Вооружение, политика, конверсия» (с 1993), «Информатизация и связь» (с 1993), «Journal of Intelligent Control» (США, с 1995), «Телекоммуникации» (с 2000), «Информационно-управляющие системы» (с 2002), «Мехатроника, автоматизация и управление» (с 2003), он председатель докторского диссертационного совета при СПИИРАН, член докторского диссертационного совета при ВКА им. А. Ф. Можайского.

По его инициативе в здании СПИИРАН в 1995 году создан музей всемирно известной гимназии Карла Мая, пропагандирующий лучшие педагогические и культурно-нравственные традиции российской школы.

Р. М. Юсупов избран действительным членом Академии военных наук (1996), Международной академии навигации и управления (1995), Международной академии наук экологии, безопасности человека и природы (1996), Санкт-Петербургской инженерной академии (1992), Российской академии естественных наук (1999), которая присвоила ему за научные, педагогические и общественные заслуги почетное звание «Рыцарь науки и искусств» и наградила одноименным знаком (2004).

В период военной службы Р. М. Юсупов награжден орденом Красной Звезды (1978) и тринадцатью медалями, ему присвоено воинское звание генерал-майор (1980). За научные достижения и педагогические заслуги ему присвоено звание «Почетный радист СССР» (1974), «Заслуженный дея-

тель науки и техники РФ» (1984), он награжден Министерством высшего и среднего специального образования СССР Первой премией и медалью за лучшую научную работу (1983), орденом Почета (1999), а в мае 2005 года — орденом «За заслуги перед Отечеством» IV степени (см. фото на 4-й стороне обложки).

Естественным и закономерным результатом многолетней плодотворной деятельности в науке Р. М. Юсупова стало избрание его в мае 2006 года членом-корреспондентом РАН. Редакция журнала от души поздравляет Рафаэля Мидхатовича и желает дальнейших творческих успехов.

*Редакция журнала  
«Информационно-управляющие системы»*

---

---

**АВТОМАТИЗАЦИЯ. ЭЛЕКТРОНИКА-2007**  
**10-Я МЕЖДУНАРОДНАЯ СПЕЦИАЛИЗИРОВАННАЯ ВЫСТАВКА**  
**20–23 марта 2007 г.**

Место проведения: Минск, пр. Победителей, 14, выставочный павильон «Грин Экспо»

**Организаторы**

Выставочный центр «Грин Экспо»  
Выставочная компания «Минскэкспо»  
Министерство промышленности Республики Беларусь  
Национальная академия наук Беларуси

**Направления работы выставки**

Технические средства, системы автоматического контроля и управления  
Контрольно-измерительные приборы  
Компьютерная техника и информационные технологии  
Информационные технологии и программное обеспечение: системы CAD/CAM, базовые системы и средства разработки программ, инжиниринг, услуги и сервис  
Автоматизированные системы и технические средства управления производством и технологическими процессами  
Промышленная автоматизация. Роботы. Манипуляторы. Периферийное оборудование  
Промышленные контроллеры  
Лазерная техника

Обработка изображений в промышленном процессе

Полупроводниковые устройства  
Электромеханические компоненты и технологии соединений  
Встроенные системы  
Датчики и микросистемы  
Оборудование и программное обеспечение — тестирование измерений  
Микросистемная технология  
Электротехника  
Обработка материалов  
Производство компонентов, технологии для обработки кабелей  
Технологии производства печатных плат и других носителей схем  
Технология монтажа компонентов на поверхность плат  
Технология пайки  
Чистовая обработка изделий  
Испытания и измерения

**Дополнительная информация**

<http://www.exponet.ru/exhibitions/by-id/automaticmi/automaticmi2007/index.ru.html>



**БЕЗЗАТЕЕВ**  
**Сергей**  
**Валентинович**



Доцент Санкт-Петербургского государственного университета аэрокосмического приборостроения.

В 1980 году окончил Ленинградский институт авиационного приборостроения.

В 1987 году защитил диссертацию на соискание ученой степени кандидата технических наук.

Является автором более 40 научных публикаций.

Область научных интересов — теория информации, теория кодирования, системы информационной безопасности.

**ГОЛЛАНДЦЕВ**  
**Юрий**  
**Алексеевич**



Ведущий научный сотрудник ЦНИИ «Электроприбор», действительный член Международной энергетической академии, член Международного общества приборостроителей ISA. В 1967 году окончил Ленинградский институт авиационного приборостроения.

В 2005 году защитил диссертацию на соискание ученой степени доктора технических наук. Является автором 95 научных публикаций.

Область научных интересов — электромеханика и робототехника.

**КАНЖЕЛЕВ**  
**Сергей**  
**Юрьевич**



Магистрант Санкт-Петербургского государственного университета информационных технологий, механики и оптики.

В 2005 году защитил диплом бакалавра.

Имеет одну научную публикацию.

Область научных интересов — объектно-ориентированное проектирование, автоматное программирование и графические модели программ.

**КАРПОВ**  
**Алексей**  
**Анатольевич**



Аспирант Санкт-Петербургского института информатики и автоматизации РАН, лауреат диплома 2-й степени за устный доклад на 7-й конференции молодых ученых «Навигация и управление движением».

В 2001 году окончил с отличием Санкт-Петербургский государственный университет аэрокосмического приборостроения.

Является автором 30 научных публикаций.

Область научных интересов — многомодальные интерфейсы.

**ЛОБАНОВ**  
**Борис**  
**Мефодьевич**



Заведующий лабораторией распознавания и синтеза речи Объединенного института проблем информатики НАН Беларуси.

В 1965 году окончил Минский радиотехнический институт.

В 1983 году защитил диссертацию на соискание ученой степени доктора технических наук. Является автором 211 научных публикаций.

Область научных интересов — речевые технологии.

**Нестеренко**  
**Владимир**  
**Дмитриевич**



Начальник отдела новых технологий Генеральной дирекции ОАО «Северо-Западный Телеком».

В 1976 году окончил Ленинградский электротехнический институт связи им. проф. М. А. Бонч-Бруевича.

В 1989 году защитил диссертацию на соискание ученой степени кандидата технических наук по специальности «Сети, узлы связи и распределения информации».

Является автором более 50 научных публикаций.

Область научных интересов — разработка концепций и проектов сетей передачи данных, сети следующего поколения (NGN), централизованные системы управления сети NGN.

**РОНЖИН  
Андрей  
Леонидович**



Старший научный сотрудник, руководитель группы речевой информатики, председатель совета молодых ученых Санкт-Петербургского института информатики и автоматизации РАН, доцент кафедры робототехники и нейроинформатики Санкт-Петербургского государственного университета аэрокосмического приборостроения.

В 1999 году окончил Санкт-Петербургский государственный университет аэрокосмического приборостроения.

В 2003 году защитил диссертацию на соискание ученой степени кандидата технических наук.

Является автором 35 научных публикаций.

Область научных интересов — многомодальные интерфейсы.

**РЫЖИКОВ  
Юрий  
Иванович**



Профессор кафедры математического обеспечения ЭВМ Военно-космической академии им. А. Ф. Можайского. Заслуженный деятель науки РФ.

В 1958 году окончил Черноморское высшее военно-морское училище им. П. С. Нахимова. В 1969 году защитил диссертацию на соискание ученой степени доктора технических наук.

Является автором более 200 научных публикаций, в том числе 40 учебных пособий и монографий.

Область научных интересов — теория очередей, имитационное моделирование, вычислительные методы и прикладное программирование, управление запасами, науковедение и педагогика высшей школы.

**ТРОЯНОВСКИЙ  
Борис  
Константинович**



Доцент кафедры безопасности информационных систем Санкт-Петербургского Государственного университета аэрокосмического приборостроения.

В 1976 году окончил Ленинградский институт авиационного приборостроения.

Является автором более 30 научных публикаций.

Область научных интересов — алгебраическое кодирование, компрессия речевых сигналов, сжатие аудио, телефония с мутацией пакетов.

**ЦИРУЛЬНИК  
Лилия  
Исааковна**



Научный сотрудник лаборатории распознавания и синтеза речи Объединенного института проблем информатики НАН Беларуси.

В 1995 году окончила Белорусский государственный университет.

Является автором 22 научных публикаций.

Область научных интересов — персонализированный синтез речи по тексту, компьютерное клонирование голоса и дикции личности.

**ШАЛЫТО  
Анатолий  
Абрамович**



Заведующий кафедрой технологий программирования Санкт-Петербургского государственного университета информационных технологий, механики и оптики. Ученый секретарь НПО «Аврора».

В 1971 году окончил Ленинградский электротехнический институт им. В. И. Ульянова (Ленина) по специальности «Автоматика и телемеханика».

В 1999 году защитил диссертацию на соискание ученой степени доктора технических наук. Является автором более 250 научных публикаций, трех монографий и 70 изобретений.

Область научных интересов — системы логического управления, автоматное программирование.

**ШОЛОХОВ  
Алексей  
Викторович**



Доцент Серпуховского военного института ракетных войск, докторант.

В 1994 году окончил Серпуховское высшее военное командно-инженерное училище.

В 1999 году защитил диссертацию на соискание ученой степени кандидата технических наук.

Является автором 40 научных публикаций.

Область научных интересов — наземные навигационные системы, интегрированные с геоинформационными системами.

УДК 681.3

Выбор алгоритма преобразования, обеспечивающего изменение структуры изображения

*Беззатеев С. В., Литвинов М. Ю., Трояновский Б. К., Филатов Г. П.* Информационно-управляющие системы, 2006. № 6. С. 2–6.

Рассматривается проблема выбора эффективного алгоритма преобразования видеoinформации в видеосистемах встраиваемого класса, обеспечивающего ее конфиденциальность при передаче и хранении. Предлагается модификация алгоритма VEA, позволяющая повысить защищенность информации в условиях атак по перехваченной преобразованной информации и выбранному исходному изображению и обеспечивающая эффективное уничтожение структуры передаваемой видеoinформации.

Список лит.: 15 назв.

УДК 621.391

Метод оценки достоверности информации при периодической коррекции наземных навигационных систем

*Шолохов А. В.* Информационно-управляющие системы, 2006. № 6. С. 7–14.

Разработан способ определения достоверности информации навигационных систем, корректируемых эпизодически с привлечением различных внешних источников, на основе методов теории марковских случайных процессов. Рассмотрены отличительные особенности задачи, приводящие к значительным погрешностям при оценке достоверности известными методами, показаны пути учета рекуррентного характера событий и непосредственности интенсивностей их потоков.

Список лит.: 13 назв.

УДК 681.327.8

Концепция построения архитектуры моделей процессов управления инфокоммуникационными сетями (Часть 1)

*Нестеренко В. Д.* Информационно-управляющие системы, 2006. № 6. С. 15–23.

Предложена архитектура инфокоммуникационной сети как объекта управления, основанная на модели мультиисреды, охватывающей среды взаимодействия, генерации, распространения инфокоммуникационных сетей и позволяющая провести структуризацию моделей и алгоритмов их адаптивного управления.

Список лит.: 3 назв.

UDK 681.3

The choice of the transformation algorithm that ensures a structural change of videoinformation

*Bezzateev S. V., Litvinov M. Yu., Troyanovskii B. K., Filatov G. P.* IUS, 2006. N 6. P. 2–6.

The authors study the choice of an effective algorithm for the transformation of videoinformation in embedded video systems that ensures its confidentiality on keeping and transmission. A modification of the VEA algorithm is proposed. This modification increases the protection level of the information against the interception attacks and ensures an efficient destruction of the transmitted information.

Pefs: 15 titles.

UDK 621.391

A method of estimation of the information reliability for periodically corrected navigation systems

*Sholokhov A. V.* IUS, 2006. N 6. P. 7–14.

A method to determine the information reliability of navigating systems with periodical correction is developed. The study is based on the theory of Markov stochastic processes. We emphasize some special features of this problem which result in significant errors in the estimation of reliability by known methods, and show some ways of taking into account the recurrent character of events and the inconstancy of intensity of their streams.

Pefs: 13 titles.

UDK 681.327.8

On the architecture of control processes for infocommunication networks (Part 1)

*Nesterenko V. D.* IUS, 2006. N 6. P. 15–23.

The paper proposes an architecture for the infocommunication network as an object of control which is based on a model of composed media including the media of interaction, generation and interaction of infocommunication systems that allows the structurisation of both the models and the algorithms of their adaptive control.

Pefs: 3 titles.

УДК 004.93

Фонетико-морфологическая разметка речевых корпусов для распознавания и синтеза русской речи

*Ронжин А. Л., Карпов А. А., Лобанов Б. М., Цирульник Л. И., Йокиш О.* Информационно-управляющие системы, 2006. № 6. С. 24–34.

Показаны особенности подготовки и обработки текстовых данных для разметки речевых корпусов, обсуждается процедура анализа и сопоставления речевых сигналов в процессе разметки, описаны процедуры транскрибирования русских текстов, морфемного представления языка и речи. Описываются автоматизированные системы создания фонетико-акустических речевых баз данных для персонализированного синтеза речи по тексту и распознавания русской речи SIRIUS.

Список лит.: 11 назв.

УДК 004.93

УДК 681.3.06

Автоматическая генерация автоматного кода  
*Канжелев С. Ю., Шалыто А. А.* Информационно-управляющие системы, 2006. № 6. С. 35–42.

Рассматривается подход к автоматической генерации кода автоматных программ на любом априори заданном языке программирования по графам переходов автоматов.

Список лит.: 15 назв.

УДК 519.872.3

Средние времена ожидания и пребывания в многоканальных приоритетных системах

*Рыжиков Ю. И.* Информационно-управляющие системы, 2006. № 6. С. 43–49.

Предлагается способ расчета средних времен ожидания и пребывания заявок в  $n$ -канальной системе с относительным, абсолютным и смешанным приоритетами при произвольных и различных по типам заявок распределениях длительностей чистого обслуживания. Точность расчетов иллюстрируется сопоставлением с результатами имитационного моделирования и пересчетов на основе «инвариантов отношения». Обсуждается расчет «замкнутых» приоритетных систем.

Список лит.: 25 назв.

Phonetic-morphological mapping of speech corpora for recognition and synthesis of Russian speech

*Ronzhin A. L., Karpov A. A., Lobanov B. M., Tsirulnik L. I., Jokisch O.* IUS, 2006. N 6. P. 24–34.

The particularities of preparation and processing of text data for speech corpora labeling are presented. The problem of analysis and comparison of speech signals during the labeling is discussed. The procedures of Russian text transcription and morphemic description of language and speech are presented. SIRIUS, an automated system for phonetic-acoustical voice database creation for personalized speech synthesis and Russian speech recognition, is described.

Pefs: 11 titles.

УДК 681.3.06

Automatic generation of automaton code

*Kanzhelev S. Y., Shalyto A. A.* IUS, 2006. N 6. P. 35–42.

An approach to the automatic automata's program code generation is considered in the paper. The code is generated in any programming language by the automaton's transition graph.

Pefs: 15 titles.

УДК 519.872.3

Mean waiting and sojourn times in multichannel priority systems

*Ryzhikov Yu. I.* IUS, 2006. N 6. P. 43–49.

A method is proposed to compute mean waiting and sojourn times for  $n$ -channel system with preemptive, non-preemptive and mixed priorities. Service time distributions are supposed to be distinct and arbitrary. The precision of the algorithm is demonstrated by a comparison with the results of imitation and calculations on the basis of «relation invariants». Closed priority systems are discussed.

Pefs: 25 titles.



УДК 621.313.13

Сравнение механических характеристик асинхронных и вентильных индукторно-реактивных двигателей

*Голландцев Ю. А.* Информационно-управляющие системы, 2006. № 6. С. 50–53.

Сравниваются механические характеристики асинхронных двигателей с короткозамкнутым ротором и вентильных индукторно-реактивных двигателей, предназначенных для применения в системах автоматического управления. Обсуждаются результаты сравнения характеристик и даются рекомендации по выбору двигателей.

Список лит.: 5 назв.

UDK 621.313.13

Comparison of mechanical characteristics of asynchronous and switched reluctance motors

*Gollandtsev Yu. A.* IUS, 2006. N 6. P. 50–53.

The mechanical features of asynchronous motors are compared with short-circuit rotor and switched reluctance motors, intended for use in automatic control systems. We discuss the results of the comparison and give recommendations on the choice of the motors for specific purposes.

Pefs: 5 titles.

**АСУ ЭКСПО-2007**  
**МЕЖДУНАРОДНАЯ СПЕЦИАЛИЗИРОВАННАЯ ВЫСТАВКА**  
**13–16 марта 2007 г.**

Место проведения: Санкт-Петербург, Большой проспект В. О., д. 103, Ленэкспо, павильон 5

**Организатор**

Выставочное объединение «РЭСТЭК»

**Выставка проводится при поддержке**

Представительства Федерального Агентства по промышленности в Северо-Западном федеральном округе  
Санкт-Петербургского института информатики и автоматизации РАН

**Основные тематические разделы***Промышленная автоматизация*

Комплексная автоматизация производственных процессов  
MES-системы  
Услуги по системной интеграции в области АСУ  
ПП  
Системы безопасности управления производством  
Диспетчеризация и сбор данных  
SCADA-системы  
Управление техпроцессом в режиме реального времени  
НМІ промышленные сети  
ПЛК-системы  
Промышленные компьютеры

**Контроллеры**

Датчики, приводы, исполнительные устройства, модули ввода-вывода  
Конструктивы, шкафы, кабели, клеммы, установочные изделия  
Взрывозащищенное электрооборудование  
Испытательные системы и тренажеры для САУ

*Бортовые и встраиваемые системы*

Компьютеры ответственных применений  
Бортовые вычислительные  
Одноплатные компьютеры для мультимедийных игровых автоматов  
Панельные компьютеры для систем НМ  
Электронные компоненты и модули для жестких условий эксплуатации  
Защищенные вычислительные комплексы: ноутбуки, планшетные, карманные компьютеры  
Системы связи и навигации для мобильных и ответственных применений  
Заказные разработки и контрактная сборка электронных модулей  
Встраиваемые операционные системы, системы реального времени

**Дополнительная информация**

<http://www.exponet.ru/exhibitions/by-id/acsexpo/acsexpo2007/index.ru.html>

# СОДЕРЖАНИЕ ЖУРНАЛА «ИНФОРМАЦИОННО-УПРАВЛЯЮЩИЕ СИСТЕМЫ» за 2006 г. [№ 1–6]

Ададунов С. Е., Богданова А. Ф. Управление производительностью спутниковых информационных сетей .....	5	43
Ададунов С. Е., Мальцев Г. Н., Моторин Н. М., Ададунов А. С. Диффузионная аппроксимация процессов информационного обмена в низкоорбитальных спутниковых радиосетях .	4	40
Астратов О. С., Кузьмин С. А. Обнаружение транспортных средств и оценка параметров транспортных потоков по видеопоследовательности .....	3	19
Афанасьева В. В., Кубайчук А. Б., Шаповалов В. В. Построение решающих правил для систем автоматизированного скрининга .....	1	2
Балонин Н. А., Мироновский Л. А. Матрицы Адамара нечетного порядка .....	3	46
Бегун П. И., Кривохижина О. В., Сухов В. К. Компьютерное моделирование и биомеханический анализ критического состояния и коррекции структур сосудистой системы (Часть 2) .....	1	28
Бегун П. И., Лазарев С. М., Лебедева Е. А. Информационное обеспечение исследований и коррекций в герниологии .....	2	47
Беззатеев С. В., Литвинов М. Ю., Трояновский Б. К., Филатов Г. П. Выбор алгоритма преобразования, обеспечивающего изменение структуры изображения .....	6	3
Бурлуцкий С. Г. 60 лет факультету радиоэлектроники ВКА им. А. Ф. Можайского .....	5	55
Василевский А. М., Коноплев Г. А. Оптико-электронная информационно-измерительная система спектрального анализа состава поликомпонентных сред по доминирующей компоненте .....	1	48
Вилесов Л. Д. Фильтрация изображения при наличии фотонных шумов .....	3	37
Воронков К. Л., Шерстюк Ю. М., Романенко А. И. Методика синтеза структуры витрины данных медико-генетического мониторинга .....	1	14
Голландцев Ю. А. Сравнение механических характеристик асинхронных и вентильных индукторно-реактивных двигателей .....	6	50
Гордеев А. В. Виртуальные машины и сети .....	2	21
Гришин В. В., Богачик Ю. Е. Модели информационного взаимодействия наземной радиоэлектронной системы и космического аппарата .....	5	11
Иванов В. Н., Тихонов Э. П. Исследование алгоритмов аналого-цифрового преобразования при воздействии аддитивной помехи .....	4	17
Игнатьев М. Б., Волкова В. Н. 50 лет секции кибернетики Дома ученых РАН им. М. Горького .	4	52
Канжелев С. Ю., Шалыто А. А. Автоматическая генерация автоматного кода .....	6	35
Ковалев Д. В. Модель медико-социальной экспертизы ребенка-инвалида .....	1	7
Колесов Н. В., Толмачева М. В. Субоптимальный алгоритм построения расписаний для иерархических вычислительных систем .....	2	14
Корниенко А. А., Штанько С. В. Криптографический протокол защиты информации в радиоканалах сетевых спутниковых систем с использованием асимметричных алгоритмов ...	5	21
Кунгурцев В. В. Технологии построения бортовых информационно-управляющих комплексов космических аппаратов на базе интегральных схем с программируемой логикой .....	5	17
Мальцев Г. Н., Стогов Г. В., Терехов А. В. Перспективы создания комплексов управления космическими аппаратами на базе ключевых технологий .....	5	2
Мальцев Г. Н., Травкин В. С. Оптимальный прием сложных фазоманипулированных сигналов в спутниковых радиоканалах в условиях внутрисистемных структурных помех ..	5	36
Манило Л. А., Зозуля Е. П. Автоматическое распознавание мерцательной аритмии с использованием оценок аппроксимированной энтропии .....	1	21
Марковский С. Г., Тюрликов А. М. Использование адресов абонентов для разрешения конфликтов в канале с шумом .....	2	27
Машевский Г. А., Тарасов В. А. Развитие информационной базы при ионометрировании больных с распространенными формами рака .....	4	50
Мендельсон А. М., Бендерская Е. Н., Тенно Р. А. Параметрическая идентификация электрохимического процесса на основе генетических алгоритмов .....	2	53

	№	Стр.
<b>Моторин Н. М.</b> Исследование вероятностно-временных характеристик передачи сообщений в низкоорбитальной спутниковой системе связи в режиме «электронная почта» .....	5	27
<b>Нестеренко В. Д.</b> Концепция построения архитектуры моделей процессов управления инфокоммуникационными сетями (Часть I) .....	6	15
<b>Обухова Н. А., Тимофеев Б. С.</b> Алгоритм сегментации объектов в последовательности видеокадров методами нечеткой логики .....	3	12
<b>Осипов Л. А., Воробьева Ю. Г.</b> Генерирование гауссовых марковских последовательностей .....	4	4
<b>Павлов В. С., Савинов М. В.</b> Условия оптимального измерения двух неэнергетических параметров с различной точностью .....	4	29
<b>Пахарьков Г. Н.</b> Проблемы оптимизации медико-технического оснащения отечественного здравоохранения .....	1	39
<b>Петелин Ю. В., Ковалев М. А., Макаров А. А.</b> Перспективы использования сигнально-кодовых конструкций типа троичных m-последовательностей в спутниковых каналах связи .....	5	32
<b>Петров П. Н., Сенин А. В.</b> Пространственно-временная обработка сигналов акустоэлектронными устройствами с масштабными переизлучающими решетками .....	3	5
<b>Розов А. К., Сырцев А. Н., Кузина Н. В.</b> Оптимальное правило остановки наблюдений – способ достижения наивысшей вероятности обнаружения сигналов .....	2	2
<b>Ронжин А. Л., Карпов А. А.</b> Многомодальная система для бесконтактной работы с персональным компьютером .....	4	33
<b>Ронжин А. Л., Карпов А. А., Лобанов Б. М., Цирульник Л. И., Йокиш О.</b> Фонетико-морфологическая разметка речевых корпусов для распознавания и синтеза русской речи .....	6	24
<b>Рыжиков Ю. И.</b> Полный расчет системы обслуживания с распределениями Кокса .....	2	38
<b>Рыжиков Ю. И.</b> Средние времена ожидания и пребывания в многоканальных приоритетных системах .....	6	43
<b>Сергеев А. М.</b> Об особенностях представления чисел при знакоразрядном кодировании и вычислительный эксперимент с ними .....	3	56
<b>Стогов Г. В., Богданова А. Ф.</b> Синтез моделей потоков сообщений для управления трафиком в спутниковых сетях .....	4	36
<b>Стручков И. В., Ицыксон В. М.</b> Формализм для описания программных систем и вычислительных процессов циклической параллельной обработки данных реального времени .....	2	8
<b>Тимофеев А. В., Чернакова С. Э., Нечаев А. И., Литвинов М. В.</b> Информационные технологии и мехатронные устройства для интеллектуальных медицинских систем .....	4	45
<b>Филатов В. Н., Смирнов В. М.</b> Формирование адаптивного строга в видеосистемах наблюдения .....	3	29
<b>Харченко А. В.</b> Применение ортогональных преобразований при цифровой обработке сигналов в спутниковых радиоканалах .....	5	6
<b>Цветков К. Ю., Родионов А. В., Акмолов А. Ф.</b> Анализ системы спутниковой связи с коммутируемым моноканалом .....	5	47
<b>Цветков С. А., Шишлаков В. Ф.</b> Синтез параметров линейных систем автоматического управления с амплитудно-импульсной модуляцией .....	4	10
<b>Чернышев К. Р.</b> Рецензия на книгу И. В. Прангишвили, Ф. Ф. Пащенко и Б. П. Бусыгина «Системные законы и закономерности в электродинамике, природе и обществе» .....	5	51
<b>Шишлаков В. Ф., Шишлаков Д. В.</b> Параметрический синтез многосвязных систем автоматического управления обобщенным методом Галеркина .....	3	51
<b>Шолохов А. В.</b> Метод оценки достоверности информации при периодической коррекции навигационных систем .....	6	7
<b>Юлдашев З. М., Мухаметшина В. А.</b> Автоматизированная система для анализа ионного состава биосубстратов .....	1	33
Аннотации .....	1	63
Аннотации .....	2	62
Аннотации .....	3	59
Аннотации .....	4	57
Аннотации .....	5	58
Сведения об авторах .....	1	60
Сведения об авторах .....	2	59
Сведения об авторах .....	3	62
Сведения об авторах .....	4	61
Сведения об авторах .....	5	62
20 лет Институту проблем проектирования в микроэлектронике Российской академии наук .....	4	2

	№	Стр.
III Международный семинар «Интеграция информации и геоинформационные системы» IF&GIS-2007 .....	5	6
V Международная конференция «Авиация и космонавтика-2006» .....	2	57
VII Международная научно-техническая конференция «Кибернетика и высокие технологии XXI века» (C&T-2006) .....	1	56
XI Международная конференция «Речь и компьютер» .....	2	58
XII Международная научно-техническая конференция «Радиолокация, навигация, связь» (RLNC-2006) .....	1	55
Всероссийская научно-техническая конференция «Проблемы разработки перспективных микроэлектронных систем-2006» (МЭС-2006) .....	1	57
Всероссийская школа молодых ученых «Проблемы и перспективы разработки и практического применения САПР-изделий микроэлектроники» (ШМУ-2006) .....	1	58
Международный научно-методический симпозиум «Современные проблемы многоуровневого образования» .....	1	59
От аналогового телевидения к цифровым видеосистемам и видеокomпьютерным технологиям	3	2
Юсупов Рафаэль Мидхатович. Творческая биография .....	6	54

---

## ПАМЯТКА ДЛЯ АВТОРОВ

*Поступающие в редакцию статьи проходят обязательное рецензирование.*

При наличии положительной рецензии статья редактируется и рассматривается редакционной коллегией. Принятая в печать статья направляется автору для согласования редакторских правок. После согласования автор представляет в редакцию окончательный вариант текста статьи.

Процедуры согласования текста статьи могут осуществляться как непосредственно в редакции, так и по e-mail (80x@mail.ru).

При отклонении статьи редакция представляет автору мотивированное заключение и рецензию, при необходимости доработать статью — рецензию. Рукописи не возвращаются.

*Редакция журнала напоминает, что ответственность за достоверность и точность рекламных материалов несут рекламодатели.*