

# ИНФОРМАЦИОННО- УПРАВЛЯЮЩИЕ СИСТЕМЫ

НАУЧНО-ПРАКТИЧЕСКИЙ ЖУРНАЛ



ФГУП НПО "Аврора" – лидер в области  
автоматизированных систем управления  
судовыми техническими средствами

НПО  
**аврора**



**4/2003**

4/2003

# ИНФОРМАЦИОННО-УПРАВЛЯЮЩИЕ СИСТЕМЫ

Учредитель и издатель:

ФГУП «Издательство «Политехника»»

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

Редакционная коллегия:

В. Г. Анисимов,  
доктор технических наук, профессор

В. Ф. Мелехин,  
доктор технических наук, профессор

А. В. Смирнов,  
доктор технических наук, профессор

В. А. Фетисов,  
доктор технических наук, профессор

В. И. Хименко,  
доктор технических наук, профессор

А. А. Шальто,  
доктор технических наук, профессор

А. П. Шепета,  
доктор технических наук, профессор

З. М. Юлдашев,  
доктор технических наук, профессор

**Редактор:** Л. М. Манучарян

**Корректор:** Е. П. Смирнова

**Дизайн:** М. Л. Черненко

**Компьютерная верстка:** Ю. А. Окунева,

А. А. Буров

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

**Адрес редакции:** 191023, Санкт-Петербург,

Инженерная ул., д. 6

Тел./факс: (812) 312-53-90

E-mail: asklab@aanet.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 в любом отделении связи по каталогу агентства «Роспечать». Индекс 15385.

**Банковские реквизиты издательства:**

Получатель: ИНН 7825331278, ФГУП «Издательство «Политехника»»,

Р/с 40503810819000001855 в Куйбышевском филиале

ОАО «Банк Санкт-Петербург» в г. Санкт-Петербург,

К/с 30101810900000000790, БИК 044030790, ОКОНХ 87100,

## ОБРАБОТКА ИНФОРМАЦИИ И УПРАВЛЕНИЕ

**Андреев С. В.** Алгоритм оптимального управления движением в условиях неполной определенности среды

2

## МОДЕЛИРОВАНИЕ СИСТЕМ И ПРОЦЕССОВ

**Астапкович А. М.** Формализм адресно-временных карт для описания алгоритмов функционирования многоканальных систем управления

6

**Дашевский В. П.** Методика вероятностного анализа наборов задач в однопроцессорных системах реального времени с фиксированными приоритетами

15

## ЗАЩИТА ИНФОРМАЦИИ

**Фомин А. В., Бржезовский А. В.** Расщирения реляционной модели для обеспечения безопасности в базах данных

24

## ИНФОРМАЦИОННЫЕ КАНАЛЫ И СРЕДЫ

**Красюк В. Н., Платонов О. Ю., Мельникова А. Ю.** Особенности распространения радиоволн миллиметрового диапазона, перспективы их использования в современных радиотехнических системах

33

**Красюк В. Н., Горбацкий В. В.** Особенности применения микрополосковых антенн на летательных аппаратах из композитов с малыми характеристиками рассеяния

39

## УПРАВЛЕНИЕ В МЕДИЦИНЕ И БИОЛОГИИ

**Тихонов Э. П.** Алгоритмы обработки сигналов в медицинской диагностике с использованием опорного случайного процесса

43

## КРАТКИЕ СООБЩЕНИЯ

**Шальто А. А.** Новая инициатива в программировании. Движение за открытую проектную документацию

52

## РЕЦЕНЗИИ

«Организация ЭВМ и систем»  
Б. Я. Цилькера и С. А. Орлова

57

## ИНФОРМАЦИЯ

23-я Международная конференция «Школьная информатика и проблемы устойчивого развития»

58

## СВЕДЕНИЯ ОБ АВТОРАХ

59

## АННОТАЦИИ

61

ЛР № 010292 от 18.08.98.

Сдано в набор 29.09.2003. Подписано в печать 18.11.2003. Формат 60x90/8.

Бумага офсетная. Гарнитура Pragmatica. Печать офсетная.

Усл. печ. л. 8,0. Уч.-изд. л. 11,3. Тираж 1000 экз. Заказ 835.

Отпечатано с готовых диапозитивов в ООО «Политехника-сервис».

191023, Санкт-Петербург, Инженерная ул., д. 6.

Оригинал-макет изготовлен в ФГУП «Издательство «Политехника»»,

191023, Санкт-Петербург, Инженерная ул., д. 6,

и отделе электронных публикаций и библиографии ГУАП,

190000, Санкт-Петербург, ул. Б. Морская, 67.

УДК 623.681

# АЛГОРИТМ ОПТИМАЛЬНОГО УПРАВЛЕНИЯ ДВИЖЕНИЕМ В УСЛОВИЯХ НЕПОЛНОЙ ОПРЕДЕЛЕННОСТИ СРЕДЫ

**С. В. Андреев,**

аспирант

Санкт-Петербургский государственный университет  
аэрокосмического приборостроения (ГУАП)

*Рассмотрены различные подходы в области нечетких систем с целью разработки алгоритмов управления движением в условиях неполной определенности среды. Предложен алгоритм оптимизации пути следования объекта управления на некоторой местности, описание которой не полностью определено. Поставлена и решена задача автоматизированного принятия решений по выбору оптимального пути с учетом вероятностных свойств среды. Созданный оптимизатор пути следования основан на оптимизационном анализе информации о текущей ситуации и принимает решение в зависимости от условий управления.*

*Given article is devoted to studying of indistinct systems with the purpose of development of algorithms of management by movement in conditions not to full definiteness of environment. During research various approaches were investigated in the field of indistinct management. In particular, the methods based on application of expert systems and combinatory tasks, realizing search of optimum managing influence on object were offered.*

*The algorithm of optimization of a transit of object of management on some district which description is not completely determined is developed. The task of the automated acceptance of decisions is put and solved at the choice of an optimum way, with the account probability properties of environment. Created algorithm of transits, it is based on optimum analysis of the information on the current situation and makes a decision depending on conditions of management.*

## Введение

Применение средств цифровой вычислительной техники в информационно-измерительных и управляющих системах обусловило широкую номенклатуру методов и средств управления и принятия решений в различных ситуациях. В последнее время все более широко стало использоваться нечеткое управление, формирование управляющих сигналов в котором связано с нечетким принятием решения, предполагающим, как правило, достижение некоторой цели или, по крайней мере, последовательное приближение к некоторому наиболее предпочтительному варианту (эталону) в условиях неполной определенности. Системы с таким управлением относятся к нечетким системам управления (НСУ). В НСУ логико-вероятностного типа целью может быть выбор путем логического вывода наиболее предпочтительного решения, имеющего наибольшую вероятность из всех альтернативных решений. При этом в зависимости от сложности задачи управления и системы могут с большим

или меньшим успехом использоваться различные вычислительные методы принятия решения. К наиболее перспективным методам следует отнести математическое программирование в порядковых шкалах, обобщенное математическое программирование, основанные на использовании бинарных отношений [1, 2].

В общем случае задача оптимизации такой НСУ сводится к поиску наилучшего из всех альтернативных решений. При этом альтернативным решением будем называть вариант решения, удовлетворяющий ограничениям задачи управления и обеспечивающий получение требуемых параметров и характеристик управления, а средой задачи — эталонную модель объектов и системы управления, а также тех условий, в которых предполагается функционирование НСУ, которые необходимо учитывать при формировании и решении задачи.

Обычно в НСУ можно выделить подсистемы, имеющие четкие описания, для которых можно использовать модели всех известных типов в зависимости от типа подсистемы, и подсистемы, имеющие

нечеткие описания, для которых можно использовать модели логико-вероятностные либо логико-лингвистические, в зависимости от типа подсистемы. Нечеткие описания характерны для интеллектуальных и человеко-машинных подсистем. Неприятной особенностью таких подсистем является векторность целевой функции. Последнее требует нетрадиционных подходов к решению оптимизационных задач управления, имеющих высокую сложность и, как правило, размерность [3].

Таким образом, представляется целесообразным при формализации описания нечетких систем попытаться выделить в них четкие подсистемы того или иного типа и нечеткие подсистемы, описываемые логико-вероятностными моделями. Обычно к четкому описанию можно свести системы управления, целями которых являются стабилизация, программное управление, слежение или самонастройка. Если целями системы управления является целесообразное поведение или живучесть сложной системы, то четкого формализованного описания, как правило, получить не удастся. Однако и в таких системах управления можно выделить подсистемы стабилизации, программного управления, слежения и самонастройки, которые удастся описать как четкие [4, 5].

### Постановка задачи исследования

Пусть имеется транспортное средство, которое находится на какой-то местности, описание которой не полностью определено. Причем положим, что местность представляет собой карту, поделенную на определенное количество равных квадратов. Задачей ОУ будет отыскание оптимального пути следования по карте до момента достижения заданной точки. Ставится задача максимально автоматизировать процесс принятия решений по выбору оптимального пути. Для этой цели и создается разрабатываемый оптимизатор пути следования (ОПС) на основе информации о текущей ситуации.

Для придания конкретики уточним, что понимается под текущей ситуацией. Слово «ситуация» (*situatio*) означает: положение или сочетание условий и обстоятельств, создающих определенную обстановку, положение. Применительно к поставленной задаче под ситуацией будем понимать конкретизацию следующих моментов, выявленных в определенном временном периоде:

- границы применимости и распространения рассматриваемой ситуации;
- объекты, непосредственно составляющие ситуацию;
- условия, определяющие положение каждого объекта в ситуации;
- условия, устанавливающие правила взаимодействия объектов между собой и с объектами, не принадлежащими к данной ситуации;
- факторы, обуславливающие направление и вид взаимодействия между объектами;
- вероятностные свойства всех составляющих аспектов, приводящие к улучшению или ухудшению ситуации.

Для упрощения задачи будем полагать, что ОПС имеет непосредственный доступ к электронной карте местности, построенной на основе точной передачи координат со спутниковых систем или с помощью предварительного исследования местности. Это позволит нам с необходимой точностью классифицировать условия задачи, в частности, иметь четкое представление о различных коэффициентах и количественно-качественных характеристиках задачи управления.

Первый этап управления будет включать в себя следующие идентификационные задачи:

- идентификации местоположения (сопоставление по карте, добавление недостающих объектов);
- системы координат (звездная, солнечная, высота над уровнем моря и т. д.);
- оценки погодных условий (осадки, облачность, ветер);
- параметрические коэффициенты состояния каждого участка местности.

И только после сбора информации об условиях управления до необходимого уровня начинают работать классификатор и анализатор, обрабатывая полученную информацию для составления целостной картины текущей ситуации. В частности, анализу подлежат сравнительные характеристики участков местности, такие как рельеф, характер, а также находящиеся на них объекты. И только после тщательного анализа ОПС переходит ко второму этапу управления — к решению оптимизационной задачи путем использования исследуемых и разрабатываемых алгоритмов. Один из применяемых на этом этапе алгоритмов рассмотрим подробнее.

### Алгоритм оптимизации траектории движения

Основной смысл данного алгоритма заключается в следующем. При обращении системы к оптимизатору пути следования происходит инициализация текущего состояния. В частности, система обрабатывает информацию о координатах объекта на местности, с тем чтобы привязать точку начала движения к координатной сетке.

Объект управления, получив от системы управления задание, должен двигаться к определенной точке на местности. При этом происходит подробный анализ пути до этой точки. И если на траектории движения оказывались непредвиденные системой управления препятствия, ОУ должен давать экспертную оценку возникающей помехи и корректировать траекторию движения с целью достижения точки назначения. В том случае, если окончательная траектория движения установлена, ОУ начинает движение к точке, во время которого могут возникать три дополнительных помехи, вероятность которых моделируется, т. е. на произвольных участках траектории с какой-то долей вероятности могут возникнуть непредвиденные помехи для движения. В частности, моделировались закрытые для проезда дороги, ранее не замеченное препятствие, а также существенное изменение погодных условий, ока-

зываются зависимость на приоритетный выбор того или иного пути следования.

Оптимизатор пути следования на основе электронной карты местности получает задание подобрать наиболее подходящий поставленным условиям путь передвижения от начальной до конечной точки. Карта представляет собой направленный граф, каждое ребро которого имеет определенный набор вероятностных коэффициентов, определяемых по имеющейся таблице.

Таким образом, в зависимости от поставленной задачи на граф наносятся соответствующие веса, которые будут различны для оптимальной скорости и, например, оптимального поиска. А так как каждый коэффициент имеет свой вес, определенный условиями задачи, то данный граф будет заполнен соответствующими весами, по которым ОПС и будет определять оптимальную траекторию движения. Затем полученный связанный граф обрабатывается программой по нижеприведенной методике. В результате чего на выходе программы получается оптимальный путь — от начальной заданной точки до конечной и несколько альтернативных решений.

Далее для моделирования поведения управляемого объекта на третьем (заключительном) этапе в процесс принятия программой решения добавляются различные ограничения на процесс формирования управляющего воздействия. В частности, вводятся «непроходимые» коэффициенты, а также моделируются различные характеристики ЛПР, в зависимости от которых принимаются принципиально различные решения. Например, при исследовании ЛПР «боязливого» типа получаются более длинные траектории движения, но относительно более безопасные. Причем все возможные ограничения имеют динамический характер и обрабатываются оптимизатором непосредственно в процессе выполнения задания. При этом имеется возможность влиять на принятие решений как коренным, так и незначительным образом.

Очевидно, что поиск оптимального пути может быть найден полным перебором, но это заведомо приведет к значительной потере скорости подсчета. В данном алгоритме предложен подход к решению подобных комбинаторных задач, основанный на отсеке большого числа априорно неоптимальных связей между точками. Поясним этот подход с помощью рис. 1, рассмотрев восемь центральных элементов, граничащих друг с другом.

Казалось бы, из центральной точки идет восемь возможных путей, но это не так. Во-первых, в предложенном алгоритме производится заполнение промежуточных массивов, что ведет к исключению закливания и заметно увеличивает общую скорость работы алгоритма. Таким образом, если принять, что в точку  $nm$  мы пришли из точки, обозначенной как  $n(m-1)$ , то помимо непосредственно самой точки  $n(m-1)$  мы сразу имеем возможность исключить все общие пограничные точки  $nm$  и  $n(m-1)$ . Другими словами, исключаются точки  $((n-1)m, (n-1)(m-1), (n+1)m, (n+1)(m-1))$ . Это объясняется тем, что любая цепочка, захватываю-

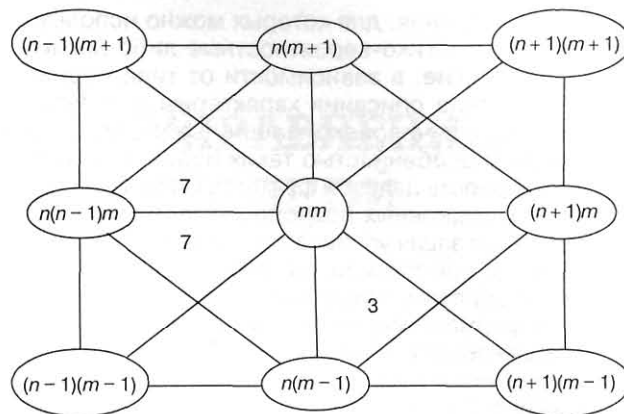


Рис. 1. Восемь произвольных центральных элементов

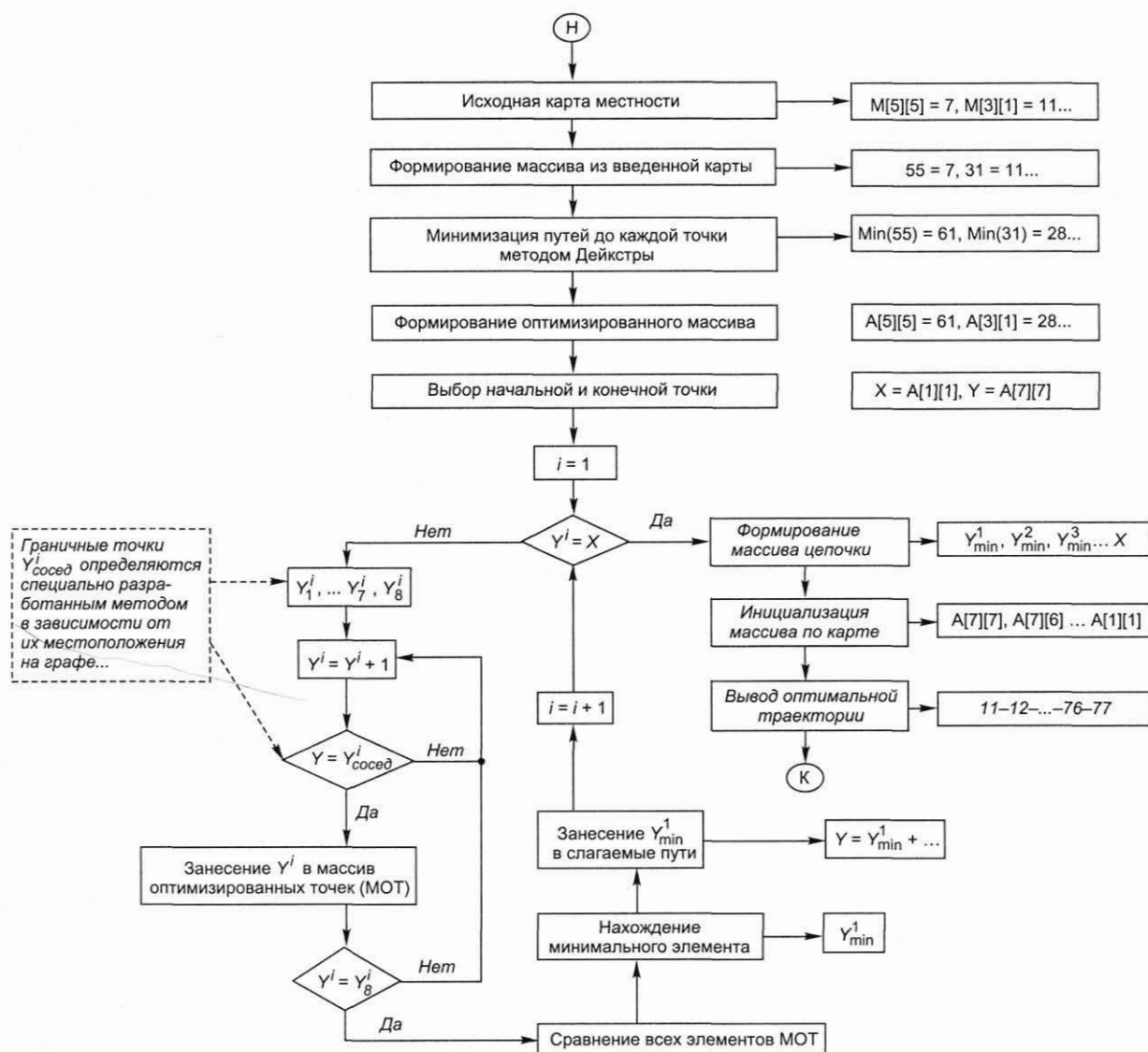
щая  $nm$ , будет заведомо длиннее, чем прямая связь. Таким образом, из центральной точки логично рассматривать только три возможных выхода, обозначенные как  $(n-1)(m+1), n(m+1), (n+1)(m+1)$ .

Вторым важным аспектом предложенного алгоритма является процесс заполнения специальной массива уже пройденных точек для того, чтобы исключить повторное прохождение одной и той же точки, а следовательно, и исключить возможное закливание оптимальной траектории движения (рис. 2).

Кратко поясним общий ход алгоритма. Первоначально происходит инициализация условий работы. В частности, заданная карта местности приводится к виду, с которым оперирует рассматриваемая программа. Для этого карта разбивается на равные квадраты, которым присваиваются соответствующие коэффициенты. Далее связанный граф формируется таким образом, что учитываются особенности местности и делается вывод о направлении его дуг, т. е. исключаются заведомо неоптимальные пути при построении необходимой траектории движения. Это позволяет в дальнейшем существенно сократить время работы программы. Затем применяется метод Дейкстры [6]. Результатом является получение минимально возможного пути для каждой вершины графа. Однако мы не можем считать рассчитанный минимум за окончательное решение задачи, так как задача алгоритма — найти несколько оптимальных траекторий, чтобы путем их анализа выбрать наилучший, удовлетворяющий всем условиям.

Далее происходит улучшенный перебор всех возможных вариантов. Из начальной точки начинают выстраиваться всевозможные пути с учетом ранее введенных ограничений и коэффициентов. Причем в алгоритм дополнительно вводится блок запоминания уже пройденных точек, называемый массивом оптимизированных точек, что позволяет исключить закливание в программе.

В результате работы программы мы получаем информацию о пяти предпочтительных траекториях движения, оцененных по степени максимального удовлетворения всех условий задачи. Полученные



■ Рис. 2. Алгоритм оптимизированного поиска на графе

траектории проходят процедуру обратной трансформации для представления их в заданной форме. Далее они изображаются графически на начальной карте местности и управление передается оператору для выбора им любого из пяти предложенных вариантов движения.

Таким образом, предложенный алгоритм реализации оптимизационной задачи управления позволяет найти необходимое решение задачи, на порядок сократив время поиска оптимальной траектории на карте местности. Однако данный алгоритм является узкоспециализированным, так как рассчитан на заведомо известную окружающую среду со статическими характеристиками. В дальнейшем представляется целесообразным его последовательное усложнение путем введения в условия задачи динамически изменяющихся условий среды, а также вероятностных коэффициентов для различных характеристик задачи управления.

### Литература

1. Городецкий А. Е., Тарасова И. Л. Интегрированные системы автоматизации НИОКР // Учебное пособие. — СПб.: Изд-во СПбГТУ, 1998. — 96 с.
2. Городецкий А. Е., Дубаренко В. В. Комбинаторный метод вычисления вероятности сложных логических функций // ЖВМ и МФ. — 1999. — Т. 39. — № 7. — С. 1246–1248.
3. Табак Д., Куо Б. Оптимальное управление и математическое программирование. — М.: Наука, 1975. — 234 с.
4. Тимофеев А. В., Юсупов Р. М. Интеллектуализация систем автоматического управления // Изв. РАН. Техническая кибернетика. — 1994. — № 5. — С. 31–37.
5. Юдин Д. Б. Вычислительные методы теории принятия решений. — М.: Наука, 1974. — 234 с.
6. Липский В. Комбинаторика для программистов. — М.: Мир, 1988. — С. 101–105.

УДК 681.325.5: 518.5

## ФОРМАЛИЗМ АДРЕСНО–ВРЕМЕННЫХ КАРТ ДЛЯ ОПИСАНИЯ АЛГОРИТМОВ ФУНКЦИОНИРОВАНИЯ МНОГОКАНАЛЬНЫХ СИСТЕМ УПРАВЛЕНИЯ

### Введение в формализм адресно–временных карт

**А. М. Астапкович,**

канд. техн. наук, начальник СКБ

Санкт-Петербургский государственный университет  
аэрокосмического приборостроения (ГУАП)

*Работа посвящена изложению нового способа укрупненного описания алгоритмов функционирования многоканальных систем управления реального времени. Предложенный формализм включает в себя понятия: процессограмма, циклограмма, адресно-временные карты. Рассматриваются алгоритмические аспекты использования предложенного способа описания и символическое представление алгоритмов для наиболее распространенных дисциплин диспетчеризации. Вводятся три типа конфликтов в глобальном конкурентном пространстве.*

*Article concerns the new method of the robust description of the multichannel real-time control system algorithm. Proposed formalism includes: processogram, cyclogram, address-time card. Algorithm aspects were examined also as symbolic description for the most popular scheduler disciplines. Three types of the conflicts in the global concurrent space were introduced*

#### Введение

В связи с бурным развитием встраиваемых систем управления проблема существенного повышения скорости разработки прикладного программного обеспечения (ППО) для них приобретает особую актуальность. Особый интерес представляют многоканальные микропроцессорные системы управления реального времени встраиваемого класса, для которых принципиально важной является реализация предписанного алгоритма управления по каждому из каналов за время, не превышающее специфицированного. Высоконадежные многоканальные системы управления, обрабатывающие асинхронные входные воздействия, относятся к подклассу так называемых «систем жесткого реального времени».

Характерной особенностью таких систем является их «закрытость», т. е. неизменяемость программного обеспечения в нормальных режимах эксплуатации. Для такого класса систем управления в качестве структурной основы прикладного программного обеспечения, как правило, используют специализированные операционные системы ре-

ального времени (ОСРВ) [1, 2]. Однако наблюдается тенденция все более широкого использования микрооперационных систем реального времени (mОСРВ) [3], представляющих собой узкоспециализированные ОСРВ для конкретного класса микропроцессорных устройств. С учетом этого проблема автоматизации системной интеграции применительно к многоканальным системам управления реального времени в настоящее время выдвигается на первый план. Решение задачи системной интеграции требует обеспечения бесконфликтного разделения ресурсов между параллельно выполняющимися задачами, а также обеспечения специфицированных времен реакции по каждому из каналов на всем допустимом множестве входных воздействий на объект управления. Вторая по важности задача заключается в необходимости, с одной стороны, обеспечить мобильность разрабатываемого программного обеспечения, а с другой — по возможности автоматизировать процесс интеграции.

Критически взглянув на сложившееся положение в области разработки ППО систем реального времени, можно констатировать, что современные

методы в большей степени фокусируются на разработке собственно логической структуры алгоритма управления, а временной характер задач и их взаимное влияние остаются в тени. Позаимствовав из прикладной математики классификацию «прямая задача — обратная задача», будем рассматривать разработку ППО по традиционной технологии как решение прямой задачи. Причем ее решение осуществляется фактически подбором как с точки зрения логической структуры алгоритма, так и обеспечения временных параметров системы управления.

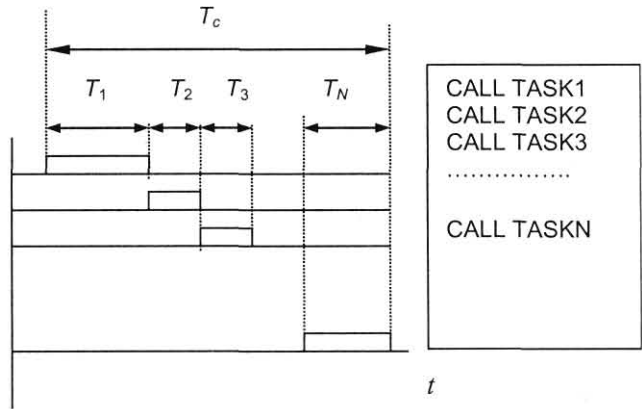
Для многоканальных систем управления, в силу сложного характера информационного взаимодействия, детальное описание алгоритма функционирования становится громоздким из-за необходимости рассматривать поведение системы при широком спектре входных воздействий на длительных интервалах времени. Известна методика применения аппарата имитационного моделирования в сочетании с применением формализма сетей Петри [4, 5], которая в большей степени ориентирована на решение задач анализа логической структуры алгоритмов с использованием технологии типа «прямая задача». Иногда такой подход называют «синтезом через анализ». Сети Петри в этом случае используются как универсальный способ описания структуры исследуемых алгоритмов, а аппарат имитационного моделирования обеспечивает выработку входных воздействий и обработку полученных результатов.

Отсутствие должного внимания к временному аспекту логики работы алгоритма резко ограничивает возможность применения данного аппарата к многоканальным системам управления реального времени. При ближайшем рассмотрении оказывается, что переход к концептуально новым подходам упирается в проблему описаний алгоритмов функционирования, приспособленных к специфике многоканальных систем управления реального времени, когда необходимо некоторое укрупненное описание, позволяющее проводить анализ наличия конфликтов в глобальном конкурентном пространстве для многозадачной ОСРВ. Если пользоваться употребляемой терминологией, то речь идет о задаче анализа критической секции.

В статье сделана попытка введения математического аппарата, предназначенного для укрупненного и формально строгого описания алгоритмов функционирования многоканальных систем управления реального времени с учетом доступных ресурсов (память, время реализации и т. п.), исходя из общесистемных требований.

### Процессограмма и циклограмма

В работе [3] было введено понятие «процессограмма», которое в явном виде отображает параллельный характер процессов. Процессограмма является естественным расширением средств описания алгоритмов функционирования устройств, применяемых до настоящего времени.



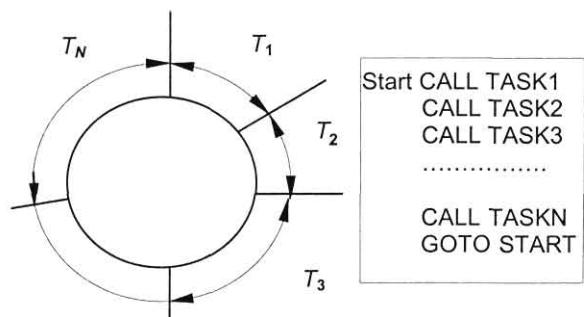
■ Рис. 1. Процессограмма для диспетчера FIFO (бесприоритетная линейная очередь с обслуживанием в порядке поступления)

Иллюстрацией этого понятия является рис. 1, где каждая из задач имеет свою длительность  $T_i$ . Структура алгоритма диспетчеризации предельно проста: управление последовательно передается от подпрограммы к подпрограмме. Если принять, что каждая из подпрограмм обеспечивает управление по одному из каналов, то можно рассматривать бесконечный цикл как вариант реализации бесприоритетного циклического диспетчера. В терминах ОСРВ такую диспетчеризацию иногда называют «ядром с кооперативным взаимодействием задач» [1]. Графически эта дисциплина диспетчеризации лучше всего отображается в виде круговой диаграммы, которую будем называть циклограммой (рис. 2).

Процессограмму можно рассматривать как описание одного цикла циклограммы, следовательно, это взаимно дополняющие друг друга понятия. Время опроса  $T_c$  всей очереди задач в соответствии с рис. 1 и 2 описывается соотношением

$$T_c = \sum_{i=1}^{N_i} T_i. \quad (1)$$

Как следует из рис. 1 и 2 и формулы (1), для диспетчеризации с кооперативным взаимодействием задач параллельная обработка входных воздействий обеспечивается при условии, что время



■ Рис. 2. Иллюстрация понятия «циклограмма»



обработки всей очереди задач  $T_c$  будет меньше, чем минимальное специфицированное время реакции для всего множества  $N_t$  контролируемых каналов. В случае однопроцессорных систем управления речь идет о псевдопараллельной обработке, хотя с точки зрения конечного пользователя обеспечивается функциональная параллельность. Более детальное обсуждение этой проблемы было представлено в работе [3].

Для того чтобы подчеркнуть универсализм предлагаемого описания, рассмотрим несколько дополнительных примеров. Использование процессограммы для описания начальной фазы функционирования встроенной системы управления иллюстрирует рис. 3. На рис. 4 приведена процессограмма, описывающая диспетчеризацию с квантованием времени при использовании одного из системных таймеров. За время одного цикла опроса очереди задач длительностью  $T_c$  каждой из задач выделяется одинаковый квант времени  $T_k$ . Для реализации такого диспетчера мОСРВ потребуется время на распознавание источника прерывания, время  $T_{isr}$  на его обработ-

ку, на сохранение контекста текущей выполняемой программы, на обновление счетчиков очередей самого диспетчера, на восстановление контекста следующей по очереди задачи. Все это займет время  $T_{disp}$ . Таким образом, полное время опроса очереди команд  $T_c$  с учетом «накладных расходов» определяется как

$$T_c = \sum_{i=1}^{N_{ch}} (T_k + (T_{isr} + T_{disp})) = \sum_{i=1}^{N_{ch}} (T_k + T_{sys}).$$

Для операционных систем с кооперативной организацией взаимодействия задач или систем с квантованием времени  $T_c$  представляет собой оценку времени реакции системы  $T_r$  на внешние воздействия по любому из параллельных каналов.

Системы диспетчеризации с приоритетами, как правило, широко используют систему прерываний. Возможность укрупненного описания алгоритма функционирования системы управления, использующей прерывания, с помощью процессограмм показана на рис. 5. Следует отметить, что, если речь идет о многоканальных системах управления «жесткого реального времени», диспетчеризация, использующая систему прерываний, приводит к целому ряду проблем, связанных с тем, что время ее реакции становится уже вероятностной характеристикой системы управления. Детальному рассмотрению круга проблем, связанных с использованием прерываний, посвящена работа [6].

### АТ-модель глобального конкурентного пространства

В современном стандарте на программное обеспечение POSIX понятие критической секции (critical section) соответствует определению «конкурентное пространство» и «разделяемые ресурсы». В самом общем случае в конкурентное пространство входят: общие периферийные модули, используемые разными процессами, общие зоны памяти (вы-

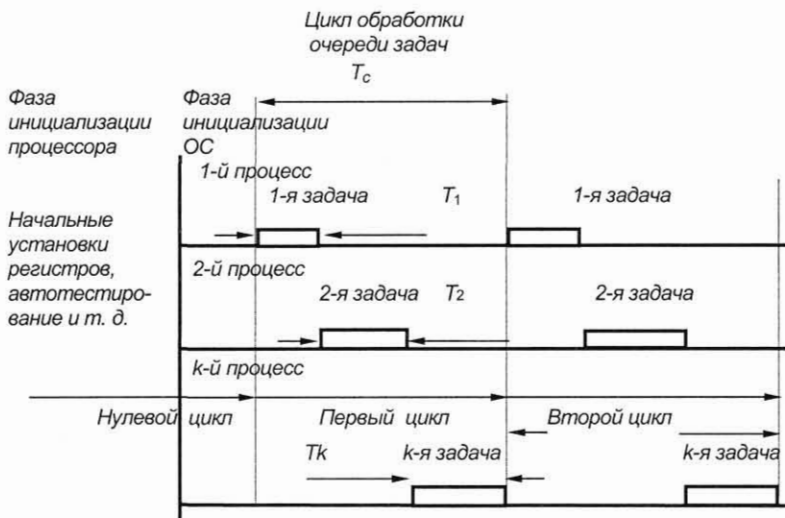


Рис. 3. Процессограмма начальной стадии работы многоканальной системы управления

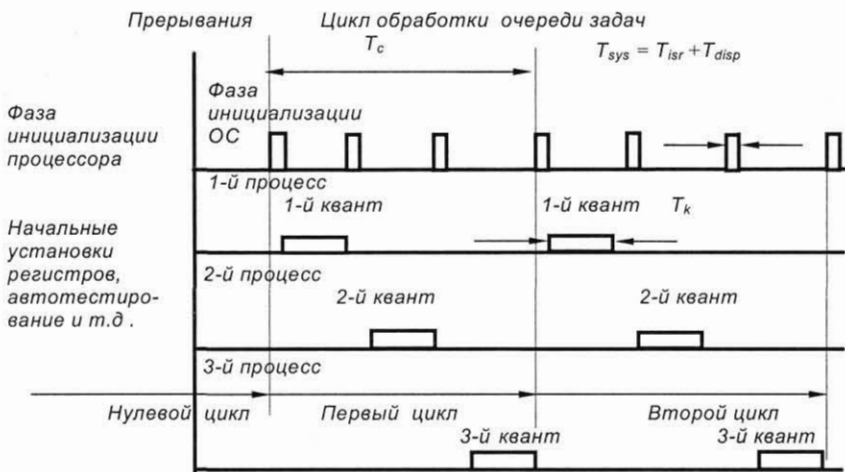


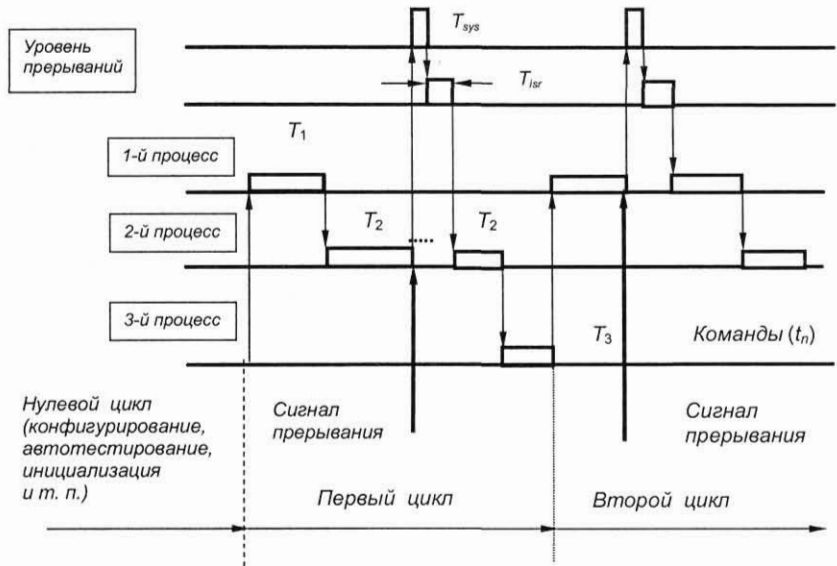
Рис. 4. Процессограмма диспетчера с квантованием времени

зов одних и тех же подпрограмм из разных процессов), общие временные интервалы (для однопроцессорных систем). На рис. 6 представлен пример графического описания ресурсов современного микропроцессора, разделяемых параллельными процессами между собой в динамическом режиме. Для систем управления реального времени требуется описание временной последовательности реализации алгоритма за счет введения некоторой дискретизации и соответствующего описания оси времени, например, за счет номера выполняемого такта.

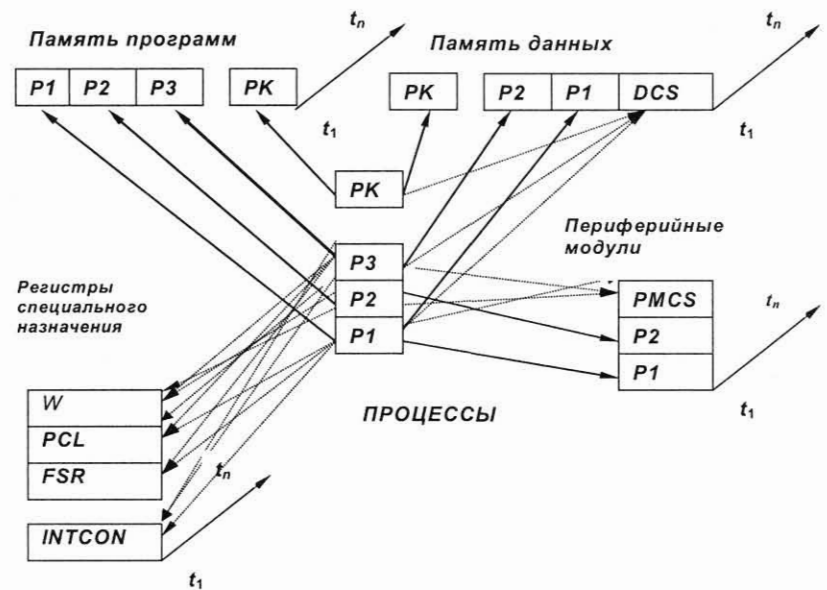
Описание разделяемых ресурсов как совокупности используемых адресов и временных тактов, задействованных конкретной задачей, процессом, каналом управления и т. п., будем называть адресно-временной моделью (АТ-моделью от англ. Address-Time) глобального конкурентного пространства [3]. Такая АТ-модель, представленная на рис. 6, описывает ресурсы микропроцессора с гарвардской архитектурой, в которой память программ и память данных разделены. Она же явно отражает факт динамически меняющейся ситуации, о чем свидетельствует ось времени, присвоенная к каждому конкретному виду разделяемых между процессами ресурсов. По оси времени отображаются дискретные отсчеты, что соответствует тактам микропроцессора, реализующего алгоритм управления.

На низком уровне описание алгоритмов функционирования осуществляется на ассемблере, в котором каждому из регистров присваивается уникальное символическое имя. Для упорядочения всех доступных ресурсов микропроцессора представляется разумным присваивать уникальные номера всем регистрам, а также элементам ядра микропроцессора, которые, по сложившейся к настоящему времени традиции, будем называть адресами. Это позволяет однозначно идентифицировать каждый элемент микропроцессора.

Проблема синтеза алгоритмов функционирования многоканальных систем управления реального времени требует разработки



■ Рис. 5. Процессограмма описания диспетчеризации с приоритетами



■ Рис. 6. Структура глобального конкурентного пространства:

*P1—PK — процесс № 1—K; DCS — общее пространство памяти данных (Data Concurrent Space); PMCS — разделяемые периферийные модули (Peripheral Module Concurrent Space);  $t_1$ — $t_n$  — временные отсчеты, согласованные с тактовой частотой процессора; W, PCL, FSR, INTCON — регистры: рабочий, счетчика команд, неявной адресации и контроля прерываний, соответственно*

эффективных методов контроля за разделяемыми ресурсами. Представляется целесообразным разделить ее на следующие задачи: разработка логической структуры алгоритма по каждому из каналов; бесконфликтное разделение адресно-временных ресурсов между параллельно выполняющимися задачами.

При наличии библиотек прикладных программ для выполнения типовых задач систем управления стоит проблема корректного переноса соответствующих фрагментов кода в тело разрабатываемого программного обеспечения. Это, в свою очередь, требует корректного разделения адресного пространства между микроопераци-

онной системой, прикладными задачами, реализующими логическую структуру алгоритма управления, и задачами управления периферийными устройствами.

Отдельную проблему представляет задача обеспечения мобильности программного обеспечения в связи с необходимостью переопределения адресов при переносе фрагментов кода на различные устройства с однотипными функциями, но отличающиеся друг от друга по схемотехнике.

В этом случае в зависимости от конкретных условий может потребоваться решение не только задачи бесконфликтного разделения адресного конкурентного пространства, но и задачи корректного разделения адресно-временного конкурентного пространства. Можно также говорить о поставке оптимизационной задачи по выбору типа используемого микропроцессора.

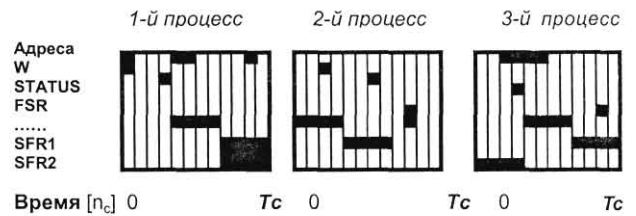
### Адресно-временная карта процесса

АТ-модель глобального конкурентного пространства допускает простой способ формального описания. Единое и универсальное описание глобального конкурентного пространства может быть получено за счет присвоения каждому элементу конкурентного пространства своего номера и дискретизации временной оси в соответствии с монотонно возрастающим числом командных циклов, начиная с некоторого начального отсчета. В двумерном адресно-временном множестве, элементы которого распределяются между параллельно выполняющимися процессами, каждому из них будет соответствовать некоторое подмножество, которое в дальнейшем будем называть адресно-временной картой процесса или АТ-картой.

Возможность такого представления для описания распределения используемых регистров между параллельно выполняющимися процессами совершенно естественна для микропроцессоров, использующих концепцию регистрового файла, и требует некоторых предварительных соглашений о нумерации регистров для других архитектур. Поскольку любой процесс состоит из задач, каждая из которых представляет собой некоторую совокупность последовательно выполняемых команд, то описание временной оси можно представить в дискретном виде. Соответствующая АТ-карта  $k$ -го процесса представляется в виде прямоугольной матрицы описания  $AT_k$  с элементами:

$$AT_k(i, j) = \begin{cases} 1 & \text{если } k\text{-й процесс в } j\text{-м такте} \\ & \text{использует } i\text{-й адрес} \\ & \text{микропроцессора;} \\ 0 & \text{в противном случае.} \end{cases}$$

На рис. 7 приведена графическая иллюстрация предлагаемого способа описания используемых ресурсов микропроцессора с помощью АТ-карт. Ее можно применять и для описания алгоритмов функционирования распределенных (сетевых) систем управления. При этом АТ-карты описания ресурсов,

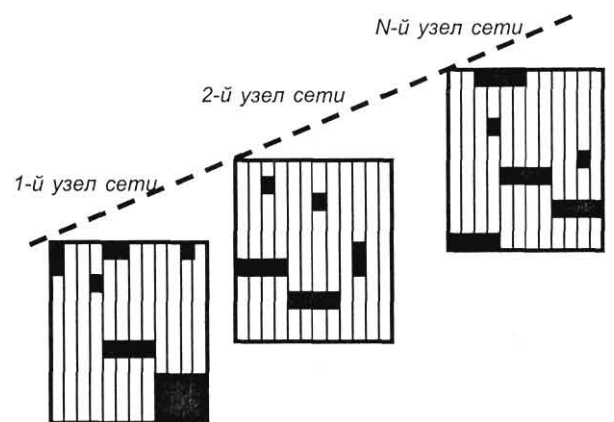


■ Рис. 7. Иллюстрация к определению понятия «адресно-временная карта»

используемых каждым из узлов, образуют трехмерную АТ-карту, как это показано на рис. 8. В этом случае логично ввести понятие  $AT_N$ -карты, представляющей собой трехмерную матрицу описания системы параллельно выполняющихся процессов. Символ  $N$  соответствует порядковому номеру узла сети.

Предложенный способ описания параллельных процессов с помощью адресно-временных карт и матриц позволяет достаточно просто сформулировать критерий наличия конфликтной ситуации, который к тому же достаточно просто алгоритмизируется. В первом приближении наличие конфликта между параллельными процессами для корректно сгенерированной прикладной программы означает использование одного и того же адреса в один и тот же квант времени более чем одним процессом [3]. Поставим каждому из параллельно выполняющихся процессов в соответствие АТ-карту, описывающую использование конкретных адресов микропроцессора в конкретных командных циклах, отсчитываемых от предписанного начального значения (например, от начала опроса очереди задач). Тогда при суммировании матриц описания карт адресно-временных подмножеств, соответствующих разным процессам, получим формальный и алгоритмически простой критерий отсутствия конфликтов между параллельно выполняющимися процессами в одном цикле опроса очереди задач в виде

$$\max_{i, j} \sum (AT_k) = 1.$$



■ Рис. 8. Вариант использования трехмерных АТ-карт ( $AT_N$ -матриц) для описания алгоритмов функционирования сетевых систем управления

В качестве меры в пространстве конфликтов можно использовать значение максимального элемента суммы АТ-карт процессов.

При использовании алгоритмов обработки разреженных матриц, хорошо разработанных для численного решения многомерных задач математической физики, матрица описания АТ-карты представляется в виде трехмерного массива (или трех одномерных массивов равной размерности) описания ненулевых элементов в виде

$$AT = \{i, j, V_{i,j}\}. \quad (2)$$

Фигурные скобки в выражении (2) означают массивы описания АТ-карты, организованные тем или иным способом: в виде трехмерного массива, в виде трех одномерных массивов, в виде структуры и т.п. Для современных микропроцессоров значение размерности адресной компоненты лежит в интервале от 100 до 10 000 0000, а количество командных циклов определяется спецификой решаемой задачи. Представление (2) можно усовершенствовать за счет более рационального способа описания АТ-карты. Если трактовать  $V_{i,j}$  как количество ненулевых элементов карты, начиная с элемента  $(i, j)$ , то удастся существенно сократить общий объем описаний требуемых ресурсов. Согласно этому способу, наличие тройки  $(i, j, V_{i,j})$  в описании АТ-карты означает, что адрес  $i$  занят на протяжении  $V_{i,j}$  тактов, начиная с  $j$ -го. Так как описываются только ненулевые компоненты матрицы, то для полноты описания потребуется знание размерности массивов описания (например,  $N_1$  для  $AT_1$ ), а также границ  $A_{beg}$ ,  $A_{end}$ ,  $T_{beg}$  и  $T_{end}$ . При таком способе представления ненулевых элементов описание АТ-карты представляет собой структуру, содержащую размерности  $[N_A, N_T, N_1]$  и трехмерный массив описания индексов ненулевых элементов  $\{i, j\}$ :

$$AT_k = [N_A, N_T, N_1] \{i, j, V_{i,j}\}_k, \quad (3)$$

где квадратные скобки означают структуру, содержащую размерности описания  $k$ -й АТ-карты. Речь идет о структуре в смысле определения элементов данных, используемых в языке программирования С.

Поскольку элементы  $V_{i,j}$  матриц полного описания АТ-карт могут принимать лишь два значения (0 или 1), то описание (2) без потери информативности может быть представлено индексами  $i$  и  $j$ . При этом предполагается, что наличие такой пары означает, что соответствующий элемент матрицы описания  $V_{i,j}$  равен 1. При таком способе представления ненулевых элементов описания АТ-карты представляет собой тройку размерностей  $(N_A, N_T, N_1)$  и двумерный массив описания индексов ненулевых элементов  $\{i, j\}$  в виде

$$AT_k = (N_A, N_T, N_1) \{i, j\}_k. \quad (4)$$

При этом предполагается, что наличие пары  $(i, j)$  в описании означает, что соответствующий элемент АТ-карты равен 1, хотя в принципе не исключено

и применение инверсного способа описания используемых адресов. Понятно, что рассмотренные выше представления одномерны и относятся к временной оси АТ-карты. Аналогичные способы описания могут быть использованы для адресной компоненты АТ-карты. В этом случае дальнейшие обобщения могут потребоваться при описании некоторой зоны памяти на протяжении заданного числа циклов, что приведет к следующей базовой структуре описания в виде:

$$AT_k = \{i_{beg}, j_{beg}, A_{i,j}, T_{i,j}\}_k, \quad (5)$$

которая подразумевает, что адреса процессора с  $i_{beg}$  по  $i_{beg} + A_{i,j}$  на моменты времени с  $j_{beg}$  по  $j_{beg} + T_{i,j}$  закреплены за  $k$ -й задачей.

Универсальный и развиваемый способ описания АТ-карт представляет собой возможность использования выражений (2) – (5) и ряда других, потребность в которых может возникнуть при решении конкретных задач. В этом случае для каждого типа описаний потребуется введение своего идентификатора, а выполнение операций с АТ-картами будет представлять собой операции с массивами описаний, определяемыми непосредственно типом выполняемой операции и способом описания конкретной АТ-карты. Трудоемкость решения прикладных задач в смысле количества требуемых операций будет зависеть от специфики решаемой задачи, но с точки зрения прикладной математики это сводится к хорошо изученным задачам сортировки массивов, а также переборным алгоритмам.

### Символьный способ описания

При более глубоком рассмотрении проблемы анализа реализуемости алгоритмов с точки зрения обеспечения специфицированных времен реакции на внешние воздействия оказывается, что одного лишь численного представления недостаточно. В качестве иллюстрации этого рассмотрим одну важную проблему, с которой приходится сталкиваться на практике. Зачастую встроенные системы управления работают в необслуживаемом режиме на протяжении длительного времени.

В качестве простейшего примера можно привести, например, многотарифный трехфазный счетчик электроэнергии, который должен обеспечивать измерение значения потребляемого тока, рассчитывать мгновенную потребляемую мощность по каждой фазе и количество потребленной энергии по каждой тарифной зоне, обеспечивать коммуникационные процедуры служб технической поддержки, обрабатывать сбои и т. п. Общее число параллельно обрабатывающихся процессов в такой системе составляет от 20 до 30. Важно отметить, что межповоротный интервал такого класса устройств составляет 10 лет, при этом, как правило, должна быть обеспечена точность измерений порядка 0,5–1,0 %.

Микропроцессор такой системы конфигурируется на самой первой стадии работы, далее ини-

циализируется mOCPB и контролируемые ею задачи. От корректных значений параметров инициализации задач зависит корректность работы всей системы управления и устройства в целом. Для корректной работы диспетчера, который является частью mOCPB, если позволяют временные характеристики, следует включить в цикл опроса очереди задач и задачу инициализации процессора, для обеспечения гарантированных начальных условий выполнения задач, образующих процессы. В целях экономии циклов процессора на инициализацию возможна диспетчеризация посредством двух вложенных очередей, что иллюстрирует пример 1. Здесь использован синтаксис ассемблера микропроцессорной платформы PIC16/17.

**Пример 1.** Диспетчер для двух вложенных очередей

```

bcf    INTCON,GIE    ; запрет прерываний
LINE0  call  INIT      ; инициализация процессора,
                        ; диспетчера и задач
LINE1  call  TASK1     ; вызов 1 задачи
      call  TASK2     ; вызов 2 задачи
      .....
      call  TASKN     ; вызов задачи N
      call  COUNT     ; вызов служебной
                        ; подпрограммы диспетчера
                        ; подсчета числа проходов
      btfs  PSW,INIT_FLAG ; бит флага необходимости
                        ; инициализации
goto   LINE0         ; устанавливается
                        ; подпрограммой COUNT
goto   LINE1         ; формирование временного
                        ; процесса
    
```

Диспетчеризация осуществляется таким образом, что во внутреннем цикле, обрамленном меткой LINE1 и оператором goto LINE1, осуществляется поочередный вызов задач. С помощью служебной подпрограммы COUNT, являющейся принадлежностью диспетчера и соответственно mOCPB, производится подсчет числа проходов по внутреннему циклу. При равенстве числа проходов некоторому заданному значению подпрограмма COUNT выставляет соответствующий бит INIT\_FLAG в регистре PSW. В конце каждого прохода очереди команд производится проверка этого флага. Диспетчер, обнаружив, что флаг выставлен, передает управление во внешний цикл, в котором происходит реинициализация процессора, диспетчера, задач. В этом модельном примере рассматривается диспетчеризация без прерываний, и для этого перед началом работы система прерываний блокируется выставлением бита GIE в регистре управления системой прерываний INTCON.

Расширение приведенного подхода возможно, во-первых, на большее число задач в редко выполняющейся очереди, начинающейся с метки LINE0, во-вторых, на большее число вложенных очередей. Каждая вложенная очередь потребует отдельного регистра (по крайней мере, одного) на организацию счетчика, а также от 4 до 10 операций за один цикл выполнения очереди задач, которые уйдут на его инкрементирование, проверку его переполне-

ния и передачу управления. Следует обратить внимание на то, что использование многопетлевого диспетчера в приведенном примере с включением во внешние циклы объемных по времени задач, существенным образом ухудшает время реакции системы управления  $T_r$ . В случае двухпетлевого диспетчера в качестве оценки времени реакции системы управления будет фигурировать время, требуемое на инициализацию задач, суммированное со временем их выполнения.

Приведенный пример представляет возможность рассмотреть еще один аспект использования аппарата AT-карт для описания разнообразных дисциплин диспетчеризации. Если между частотой вызовов подпрограмм внутреннего цикла и инициализацией имеется существенное различие, то процессорные и соответственно AT-карты, отображаемые непосредственно на мониторе, много теряют в своей наглядности. Для того чтобы добраться до цикла инициализации, придется достаточно долго «листать» экраны. С точки зрения потребностей реальной практики, целесообразно развивать способы символического описания суммарных AT-карт. Идею их использования иллюстрирует формула (6), описывающая циклограмму диспетчера из примера 1:

$$AT_s = \sum_{k=1}^{\infty} AT_0 + \sum_{COUNT=1}^{N_1} \sum_{i=1}^{N_i} AT_i. \quad (6)$$

Один цикл прохода очереди задач описывается суммарной картой, сформированной из карт  $AT_i$ , общее число которых составляет  $N_1$ . Эти карты соответствующим образом складываются  $N_1$  раз, образуя суммарную карту внутреннего цикла. К этой карте добавляется карта  $AT_0$ , описывающая процесс инициализации. Сформированная AT-карта репродуцируется бесконечное количество раз, что отражает символ  $\infty$  во внешнем цикле суммирования. При детальном рассмотрении оказывается, что использование напрямую операций суммирования матриц в соответствии с классической теорией матриц возможно, но неудобно. Поэтому необходимо разработать символичный способ описания AT-карт, а также ввести иерархию операций для работы с прямоугольными матрицами, что детально будет рассмотрено далее [3].

Представление (6) позволяет компактно и обзорно представить алгоритм формирования суммарной AT-матрицы, обеспечивая тем самым доступ к любому фрагменту циклограммы. Необходимо также отметить, что альтернативы этому способу нет, так как даже при самом экономном способе организации хранения элементов AT-карт невозможно будет хранить AT-карту бесконечного цикла. В соответствии с изложенным становится ясно, что при разработке соответствующего программного инструментария для работы с AT-картами целесообразно ориентироваться на применение комбинированных представлений в виде иерархических структур из численных и символических описаний.

### Классификация конфликтов в глобальном конкурентном пространстве

Из-за сложного характера информационного взаимодействия детальное описание алгоритма функционирования систем управления с числом каналов управления, превышающим 10, становится громоздким. Это обусловлено необходимостью рассматривать сложное поведение системы при широком спектре входных воздействий на длительных интервалах времени. Рассмотрим возможность использования АТ-карт для анализа конфликтных ситуаций. Начнем с более детального рассмотрения определения конфликта в глобальном конкурентном пространстве, введя его в следующем виде.

**Определение 1.** Наличие АТ-конфликта между параллельными процессами для корректно сгенерированной прикладной программы означает попытку использовать один и тот же адрес в один и тот же командный цикл более чем одним процессом. При суммировании матриц описания карт адресно-временных подмножеств, соответствующих разным процессам, получаем легко формализуемый и алгоритмически простой критерий отсутствия конфликтов между параллельно выполняющимися процессами в виде

$$\max_{i,j} \sum_k (AT_k) = 1, \quad (7)$$

где  $i, j$  — индексы для описания элементов АТ-карт ( $i$  — индекс описания адреса;  $j$  — индекс нумерации столбцов в командных циклах);  $k$  — индекс в описании количества параллельно выполняющихся процессов.

Использованная в выражении (7) операция суммирования матриц представляет собой обыкновенную операцию поэлементного суммирования матриц и подразумевает, что суммируемые матрицы имеют одинаковые размерности. Это накладывает следующие ограничения на выбор формы описания АТ-карт. Во-первых, следует использовать полное адресное пространство микропроцессора в совокупности с регистрами специального назначения. Во-вторых, в качестве начального отсчета, соответствующего первому столбцу каждой из АТ-карт, следует брать командный цикл, соответствующий запуску микрооперационной системы. В-третьих, требуется выбор отрезка временной оси, на котором будет проводиться анализ на наличие конфликта, так как для большинства типовых применений суммарная матрица описания АТ-карты бесконечна.

Отметим, что выполнение условия (7) соответствует отлаженной программе. При автоматизированном синтезе ППО для многоканальной системы управления потребуется решение задачи разделения ресурсов между параллельными процессами, так как с большой вероятностью на первых этапах синтеза АТ-карты разных процессов будут иметь пересечения. В этом случае максимум  $F(*, *) = \max \sum (AT_k)$  можно использовать в качестве функ-

ционала, численно отражающего конфликтность ситуации. В качестве аргументов в этом функционале будут выступать свободные для подбора параметры, определяющиеся конкретной задачей. Использование такого функционала обеспечивает формальное сведение задачи бесконфликтного разделения адресно-временных ресурсов к классической задаче целочисленного программирования. Однако интуитивно ясно, что функционал явно не будет попадать в класс унимодальных, и на практике в дальнейшем потребуются разработка соответствующих специализированных численных методов.

Критерий наличия конфликта в адресном пространстве для задач, описываемых картами  $AT_1$  и  $AT_2$ , может быть сформулирован с использованием аппарата классической теории множеств в следующем виде.

**Определение 2.** Наличие А-конфликта между двумя программами означает, что пересечение множества адресов, используемых каждой из подпрограмм, не пусто, т. е.  $\{i\}_1 \cap \{i\}_2 \neq \emptyset$ .

Вводя соответствующий формализм, определим множества  $\{i\}$  как проекции АТ-карт на адресную ось, что может быть записано в виде

$$[\perp A] AT_1 = \{i\}_1, [\perp A] AT_2 = \{i\}_2. \quad (8)$$

Квадратные скобки в выражении (8) использованы для удобства обозначения и восприятия конкретной операции. А-конфликт служит характеристикой того, что разные процессы имеют попеременный доступ к одним и тем же ресурсам, например, переменным. В рамках использования предлагаемого формализма это означает совпадение индексов  $i$  в описаниях АТ-карт, принадлежащих разным задачам. Понятно, что в силу самой постановки задачи предлагаемое описание не учитывает ни семантику вычислений, ни алгоритмические аспекты вычислений. Вместе с тем введение такого рода конфликта позволяет автоматизировать поиск ошибочно введенных глобальных переменных, которые являются неиссякаемым источником ошибок при интеграции программистских проектов.

При этом речь идет не обязательно о системах управления реальным временем. Использование адресной составляющей АТ-карт позволяет провести анализ на совпадение символьных имен (адресов) в рамках интеграции отработанных процедур в состав вновь разрабатываемого ППО. При этом могут решаться задачи бесконфликтного разделения как памяти, так и периферийных устройств: портов ввода-вывода, коммуникационных модулей, модулей АЦП и т. п. Дальнейшее движение по этому пути позволяет сформулировать не только задачу бесконфликтного разделения ресурсов при заданном типе микропроцессора, но и задачу выбора микропроцессора.

По аналогии с А-конфликтом может быть введена операция проекции АТ-карты на ось временных отсчетов в виде  $[\perp T] AT_1 = \{j\}_1$ .

**Определение 3.** Наличие Т-конфликта между двумя программами означает, что пересечение мно-

жества временных отсчетов, используемых каждой из программ, не пусто. В этом случае критерий наличия конфликта между двумя процессами в пространстве временных отсчетов записывается в виде  $\{t\}_1 \cap \{t\}_2 \neq \emptyset$ .

Такого рода характеристика может потребоваться при проектировании коммуникационных процедур для многоканальных систем управления. В этом случае наличие Т-конфликта означает потерю информации по одному из конфликтующих каналов. Понятие Т-конфликта позволяет формулировать как задачу проверки адекватности выбора тактовых частот, так и соответствующие задачи оптимизации.

Введенная система определений обеспечивает следующие классификации конфликтов в глобальном конкурентном пространстве: Т- конфликт, А-конфликт, АТ- конфликт. Очевидно, что одновременное отсутствие А-конфликта и Т-конфликта является достаточным условием корректности программы при использовании критической секции. Соответственно явно просматриваются способы построения численных процедур автоматического синтеза программного обеспечения для многоканальных систем управления реальным временем. Развитие соответствующих методов и методик автоматизированного решения подобного класса задач следует вести параллельно с разработкой соответствующих программно-инструментальных средств, что представляет собой задачу следующего шага.

## Литература

1. **Астапкович А. М.** Аналитический обзор: высокоуровневые программно-инструментальные среды для встраиваемых приложений реального времени // Микропроцессорные информационно-управляющие системы реального времени. Семинары ASK Lab 2000/Под ред. М. Б. Сергеева. — СПб.: Политехника, 2000. — С. 4–39.
2. **Астапкович А. М., Востриков А. А., Гуляев А. М.** Аналитический обзор: операционные системы реального времени для встраиваемых приложений // ВУТЕ Россия, 2000. — № 9. — С. 29–37.
3. **Астапкович А. М.** Микрооперационные системы (mОСРВ) для встроенных систем управления // Информационно-управляющие системы и сети. Структуры. Моделирование. Алгоритмы / Под ред. М. Б. Сергеева. — СПб.: Политехника, 1999. — С. 196–224.
4. **Основы имитационного и статистического моделирования** / Ю. С. Харин и др. — Минск: Дизайн ПРО, 1997. — 288 с.
5. **Игнатъев М. Б., Фильчаков В. В., Осовецкий Л. Г.** Активные методы обеспечения надежности алгоритмов и программ. — СПб.: Политехника, 1992. — 285 с.
6. **Астапкович А. М.** Вероятностное поведение многоканальных систем управления реальным временем и логическое проектирование алгоритмов // Микропроцессорные информационно-управляющие системы реального времени. Семинары ASK Lab 2000/Под ред. М. Б. Сергеева. — СПб.: Политехника, 2000. — С. 104–134.

УДК 681.32

# МЕТОДИКА ВЕРОЯТНОСТНОГО АНАЛИЗА НАБОРОВ ЗАДАЧ В ОДНОПРОЦЕССОРНЫХ СИСТЕМАХ РЕАЛЬНОГО ВРЕМЕНИ С ФИКСИРОВАННЫМИ ПРИОРИТЕТАМИ

**В. П. Дашевский,**

аспирант

Санкт-Петербургский институт

информатики и автоматизации РАН (СПИИРАН)

*Предложена методика вероятностного анализа выполнимости периодических задач в системе реального времени. Методика позволяет гарантировать качество обслуживания каждой задачи в системе, где максимально возможная загрузка процессора превышает единицу, а средняя остается меньше единицы. Представленный алгоритм численного анализа является эффективным как в отношении машинного времени, так и в отношении требуемой оперативной памяти.*

*This paper describes probabilistic analysis of periodic real-time tasks with varying execution times. The analysis provides performance guaranties for the task sets with total maximum utilization higher than one while the average utilization remains less than one. Each task may be pre-empted by higher priority task and has deadline not greater than its period. Proposed algorithm is memory and time-efficient.*

## Введение

Системами реального времени (СРВ) принято считать такие системы, в которых наряду с логической правильностью вычислительного процесса необходимо обеспечивать также своевременность его протекания. При разработке и тестировании этих систем приходится учитывать не только функциональный, но и временной аспект.

В большинстве случаев процесс проектирования вычислительной системы реального времени начинается с абстрактного описания ее функциональных возможностей, указания ресурсных и временных ограничений. Затем весь спектр функциональных возможностей представляется в виде набора более простых задач, для выполнения которых выбираются процессорные элементы. В завершение для каждого процессорного элемента назначается дисциплина планирования, в соответствии с которой проводится анализ временных требований задач.

Во многих приложениях СРВ нарушение сроков выполнения задач может приводить к катастрофическим последствиям. По этой причине основные методики анализа систем реального времени, развитые в 1970–1980 годы, были ориентированы на наихудшее время выполнения программ (НВВП). Недостаток такого подхода заключается в том, что

время выполнения программ сравнительно редко приближается к наихудшему. В большинстве случаев оно остается существенно меньше, поэтому процессоры оказываются слабо загруженными. Кроме того, методы повышения производительности вычислительных систем, такие как кэширование данных, предсказание переходов, конвейерная обработка с неявным запретом прерываний, использование контроллеров прямого доступа в память, лишь увеличивают разрыв между средним и наихудшим временем выполнения. Таким образом, консервативные методики анализа с использованием лишь НВВП накладывают все большие ограничения на эффективность использования аппаратных средств.

В последние годы появилось много новых приложений СРВ, в которых нарушение сроков задач не приводит к заметным последствиям, пока вероятность таких нарушений остается меньше некоторого порога. Таковы, например, различные приложения мультимедийных и телекоммуникационных систем. Некоторые авторы отмечают, что анализ устойчивости в системах «чистого» управления также допускает редкое нарушение сроков без заметной потери управляемости. Указанные особенности создают предпосылки для развития методик вероятностного анализа СРВ, с помощью которых можно повысить эффективность использования



аппаратуры, а значит снизить затраты на оборудование.

Методы вероятностного анализа, развитые ранее, опираются на простые модельные законы распределения времени выполнения, такие как равномерный или экспоненциальный закон распределения. Более точные модели используют произвольные распределения, однако одной из основных трудностей при работе с ними является быстро растущая с числом задач сложность анализа.

В данной работе рассматривается методика вероятностного анализа набора задач для одного процессора, адаптированная для широко применяемой дисциплины планирования с фиксированными приоритетами задач. Законы распределения времени выполнения задач могут быть выбраны произвольно, но при некоторых предположениях анализ может быть еще более упрощен без существенной потери точности. Анализ является эффективным как в отношении машинного времени, так и в отношении требуемой оперативной памяти.

Практически во всех исследованиях, посвященных CPB, каждая из задач решается системой многократно. В первую очередь это связано со спецификой применения CPB в качестве подсистем сбора информации и управления. Для описания задач традиционно используются несколько параметров: период повторения запросов на решение задачи, относительный срок выполнения, время выполнения, время блокировки из-за доступа к общим данным. Разными авторами допускаются различные предположения относительно значений этих параметров.

Методики анализа систем с детерминированными параметрами времени выполнения, рассмотренные в работах [2, 5–10, 13–15, 17], основываются на предположении, что время выполнения задачи постоянно. В системах с жесткими требованиями на выполнение задач в срок приходится рассчитывать систему исходя из НВВП для каждой задачи. Чаще всего это приводит к выводу о перегрузке системы и невозможности ее работы в реальном времени, однако, если производительности процессора хватает для наихудшего случая, то система оказывается слабо загруженной в среднем, что приводит к выводу о плохой эффективности ее использования.

Авторы [16] предлагают пять различных алгоритмов планирования для аperiodических задач. Используемая ими модель задач предполагает случайные времена возникновения запросов на выполнения задач и постоянные времена выполнения, оцененные исходя из НВВП.

В работе [1] рассмотрена система, в которой времена выполнения задач могут быть случайными. На основе работы [7] авторами разработана методика вероятностного анализа набора задач, с помощью которой можно оценить вероятность нарушения срока для каждой задачи. Однако вычислительная сложность предложенной методики оказывается довольно высокой, поэтому авторы прибегают к приближенным вычислениям, используя центральную предельную теорему. Использование

приближенных вычислений может приводить к завышенным оценкам вероятности выполнения задачи в срок, что делает проблематичным использование подобной методики для систем с жесткими требованиями. Авторами предложен также другой метод организации вычислений, при котором задачи с большим разбросом времени выполнения разбиваются на две подзадачи, первая из которых выполняется за время, не превышающее среднего времени выполнения исходной задачи, а вторая представляет собой случайно возникающий остаток ее выполнения. Первая подзадача планируется аналогично другим задачам с детерминированными характеристиками, а вторая ставится в очередь к спорадическому серверу. Показано, что с помощью подобных преобразований удастся достигнуть более высокой эффективности использования процессоров, однако разделение далеко не всегда удается произвести на практике, что ограничивает применение этого метода.

В работе [11] авторы обобщают результаты классической работы [2] на случай задач со случайными временами выполнения. Каждая задача характеризуется периодом повторения запросов, функцией распределения времени выполнения, а также качеством обслуживания, представляющим собой необходимую вероятность выполнения задачи в срок. Помимо планирования задач система осуществляет контроль над их принятием в расписание, с помощью которого, во-первых, ни одна задача не использует больше процессорного времени, чем ей гарантировано, во-вторых, ни одна задача не принимается в расписание, если ее выполнение не может быть гарантировано. Подобный контроль обеспечивает защиту качества обслуживания задач с большим периодом (т. е. имеющих меньший приоритет в соответствии с RMS) от вытеснения задачами с меньшим периодом. Авторами предложен алгоритм анализа выполнимости набора задач при условии, что времена выполнения всех задач распределены равномерно. К сожалению, методика анализа не может быть легко обобщена на случай произвольного распределения времени выполнения задач.

Альтернативная методика анализа набора задач со случайным временем выполнения предложена в работе [3]. В ее основе также лежит анализ системы с детерминированными параметрами задач, развитый в работе [7]. Подобно работе [1] целью анализа является оценка нижней границы для вероятности выполнения задач в срок. Предлагаемая методика учитывает также влияние необходимости завершения предыдущих запросов на обработку текущего запроса, которое становится особенно заметным, когда средняя загрузка CPB приближается к единице, а максимальная превосходит ее. Однако приводимые автором данные о сравнении результатов развитой методики с численным моделированием того же набора задач показывают, что она дает слишком заниженную оценку на вероятность выполнения задач в срок. Чем выше становится перегрузка системы, тем сильнее отличаются результаты анализа от результатов

моделирования. Из результатов моделирования, полученных автором, видно, что при стохастическом времени выполнения программ искомая вероятность выполнения в срок очень слабо зависит от постоянства фазовых соотношений между запросами на решение различных задач. Автором предложено обобщение методики анализа на случаи задач, в алгоритмах которых присутствуют непрерываемые критические секции, а также на случаи распределенных систем.

Работа [12] является развитием [1]. Однако авторы рассматривают вероятность гарантирования не отдельных задач, а всего набора в целом. Для этого им приходится рассматривать все возможные сочетания времен выполнения задач, в результате чего сложность анализа не позволяет исследовать большое число задач.

В работе [4] показано, что редкое нарушение сроков в задачах «чистого» управления не означает потери устойчивости в управлении, хотя традиционно многие задачи управления (в особенности опасными или дорогостоящими объектами) относятся к задачам жесткого реального времени. Этот результат показывает, что надежность системы управления в реальном времени связана с жесткостью требований реального времени лишь косвенно, и вероятностный анализ может быть применен к системам с жесткими требованиями.

Анализ, приводимый в данной работе, является развитием методики, изложенной в работе [1]. По сравнению с [1] был улучшен алгоритм проверки контрольных точек: уменьшено число проверяемых точек, в которых приходится вычислять свертки, а также уменьшена сложность вычисления самих свертков. Эти упрощения позволяют проводить анализ всей системы задач, не прибегая к центральной предельной теореме, благодаря которой авторам [1] удалось сократить объем вычислений, однако при этом теряются гарантии того, что полученная в результате анализа оценка вероятности является оценкой снизу.

### Формулировка задачи вероятностного анализа

Для простоты изложения ограничимся рассмотрением одного процессора. Методику можно применить и в случае многопроцессорной системы, если предварительно задачи распределить по процессорам, а связи между ними специфицировать таким образом, что можно провести анализ для каждого процессора отдельно.

**Предположения относительно системы задач.** В качестве модели задач используется традиционная для большинства коммерческих операционных систем реального времени модель периодических задач с фиксированными приоритетами, удовлетворяющих следующим предположениям:

1) запросы на решение задач поступают в систему периодически; относительный срок, за который запросы должны быть обработаны, не превосходит периода следования запросов;

2) время выполнения задачи является случайной величиной, распределенной по некоторому, заранее выбранному закону; при этом максимальное время выполнения не превосходит периода повторения запросов;

3) времена выполнения различных задач являются статистически независимыми случайными величинами (в отличие от времен отклика);

4) обработка запроса на выполнение задачи прекращается принудительным образом, если она не укладывается в назначенный срок;

5) задача с более высоким приоритетом вытесняет менее приоритетную, не дожидаясь завершения ее выполнения;

6) время переключения контекста процессора между задачами пренебрежимо мало по сравнению с характерными временами их выполнения;

7) для каждой задачи множество задач, способных ее вытеснить, задается разработчиком системы при проектировании и не изменяется во время функционирования системы;

8) средняя загрузка процессора всеми задачами не превосходит единицы. Однако в силу случайного характера, а также большого разброса значений времени выполнения максимальная загрузка может быть больше единицы.

Поскольку в каждом периоде повторения время вычисления задачи меняется случайным образом, то часть параметров каждой задачи будет зависеть также от номера периода. В дальнейшем будем описывать  $j$ -й запрос  $W_{i,j}$  на решение  $i$ -й задачи  $W_i$  следующими параметрами:

$T_i$  — период повторения запросов на решение задачи;

$D_i$  — относительный срок выполнения;

$\phi_i$  — приоритет задачи;

$r_{i,j}$  — момент поступления запроса на решение задачи;

$c_{i,j}$  — время выполнения запроса (без вытеснения другими задачами);

$d_{i,j}$  — срок завершения выполнения запроса;

$R_{i,j}$  — момент завершения выполнения запроса;

$\bar{c}_i$  — среднее время выполнения запроса;

$c_i^+$  — максимальное время выполнения запроса;

$c_i^-$  — минимальное время выполнения запроса;

$\bar{u}_i$  — средняя загрузка процессора данной задачей;

$u_i^+$  — максимальная загрузка процессора данной задачей;

$u_i^-$  — минимальная загрузка процессора данной задачей;

$f_i(x)$  — функция плотности распределения времени выполнения;

$p_i$  — вероятность выполнения запроса в срок.

Приведенные выше условия в принятых обозначениях можно записать следующим образом:

$$c_{i,j} \leq D_i \leq T_i;$$

$$r_{i,j} = r_{i,1} + (j-1)T_i;$$

$$d_{i,j} = r_{i,j} + D_i;$$

$$\bar{c}_i = \lim_{j \rightarrow \infty} \frac{1}{j} \sum_{k=1}^j c_{i,k};$$

$$c_i^+ = \max c_{i,j};$$

$$c_i^- = \min c_{i,j};$$

$$\bar{u}_i = \frac{\bar{c}_i}{T_i};$$

$$u_i^- = \frac{c_i^-}{T_i};$$

$$u_i^+ = \frac{c_i^+}{T_i};$$

$$\bar{U} = \sum_{i=1}^N \bar{u}_i < 1;$$

$$U^+ = \sum_{i=1}^N u_i^+ > 1;$$

$$p_i = P(R_{i,j} \leq d_{i,j}).$$

В добавление к принятым обозначениям будем также считать, что задачи пронумерованы в соответствии с их приоритетами:

$$i < j \Rightarrow \phi_i > \phi_j. \quad (1)$$

Строго говоря, можно не накладывать ограничения на время выполнения задачи в предположении 2, однако естественно предположить, что каждая отдельно взятая задача не приводит к перегрузке процессора. Таким образом, целью анализа, в первую очередь, является определение частоты нарушения сроков при совместном выполнении задач, т. е. при условии, когда только случайное увеличение времени выполнения сразу нескольких задач в совокупности способно привести к перегрузке процессора.

**Формулировка задачи вероятностного анализа СВВ.** Задача формулируется следующим образом:

Для заданного набора независимых задач  $W_i$ ,  $i = 1, \dots, N$  с указанными вероятностными характеристиками, задаваемыми с помощью функций плотности распределения времени выполнения, требуется определить вероятность  $p_i$  завершения задачи  $W_i$  до назначенного ей срока  $D_i$ . Если полученная вероятность окажется не меньше некоторого заранее заданного значения, то будем считать, что система способна справиться с решением данной задачи в реальном времени.

Сложность проведения анализа подобного рода заключается, как и в случае с детерминированными временами выполнения, в том, что каждая задача может вытесняться в процессе решения более приоритетными задачами, что приводит к увеличению времени отклика. Чем больше становится время отклика, тем большее число задач с более высоким приоритетом могут вытеснить текущую задачу, что приводит к еще большему увеличению времени отклика. Влияние этой положительной связи становится еще более трудно прогнозируе-

мым, если время выполнения задачи может изменяться в широких пределах. В такой ситуации можно ожидать, что вероятностный анализ способен дать лучшие результаты, поскольку наиболее вероятное время отклика существенно отличается от максимально возможного.

### Вероятностный анализ времени выполнения задач

**Профили времени выполнения и операции с ними.** Случайный характер времени выполнения задачи может быть описан с помощью функции распределения времени выполнения. По определению функция распределения времени выполнения задачи в точке  $x$  есть вероятность того, что время выполнения  $c$  не превзойдет значения  $x$ :

$$F(x) = P(c < x).$$

Функцию плотности распределения

$$f(x) = F'(x)$$

для краткости называют также *профилем времени выполнения* задачи [18]. Поскольку время выполнения всегда неотрицательно, то функции распределения определены на положительной полуоси.

В общем случае каждую задачу можно представить совокупностью некоторых элементарных выполняемых блоков, каждый из которых описывается своим профилем времени выполнения. Однако в данной статье ограничимся рассмотрением всей задачи как одного неделимого блока.

Для анализа набора задач необходимо уметь находить профиль суммарного времени выполнения двух и более задач. В общем случае задача является довольно трудной, поскольку суммарное время выполнения нескольких задач зависит от их порядка, количества вытеснений более приоритетными задачами и использования общих данных. Тем не менее в практике достаточно часто встречаются примеры слабо зависимых задач. В данном случае будем исходить из предположения 3 (см. ранее).

Основной операцией над профилями времени выполнения, применяемой для анализа совокупностей независимых задач, является операция свертки

$$h(x) = f_1(x) \otimes f_2(x) = \int_0^x f_1(y) f_2(x-y) dy, \quad (2)$$

с помощью которой можно получить профиль суммы времен выполнения двух задач. Важными свойствами свертки являются:

- коммутативность —  $f_1 \otimes f_2 = f_2 \otimes f_1$ ;
- ассоциативность —  $(f_1 \otimes f_2) \otimes f_3 = f_1 \otimes (f_2 \otimes f_3)$ ;
- дистрибутивность —  $(f_1 + f_2) \otimes f_3 = (f_1 \otimes f_3) + (f_2 \otimes f_3)$ ;

Для дальнейшего изложения полезен также профиль нулевого времени выполнения, или нулевой профиль выполнения, который будем обозначать

$\delta(x)$ . Основное свойство нулевого профиля выполнения состоит в том, что для любого профиля  $f(x)$   $f(x) \otimes \delta(x) = f(x)$ .

В машинном представлении профиль времени выполнения описывается дискретным образом с помощью массива чисел:

$$a_k = P(h(k-1) < c \leq hk), \quad k = 1, \dots, M,$$

где  $h$  — шаг дискретизации;  $M$  — общее количество дискрет, выбираемое таким образом, чтобы покрыть исследуемый участок временной оси для каждой задачи.

Вероятность выполнения задачи за время  $x$  с дискретно заданным профилем  $a_k$  можно оценить снизу как

$$F(x) \geq \sum_{k=0}^{\lfloor \frac{x}{h} \rfloor} a_k. \quad (3)$$

Из выражения (3), в частности, следует, что для оценки снизу вероятности выполнения задачи  $W_i$  достаточно массива размером  $\lfloor \frac{D_i}{h} \rfloor + 1$ .

Для вычисления свертки дискретно заданных профилей  $f_{1,k}$  и  $f_{2,k}$  интеграл в выражении (2) заменяется суммой:

$$f_{3,k} = \sum_{l=0}^k f_{1,l} \cdot f_{2,k-l}.$$

Нулевой профиль выполнения в дискретном случае определяется как

$$\delta_k = \begin{cases} 1, & k = 0 \\ 0, & k > 0, \end{cases}$$

так что для любого дискретного профиля  $f_k$  их свертка

$$(\delta \otimes f)_k = \sum_{l=0}^k \delta_l f_{k-l} = \delta_0 f_k + \sum_{l=1}^k \delta_l f_{k-l} = f_k$$

удовлетворяет определению нулевого профиля выполнения.

Задание профиля времени выполнения для всех задач требует значительного количества памяти, а также приводит к большому времени вычисления свертки [1]. Однако подобное усложнение вычислений не приводит к существенному повышению их точности. Точное измерение функции плотности распределения на практике представляет собой довольно сложную задачу, хотя в большинстве случаев разработчика интересует не столько точное значение, сколько порядок величины вероятности превышения срока.

Чтобы упростить задачу вычисления свертки над дискретными профилями, используем следующие два предположения.

1. Профили многих реальных алгоритмов имеют явно выраженный линейчатый характер, благодаря чему среди чисел, задающих дискретный про-

филь, сравнительно небольшое количество оказываются отличными от нуля. Естественно в такой ситуации задавать профиль не массивом чисел, а массивом пар вида  $(k_i, p_i)$ , где  $k_i$  — номер дискрета, соответствующего времени выполнения  $hk_i$ , а  $p_i$  — вероятность наблюдения этого времени выполнения. Количество таких пар соответствует количеству различных ветвлений программы.

2. Профили времени выполнения, для которых максимальное время выполнения существенно отличается от среднего, должны обладать быстро убывающей функцией  $f(x)$ . В таком случае без существенной потери точности ее можно заменить дискретным профилем, который также описывается набором пар вида  $(k_i, p_i)$ . Величина  $p_i$  представляет собой вероятность  $P(hk_{i-1} < x \leq hk_i)$ . В этом случае количество пар также может быть сделано существенно меньше, чем количество дискрет, необходимого для описания произвольного профиля.

С помощью этих упрощений можно добиться сокращения объема памяти, необходимого для хранения профиля времени выполнения каждой задачи  $W_i$ . Вычисление свертки в таком случае представляет собой операцию над двумя профилями, первый из которых задан дискретным образом в виде массива чисел  $a_m$ , а второй задан парами чисел вида  $(k_i, p_i)$ ,  $i = 1, \dots, n$ :

$$a'_m = \sum_{i=1}^n a_{m-k_i} p_i.$$

Таким образом, вычисление свертки в данном случае требует меньшего количества операций, чем над двумя дискретными профилями. Для анализа системы задач достаточно одного массива чисел, описывающего один дискретный профиль, к которому поочередно добавляются профили отдельных задач. Упрощенные профили позволяют сохранить соотношение (3) для оценки вероятности времени выполнения задачи, вытесняемой другими задачами.

**Алгоритм вероятностного анализа задачи.** Итак, задача анализа состоит в том, чтобы оценить снизу вероятность выполнения запроса  $W_{i,j}$  в срок. Используемая для этого методика базируется на расчете необходимого времени, развитом в работе [7]. Для оценки вероятности снизу мы будем рассматривать критический участок, понятие которого введено в работе [2]. Предполагается, что критический участок начинается тогда, когда запросы на все задачи приходят одновременно. Без потери общности можно считать, что начало критического участка совпадает с началом отсчета на временной оси.

Для оценки вероятности удобно воспользоваться вспомогательной функцией  $w_i(t)$ , которая ограничивает время отклика системы сверху:

$$w_i(t) = c_{i,j} + \sum_{k=1}^{i-1} \sum_{l=1}^{s_k(t)} c_{k,l}, \quad (4)$$

где

$$s_k(t) = \left\lfloor \frac{t}{T_k} \right\rfloor$$

представляет собой наихудшую оценку количества запросов задачи  $W_k$ , которые будут приняты в системе за время  $[r_{i,j}, r_{i,j} + t)$  при любом  $t > 0$ . Поскольку время выполнения каждого запроса является случайной величиной, то приходится явно суммировать  $s_k(t)$  случайных величин  $s_{k,l}$ . Первая сумма в выражении (4), в соответствии с предположением (1), представляет собой суммирование времен всех более приоритетных задач, а вторая учитывает многократные запросы на выполнение одной задачи.

Завершение выполнения запроса  $W_{i,j}$  происходит тогда, когда время, прошедшее с момента его поступления, становится не меньше необходимого времени

$$w_i(t) \leq t$$

для некоторого  $t$  из отрезка  $[r_{i,j}, r_{i,j} + D_i]$ . Поскольку в зависимости от  $t$  количество слагаемых в выражении (4) меняется, то естественно разбить этот отрезок на участки, в которых функции  $s_k(t)$ ,  $k \in \{1, \dots, i\}$  сохраняют постоянное значение. Границами этих участков являются, очевидно, точки временной оси, принимающие значения вида

$$l \cdot T_k, \quad l = 0, 1, 2, \dots, \left\lfloor \frac{D_i}{T_k} \right\rfloor; \quad k = 0, 1, \dots, i. \quad (5)$$

Обозначим множество всех участков с постоянными  $s_k(t)$  через  $\mathbf{C}_i$ . Множество их границ, описываемое выражением (5), с включенной точкой  $D_i$  обозначим  $\mathbf{B}_i$ :

$$\mathbf{B}_i = D_i \cup \{l \cdot T_k : l = 0, 1, 2, \dots, \left\lfloor \frac{D_i}{T_k} \right\rfloor; k = 0, 1, \dots, i\}.$$

Зная функции распределения отдельных задач и их количество на каждом участке, можно построить функцию распределения величины  $w_i(t)$ , вычислив их свертку.

Обозначим через  $E(x)$  вероятность того, что  $w_i(t) < x$ . На каждом участке  $h \in \mathbf{C}_i$  наибольшее значение  $E(x)$  достигается для наибольшего  $x \in h$ , которое соответствует завершению участка постоянства  $s_k(t)$  в момент прихода очередного запроса на решение одной из задач с более высоким приоритетом. Дискретный характер изменения количества запросов во времени приводит к разрыву первого рода для  $E(x)$  на границах участков. Поскольку начало участка  $h$  связано с появлением нового запроса, то значение  $E(x)$  справа от разрыва всегда оказывается меньше, чем слева. В общем случае значение  $E(x)$  на правом конце участка  $h$  может быть как больше, так и меньше значения  $E(x)$  на правом конце предыдущего участка (примыкающего к  $h$  слева). Это связано с тем, в каком соотношении находится длина участка  $h$  и среднее время выполнения одного или, возможно, нескольких запросов, добавившихся в его начале. Если среднее время обработки нового запроса оказывается меньше, чем длина участка  $h$ , то, в среднем, у системы появляется дополнительное время для выполнения запроса

задачи  $W_i$ , и тогда вероятность выполнения задачи  $W_i$  к концу участка  $h$  становится больше. В противном случае появление новых запросов только уменьшает время, которое остается системе для выполнения задачи  $W_i$ , и тогда изменение  $E(x)$  оказывается отрицательным.

Для того чтобы обработка запроса  $W_{i,j}$  на критическом участке была завершена в срок, достаточно, чтобы она завершилась на любом из участков  $h \in \mathbf{C}_i$ . Поэтому можно оценить вероятность выполнения запроса снизу:

$$p_i \geq \max_{x \in h, h \in \mathbf{C}_i} E(x). \quad (6)$$

Поскольку наибольшее значение  $E(x)$ ,  $x \in h$  достигается на правой границе участка  $h$ , то выражение (6) можно записать в виде

$$p_i \geq \max_{x \in \mathbf{B}_i} E(x - 0). \quad (7)$$

Чтобы упростить вычислительную сложность анализа, перебор значений множества  $\mathbf{B}_i$  удобно вести в порядке возрастания. В этом случае вычисление свертки для функции распределения  $E(x)$  можно производить последовательно, выполняя по одной операции свертки на каждую точку из  $\mathbf{B}_i$ . Алгоритм вероятностного анализа представлен на

#### Входные данные

Набор задач  $S = \{W_i\}$ , их периоды  $T_i$ , относительные сроки  $D_i$ , функции плотности распределения  $f_i(x)$  и приоритеты  $\phi_i$ ,  $i = 1, 2, \dots, N$ .

#### Выходные данные

Нижняя оценка  $p_i$  вероятности того, что запрос на решение задачи  $W_i$  будет выполнен в срок, в соответствии с формулой (7)

#### Алгоритм

```

for each  $W_i$ ,  $1 \leq i \leq N$  do
     $g_i(x) = \delta(x)$ 
     $p_i = 0$ 
     $X = 0$ 
     $t_k = 0, \quad 1 \leq k \leq i$ 
    while  $X \leq D_i$  do
         $p = \int_0^X g_i(x) dx$ 
         $p_i = \max\{p_i, p\}$ 
         $L = \{m : t_m = \min_{1 \leq k \leq i} t_k\}$ 
         $l : \phi_l = \max_{m \in L} \phi_m$ 
         $X = t_l$ 
         $g_i(x) = g_i(x) \otimes f_l(x)$ 
         $t_l = t_l + T_l$ 
    end while
end for
    
```

■ Рис. 1. Алгоритм вероятностного анализа набора задач

рис. 1. Общее количество вычислений свертки можно оценить по формуле

$$|B_i| = \sum_{k=1}^i \left\lceil \frac{D_i}{T_k} \right\rceil. \quad (8)$$

**Оценка сложности алгоритма.** Основной вклад в сложность алгоритма вносит вычисление свертки и вычисление вероятности  $P(c < x)$  для дискретного профиля, полученного в очередной контрольной точке из множества  $B_j$ . Общее количество контрольных точек определяется формулой (8).

В общем случае вычислительная сложность алгоритма оценивается

$$C = \sum_{x \in B_i} C_x,$$

где  $C_x$  — сложность анализа в контрольной точке  $x$ . В каждой точке  $x$  приходится вычислять вероятность  $p_i$ , а также очередную свертку для  $g_i(x)$ . Значение  $C$  можно оценить сверху так:

$$C \leq |B_i| \cdot \max_{x \in B_i} C_x \leq |B_i| (C_{pr} + C_{conv}),$$

где  $C_{pr}$ ,  $C_{conv}$  — оценки сверху для сложности вычисления вероятности и свертки, соответственно.

Сложность вычисления вероятности можно оценить сверху величиной

$$C_{pr} = n_i \rho_{add}.$$

Сложность вычисления свертки можно оценить сверху величиной

$$C_{conv} = M_i n_i (\rho_{add} + \rho_{mul}).$$

В приведенных формулах  $M_i$  обозначено количество пар чисел, задающих упрощенный профиль для задачи  $W_i$ ;  $\rho_{add}$  — относительный вес сложности операции сложения;  $\rho_{mul}$  — относительный вес сложности операции умножения;  $n_i = \left\lceil \frac{D_i}{h} \right\rceil$  — количество дискрет в профиле выполнения задачи  $W_i$ ;  $h$  — шаг дискретизации профиля выполнения в машинном представлении.

Таким образом, общая сложность вероятностного анализа для задачи  $W_i$  оценивается сверху выражением

$$n_i (M_i \rho_{mul} + (M_i + 1) \rho_{add}) \sum_{k=1}^i \left\lceil \frac{D_i}{T_k} \right\rceil.$$

В данном случае сложность линейно зависит от количества дискрет в профиле  $g_i(x)$ . В случае, когда для задания профиля задачи  $W_i$  используется полный дискретный профиль, задаваемый массивом, сложность свертки возрастает до

$$C_{conv} = \frac{1}{2} n_i^2 (\rho_{mul} + \rho_{add}),$$

и общая сложность анализа оценивается сверху формулой

$$\left( \frac{1}{2} n_i^2 (\rho_{mul} + \rho_{add}) + n_i \rho_{add} \right) \sum_{k=1}^i \left\lceil \frac{D_i}{T_k} \right\rceil.$$

В таком случае сложность зависит от количества дискрет в профиле  $g_i(x)$  квадратичным образом.

Объем памяти, необходимый для проведения анализа, можно оценить следующим образом. Для вычисления свертки требуется память для хранения одного исходного дискретного профиля и одного результирующего дискретного профиля, который впоследствии заменяет исходный. Для хранения упрощенных профилей каждой задачи требуется  $M_i$  пар чисел, первое из которых целое, а второе — вещественное. Таким образом, объем памяти, необходимой для анализа  $N$  задач, равен

$$2 z_{real} \max_{i=1, \dots, N} n_i + \sum_{i=1}^N M_i (z_{integer} + z_{real}),$$

где  $z_{integer}$ ,  $z_{real}$  — объем памяти, необходимый для хранения целого и вещественного чисел соответственно.

В случае, когда для задания профиля задачи  $W_i$  используется полный дискретный профиль, задаваемый массивом, необходимый объем памяти возрастает до

$$2 z_{real} \max_{i=1, \dots, N} n_i + \sum_{i=1}^N n_i z_{real}.$$

### Сравнение результатов вероятностного анализа с результатами численного моделирования

С помощью методики вероятностного анализа можно получить оценку снизу для вероятности выполнения задачи в срок. Поскольку методика основана на рассмотрении критического участка, как это принято при анализе СРВ с детерминированными параметрами задач, то вероятность  $p_i$  для каждой задачи рассчитывается исходя из наихудшего числа ее вытеснений более приоритетными задачами. В действительности же наихудшее число вытеснений реализуется не для каждого запроса на решение задачи, поэтому доля запросов, обработанных в срок, оказывается больше значения  $p_i$ . Тем не менее можно ожидать, что дополнения этих значений до единицы окажутся одного порядка.

Для оценки точности методики вероятностного анализа была реализована программа прямого моделирования набора задач, которая в течение длительного времени набирает статистику обработки в срок периодически поступающих в систему запросов. В качестве примера для сравнения был использован набор задач, характеристики которого приведены в табл. 1. Все задачи в этом наборе обладали функциями распределения вида

■ Таблица 1. Набор задач для сравнения

$i$	$T_i$	$D_i$	$c_i^-$	$\bar{c}_i$	$c_i^+$	$u_i^-$	$\bar{u}_i$	$u_i^+$
1	100,0	100,0	10,0	19,9889	100,0	0,1000	0,1999	1,0000
2	150,0	150,0	12,0	23,9867	120,0	0,0800	0,1599	0,8000
3	200,0	200,0	12,0	23,9867	120,0	0,0600	0,1199	0,6000
4	600,0	600,0	20,0	39,9778	200,0	0,0333	0,0666	0,3333
Всего:						0,2733	0,5464	2,7333

$$F(x) = \begin{cases} 0, & x < c^-, \\ \frac{1 - \exp\left(-\frac{x - c^-}{T}\right)}{1 - \exp\left(-\frac{c^+ - c^-}{T}\right)}, & c^- \leq x \leq c^+, \\ 1, & x > c^+. \end{cases}$$

Результаты обеих методик, представленные в табл. 2, позволяют сделать следующие выводы:

- вероятность, полученная с помощью рассматриваемой методики, действительно не больше значения, полученного в результате численного эксперимента;
- вероятность, полученная с помощью рассматриваемой методики, по порядку величины совпадает с долей успешно обработанных запросов, полученной с помощью численного моделирования; наблюдаемые отличия (в особенности у задач 2 и 3) связаны с тем, что анализируется лишь критический участок, из-за чего вероятность занижается;
- вероятность решения в срок задачи с меньшим приоритетом может быть больше вероятности решения в срок задачи с большим приоритетом, как в случае задач 3 и 4; методика вероятностного

■ Таблица 2. Результаты численного эксперимента

$i$	Вероятностный анализ	Прямое моделирование
	$p_i$	Доля запросов, выполненных в срок
1	1,0000000	1,00000000 ± 0,0000000
2	0,9989125	0,99959300 ± 0,0000090
3	0,9954908	0,99898873 ± 0,0000164
4	0,9999913	0,99999588 ± 0,0000015

анализа позволяет замечать подобные особенности.

Таким образом, методика вероятностного анализа действительно позволяет с приемлемой точностью оценить вероятность выполнения задач.

### Заключение

В статье рассмотрена методика вероятностного анализа совокупности периодических задач, выполняемых на одном процессоре в СРВ. С ее помощью можно получить нижнюю оценку для вероятности обработки в срок каждой задачи. Вероятностное описание позволяет расширить область применения традиционных дисциплин планирования на случай, когда максимальная загрузка СРВ становится больше единицы, и методики анализа, развитые ранее для систем с детерминированными параметрами, становятся неприменимы.

Наряду с изложением алгоритма анализа набора задач получена оценка его сложности, а также оценка объема оперативной памяти, требуемой для хранения данных. Сравнение результатов работы алгоритма с численным моделированием показывает, что оценка вероятности оказывается близкой к численному эксперименту.

## Литература

1. **Tia T.-S., Deng Z., Shankar M., Storch M., Sun J., Wu L.-C., Lui J.W.-S.** Probabilistic Performance Guarantee for Real-Time Tasks with Varying Computation Times // Proceedings of the Real-Time Technology and Applications Symposium. — Chicago, Illinois. — May 1995. — IEEE. — P. 164–173.
2. **Lui C.L. and Layland J. W.** Scheduling Algorithms for Multiprogramming in a Hard Real-Time Environment // Journal of the ACM. — 1973. — P. 46–61.
3. **Gardner M.** Probabilistic Analysis and Scheduling of Critical Soft Real-Time Systems // PhD Dissertation. — University of Illinois at Urbana-Champaign. — 1999.
4. **Cervin A.** Analyzing the Effects of Missed Deadlines in Control Systems // Real-Time Graduate Student Conference. — Lund, March 8-9, 2001.  
<http://www.artes.uu.se/events/gskonf01/papers/cervin.pdf>.
5. **Audsley N. C.** Deadline Monotonic Scheduling. YCS.146 // Computer Science, Department, University of York, September 1990.
6. **Audsley N. C., Burns A., Richardson M. F., Wellings A. J.** Hard Real-Time Scheduling: The Deadline-Monotonic Approach Proceedings 8th IEEE Workshop on Real-Time Operating Systems and Software. — Atlanta, USA, 15–17 May 1991.
7. **Lehoczky J., Sha L. and Ding Y.** The Rate-Monotonic Scheduling Algorithm: Exact Characterization and Average Case Behavior // Proceedings IEEE Real-Time Systems Symposium, Santa Monica, California. IEEE Computer Society Press. — 5–7 December 1989. — P. 166–171.
8. **Burns A.** Preemptive Priority Based Scheduling: an Appropriate Engineering Approach // Advances in Real-Time Systems, Prentice Hall. — 1995. — P. 225–248.
9. **Sha L., Rajkumar R. and Lehoczky J.** Priority Inheritance Protocols: An Approach to Real-Time Syn-
- chronization // IEEE Transactions on Computers. — 39(9): 1175–1185. — September 1990.
10. **Sha L., Rajkumar R., Lehoczky J., Ramamritham K.** Mode Change Protocols for Priority-Driven Preemptive Scheduling // CMU/SEI-88-TR-34. — Software Engineering Institute. — November 1988.
11. **Atlas A. K. and Bestavros A.** Statistical rate monotonic scheduling // Proceedings of the 19th Real-Time System Symposium. — December 1998. — P. 123–132.
12. **Hu Z. S., Zhou T. and Sha E. H.-M.** Estimating probabilistic timing performance for real-time embedded systems // IEEE Transactions on Very Large Scale Integration (VLSI) Systems. — 9(6): 833–844. — December 2001.
13. **Audsley N. C., Burns A., Richardson M. F., Tindell K. W. and Wellings A. J.** Applying New Scheduling Theory to Static Priority Pre-emptive Scheduling // Software Engineering Journal. — 8(5): 284–292. — September 1993.
14. **Audsley N. C., Burns A., Davies R. I., Tindell K. W. and Wellings A. J.** Fixed Priority Pre-emptive Scheduling: An Historical Perspective // Journal of Real-Time Systems. — Vol. 8. — 1995. — P. 173–198.
15. **Stankovic J. A., Spuri M., Di Natale M., Butazzo G.** Implications of Classical Scheduling Results for Real-Time Systems // IEEE Computer. — June 1995. — P. 16–25.
16. **Spuri M., Butazzo G.** Scheduling Aperiodic Tasks in Dynamic Priority Systems // Journal of Real-Time Systems. — Vol. 10. — N 2. — 1996. — P. 1979–2012.
17. **Joseph M. and Pandya P.** Finding response times in a real-time system // The Computer Journal — British Computer Society. — 29(5): 390–395. — October 1986.
18. **Bernat G., Colin A., Petters S.** WCET Analysis of Probabilistic Hard Real-Time Systems // The 23rd IEEE International Real-Time Systems Symposium. — December 3–5. — 2002. — Austin, Texas (USA).



## РАСШИРЕНИЯ РЕЛЯЦИОННОЙ МОДЕЛИ ДЛЯ ОБЕСПЕЧЕНИЯ БЕЗОПАСНОСТИ В БАЗАХ ДАННЫХ

**А. В. Фомин,**

аспирант

**А. В. Бржезовский,**

канд. техн. наук, доцент

Санкт-Петербургский государственный университет  
аэрокосмического приборостроения (СПбГУАП)

*В статье предлагаются расширения реляционной модели для обеспечения безопасности при доступе к данным на основе мандатной модели безопасности Белла и ЛаПадула. В отличие от существующих моделей предлагаемая модель позволяет обеспечить ссылочную целостность в базах данных «стандартным» образом. Это достигается за счет введения свойства «Разделение», которое заключается в том, что исходная схема отношения реляционной модели должна быть разделена на две: ключевую схему отношений  $R'$ , которая содержит уникальные первичные ключи, и схему данных  $R''$  — собственно данные исходного отношения. Такой подход позволяет исключить дублирование ключей при явлении многозначной реализации.*

*Relational model extensions for database security are suggested. These models in contrast to existing models allow supplying with reference integrity by standard manner. This arrives by division original relation scheme on two bounded scheme — «key scheme» and «data scheme». In key scheme there are apparent key of original relation. When polyinstantiation occurs in original relation scheme, in key scheme unique apparent keys keep. This allow to provide reference integrity on primary key of key scheme. For suggested concept entity integrity, null integrity, interinstance integrity, polyinstantiation integrity are proposed.*

### Введение

Одним из основных требований международных стандартов безопасности [1, 2] для информационных систем является реализация политики безопасности на основе формальных моделей безопасности. Особенно это требование важно для баз данных, которые являются хранилищем информации, в том числе конфиденциальной. Большинство современных систем управления базами данных позволяют поддерживать только дискреционную [3] и ролевую политики безопасности (ролевую модель можно рассматривать как подвид дискреционного доступа). Дискреционный доступ основан на предоставлении субъектам системы полномочий на выполнение операций над объектами системы (чтение, запись, модификация, удаление). Типичная реализация дискреционного контроля — списки прав доступа. При этом для каждого объекта системы (например, файла) указывается перечень субъектов, которые имеют (или не имеют) право на выполнение определенных операций над этим объек-

том. Но дискреционный контроль не обеспечивает гарантий по защите данных от несанкционированного доступа, поскольку не контролирует потоки информации. В частности он не решает проблему «троянского коня». Эти недостатки отсутствуют при организации мандатного управления доступом [4]. Поэтому для систем высокого класса защищенности требуется поддержка мандатной политики безопасности.

Основой мандатных моделей безопасности является модель Белла—ЛаПадула [4]. Система в модели Белла—ЛаПадула представляется в виде множества субъектов  $S$ , объектов  $O$ ,  $S \subseteq O$ , прав доступа  $read$  и  $write$ . С точки зрения теории реляционных баз данных [5], под объектом может пониматься отношение, атрибут отношения, кортеж или даже отдельный элемент данных в рамках кортежа. Субъект представляет собой некий процесс, действующий от имени пользователя, желающего получить доступ к данным. Уровни безопасности задаются функцией  $F: O \rightarrow L$ , определяющей для каждого объекта и субъекта уровень безопасности из

множества уровней  $L = \{c_1, c_2, \dots, c_n\}$ , разбивая множество сущностей системы на классы, в пределах которых их свойства, с точки зрения модели безопасности, являются эквивалентными. На множестве уровней безопасности определено частичное нестрогое отношение порядка для элементов этого множества, задаваемое оператором доминирования — « $\leq$ ». Запись « $c_1 \leq c_2$ » обозначает, что уровень  $c_2$  доминирует над уровнем  $c_1$ . Множество  $L$  в таком случае еще называют решеткой уровней безопасности. Контроль доступа осуществляется в зависимости от уровней безопасности взаимодействующих сторон на основании двух правил:

1) уполномоченное лицо (субъект) имеет право читать («read») только те документы, уровень безопасности которых не превышает его собственный;

2) уполномоченное лицо (субъект) имеет право заносить информацию («write») только в те документы, уровень безопасности которых не ниже его собственного.

Первое правило обеспечивает защиту информации, обрабатываемой более доверенными лицами, от доступа со стороны менее доверенных. Второе правило предотвращает утечку информации со стороны высокоуровневых участников процесса обработки информации к низкоуровневым.

В классической реляционной теории отсутствуют механизмы, позволяющие представить мандатное управление доступом к данным. Именно на поддержание этого вида доступа направлены описываемые ниже расширения.

### Основные положения и существующие подходы

Реляционная модель данных была предложена Коддом в 1970 г. Согласно этой модели [6], данные хранятся в таблицах, называемых отношениями ( $R$ ). Каждое отношение содержит одну или несколько колонок, называемых атрибутами. В некоторый момент времени отношение содержит в себе множество строк (которое может быть и пустым). Каждая строка называется кортежем. Количество кортежей в отношении меняется с течением времени в результате выполнения операций манипулирования данными, предусмотренных в реляционной алгебре. Набор атрибутов, однозначно идентифицирующих кортеж, называется возможным ключом отношения. Из всех возможных ключей для каждого отношения выбирается только один, который называется первичным ключом. В качестве примера рассмотрим отношение «Проекты» (табл. 1).

Это отношение состоит из трех атрибутов: «Код», «Наименование», «Описание». В отношении опре-

■ Таблица 2. Отношение «Проекты» с метками безопасности

Код	Наименование	Описание
BZM00	Прометей	Строительство казарм
ТП18	Луна	Жилой дом
K678	Роза	Химический завод

делены три кортежа, описывающие проекты некоторой строительной фирмы. Первичным ключом для данного отношения является атрибут «Код». В соответствии с мандатной политикой все данные должны иметь метку, определяющую уровень безопасности данных. Тогда будет возможно реализовать доступ к этим данным пользователей, имеющих определенный уровень безопасности. Например, пользователи, имеющие класс доступа «конфиденциально», могут «увидеть» данные с меткой уровня безопасности «конфиденциально», или ниже («общедоступно»). В связи с этим возникает необходимость в расширении стандартной реляционной модели метками безопасности (в дальнейшем такую модель будем называть расширенной реляционной моделью). Метки безопасности могут быть на разных уровнях детализации — от самих отношений до конкретных элементов данных в отдельном кортеже отношения. Рассмотрим уровень наибольшей детализации, когда доступ осуществляется к конкретным элементам данных (поэлементная классификация). В случае расширенной реляционной модели отношение, представленное в табл. 1, будет выглядеть так, как показано в табл. 2.

В табл. 2 для каждого кортежа определена метка безопасности, которая задает уровень безопасности данных на решетке уровней безопасности. В нашем примере решетка состоит из трех элементов, между которыми существует следующее отношение:  $O \leq K \leq SK$ . Собственно «O» — «общедоступно»; «K» — «конфиденциально»; «СК» — «строгое конфиденциально». Таким образом, «O» — наименее конфиденциальная информация, «СК» — наиболее.

В соответствии с мандатной политикой доступа пользователи, имеющие различные полномочия, должны получить различное «представление» отношения. Например, пользователь, имеющий уровень полномочий «O», должен «увидеть» отношение из табл. 2, «отфильтрованное» в соответствии с его уровнем привилегий (табл. 3). Пользователь с полномочиями на доступ к элементам класса «K» увидит отношение, представленное в табл. 2 так, как это представлено в табл. 4. Пользователь с наивысшими полномочиями увидит все кортежи отношения (табл. 2).

■ Таблица 1. Отношение «Проекты»

Код	Наименование	Описание
BZM00	Прометей	Строительство казарм
ТП18	Луна	Жилой дом
K678	Роза	Химический завод

■ Таблица 3. Представление для пользователя с полномочиями «O»

Код	Наименование	Описание
BZM00	Null	Null
ТП18	Луна	Жилой дом

■ **Таблица 4.** Представление для пользователя с полномочиями «К»

Код	Наименование	Описание
BZM00	О Прометей К	Строительство казарм К
ТП18	О Луна О	Жилой дом О
K678	К Null К	Null К

Предположим, что пользователь с полномочиями «О» хочет вставить в отношении «Проекты» кортеж («K678», «Сакура», «Ресторан»). С точки зрения СУБД, реализующей реляционную модель, данная операция должна быть запрещена, так как кортеж с первичным ключом «K678» уже существует в отношении. Но, с точки зрения политики безопасности, отменить эту операцию нельзя, так как в противном случае возникает нарушение второго правила мандатной политики, когда уполномоченное лицо не может передавать информацию на уровень ниже его собственного. А при отмене вставки кортежа как раз и произойдет передача информации с более высокого уровня на более низкий о том, что кортеж с ключом «K678» уже существует и что информация о нем является конфиденциальной. Стандартным подходом в разрешении этой ситуации является допущение существования обоих кортежей с одинаковыми ключами в отношении. Это явление получило название «polyinstantiation» [7] («многозначность»). В этом случае, с точки зрения пользователя с полномочиями «СК», отношение должно выглядеть так, как представлено в табл. 5.

Следует отметить, что такая же ситуация возникнет и в случае модификации какого-либо отдельного элемента данных. Например, предположим, что пользователь с полномочиями «О» хочет модифицировать кортеж с первичным ключом «BZM00» с указанием наименования проекта — «Волна» и описанием — «Пристань». Требования политики безопасности повлекут создание нового кортежа, причем так, как показано в табл. 6.

Очевидно, что такой подход обладает целым рядом недостатков. Во-первых, нарушается основной принцип уникальности первичного ключа в отношении. Во-вторых, невозможно организовать ссылочную целостность, поскольку отсутствует уникальный первичный ключ в таблице, на которую необходимо сослаться. Кроме того, существует проблема выборки данных с соединением кортежей. Рассмотрим, например, отношение «Сотрудники», которое состоит из атрибутов «Номер паспорта», «Фамилия» и «Проект» (табл. 7). Последний атрибут является «внешним ключом» на отношение «Проекты».

■ **Таблица 5.** Иллюстрация явления «многозначности»

Код	Наименование	Описание
BZM00	О Прометей К	Строительство казарм К
ТП18	О Луна О	Жилой дом О
K678	К Роза СК	Химический завод СК
K678	О Сакура О	Ресторан О

■ **Таблица 6.** Иллюстрация явления «многозначности»

Код	Наименование	Описание
BZM00	О Волна О	Пристань О
BZM00	О Прометей К	Строительство казарм К
ТП18	О Луна О	Жилой дом О

Допустим, что мы можем построить запрос на объединение двух отношений (для упрощения рассматривается запрос на языке SQL):

```
Select * from Сотрудники, Проекты
Where Сотрудники.Проект = Проекты.Код
```

Условие «Сотрудники.Проект = Проекты.Код» включает также равенство соответствующих меток уровня безопасности. Это необходимо для получения корректного результата (иначе мы получим дублирование информации о филиалах). Результатом такого запроса будет только одна строка, по которой сотрудник «Павлов» задействован в проекте «Сакура», а закрытая информация о том, что «Павлов» участвует в проекте «Роза», окажется потерянной даже для пользователей с уровнем привилегий «К» и «СК», имеющих полномочия на доступ к таким данным. По этой причине системы, построенные по принципу многозначности, производят соединение отношений не на уровне СУБД, а возлагают эту операцию на прикладную программу, которая выбирает данные из отдельных таблиц и обрабатывает их для получения корректных результатов [8].

В связи с этим представляется актуальной проблема создания таких расширений реляционной модели, которые бы позволили организовать ссылочную целостность, обеспечить корректную выборку данных с объединением таблиц и минимизировать потери от явления «polyinstantiation». Рассмотрим ее подробнее.

### Расширения реляционной модели для обеспечения безопасности баз данных с организацией ссылочной целостности

Ссылочную целостность можно реализовать при условии существования уникального первичного ключа в отношении, на которое производится ссылка из другого отношения. Но, как уже упоминалось, при наличии явления множественности ключей это невозможно. Единственно возможным первичным ключом в таком отношении могла бы быть совокупность всех атрибутов, образующих «видимый» первичный ключ, и всех меток безопасности всех атрибутов отношения. Но на такой ключ сослаться нет смысла, поскольку в противном случае мы получим

■ **Таблица 7.** Отношения «Сотрудники»

Номер паспорта	Фамилия	Проект
1111111111	О Павлов О	К678 О

проблемы типа «отказа в обслуживании», когда изменить какую-либо запись (относительно атрибутов, не входящих в «видимый» первичный ключ) станет невозможным для разноуровневых пользователей по причине нарушения ссылочной целостности. Но и отказаться от явления «polyinstantiation» полностью также невозможно, поскольку это приводит обычно к нарушению политики безопасности, всевозможным ограничениям на модификацию данных или к блокировке работы системы при определенных условиях [9–11].

Ниже предлагается модель, позволяющая организовать ссылочную целостность при существовании многозначности.

**Определение 1.** Назовем исходной реляционной схемой отношения

$$R(A_1, C_1, A_2, C_2, \dots, A_m, C_m, A_{m+1}, C_{m+1}, \dots, A_n, C_n, TC),$$

где  $A_i$  — атрибут отношения, принимающий значения из домена  $D_i = \text{dom}(A_i)$ ; атрибуты  $A_1, A_2, \dots, A_m$  входят в «видимый» первичный ключ отношения со схемой  $R$  (под «видимым» первичным ключом понимается ключ, определенный пользователем);  $C_i$  — метка уровня безопасности атрибута  $A_i$ ;  $TC$  — метка уровня безопасности кортежа;  $C_i, TC \in L$  ( $L$  — множество уровней безопасности),  $TC = C_{m+1} \cdot C_{m+2} \cdot \dots \cdot C_n$  (оператор « $\cdot$ » обозначает наименьшую верхнюю границу для пары элементов) [4].

**Определение 2.** Внешним ключом в отношении  $R1$ , ссылающемся на отношение  $R2$  (обозначим  $FK_{R2}$ ), называется совокупность метки безопасности внешнего ключа (обозначим  $C_{FK_{R2}}$ ), набора атрибутов отношения  $R1$  и их меток безопасности, являющихся точной копией набора атрибутов и меток безопасности первичного ключа отношения  $R2$ :

$$FK_{R2} = C_{FK_{R2}} \cup \{A_{R1,i}, C_{R1,i}\};$$

$$A_{R1,i} \in FK_{R2}, C_{R1,i} \in FK_{R2} \Rightarrow A_{R1,i} \in R1, C_{R1,i} \in R1,$$

$$\exists A_{R2,j} \in R2, C_{R2,j} \in R2:$$

$$A_{R1,i} = A_{R2,j}, C_{R1,i} = C_{R2,j}, A_{R2,j} \in AK_{R2}.$$

Здесь  $AK_{R2}$  — «видимый» первичный ключ отношения  $R2$  (определение см. ниже). Отношение может иметь несколько внешних ключей, каждый из которых имеет свои метки безопасности и свою совокупность атрибутов отношения; при этом  $FK_{R2} \cap FK_{R3} \cap \dots \cap FK_{Rn} = \emptyset$ .

Определим ключевое свойство модели, которое позволяет организовать ссылочную целостность в базах данных, где поддерживается мандатная политика безопасности (в многоуровневых базах данных).

**Свойство 1.** Разделение. Для организации ссылочной целостности в многоуровневой реляционной модели исходная схема отношения  $R$  должна быть разделена на две схемы —  $R'$  и  $R''$ :

$$R' (A_1, C_1, A_2, C_2, \dots, A_m, C_m);$$

$$R'' (A_{m+1}, C_{m+1}, \dots, A_n, C_n, FK_{R'}, TC).$$

Здесь  $FK_{R'} = C_{FK_{R'}} \cup A_1 \cup C_1 \cup A_2 \cup C_2 \cup \dots \cup A_m \cup C_m$  — внешний ключ из отношения  $R''$  на отноше-

ние  $R'$ .  $C_{FK_{R'}}$  — метка безопасности внешнего ключа.

Если исходная схема отношения  $R$  содержит только атрибуты «первичного» ключа и их метки безопасности, то разделение производить не надо.

Таким образом, в отношении  $R'$  мы получим только определенные пользователем атрибуты первичного ключа отношения  $R$  и их метки безопасности, а в отношении  $R''$  мы получим «информационное наполнение» отношения  $R$  и ссылку на соответствующий кортеж в отношении  $R'$ . Мы не можем представить в отношении  $R'$  только атрибуты «видимого» ключа без меток безопасности, так как в противном случае мы не могли бы скрыть конфиденциальные ключи. Но использование в отношении  $R'$  в качестве первичного ключа совокупности всех атрибутов и их меток безопасности позволит нам организовать ссылочную целостность, поскольку гарантируется уникальность такого ключа.

Рассмотрим пример. Представим отношения, показанные в табл. 6 и 7 в соответствии с предложенной концепцией.

Как видим, в отношении «Проекты'» (табл. 8) мы имеем уникальные ключи, что позволяет организовать нам ссылочную целостность на эти ключи из отношения «Сотрудники''» (табл. 11). В отношении «Сотрудники''» внешним ключом является атрибут «Проект», который состоит из трех элементов. Первые два образуют собственно ссылку на ключ отношения «Проекты'». Третий представляет собой метку безопасности самой ссылки и необходим для корректного объединения при выборке данных из двух таблиц. Обозначим этот элемент  $C_{FK_{\text{Проекты}'}}$ . Для этого примера корректным выражением для выборки данных на языке SQL является следующее:

Select \* from Сотрудники'', Проекты'  
Where Сотрудники''.Проект = Проекты''.Код  
and Сотрудники''.  $C_{FK_{\text{Проекты}'}}$  = Проекты''.ТС

■ Таблица 8. Проекты'

Код	
VZM00	О
ТП18	О

■ Таблица 9. Проекты''

Код	Наименование	Описание	ТС
VZM00	О Волна	О Пристань	О О
VZM00	О Прометей	К Строительство казарм	К К
ТП18	О Луна	О Жилой дом	О О

■ Таблица 10. Сотрудники'

Номер паспорта	
1111111111	О

■ Таблица 11. Сотрудники''

Номер паспорта	Фамилия	Проект	ТС
1111111111	О Павлов	О VZM00	О К К

В том случае, если бы ссылку создавал пользователь с полномочиями «О», мы получили бы в результате выполнения запроса объединение данных о сотруднике с несекретной информацией, по которой сотрудник Павлов участвует в проекте «Волна» по строительству пристани. В данном примере ссылку создавал пользователь с полномочиями «К». В результате выполнения запроса он получит конфиденциальную информацию о том, что Павлов работает над проектом «Прометей» и строит казармы. А пользователь с полномочиями «О» даже не «увидит» эту ссылку.

**Определение 3.** Экземпляры отношения.

1. Пусть  $r''$  — отношение со схемой  $R''$ . Для каждого отношения  $r''$  существует коллекция экземпляров отношения — по одному для каждого уровня безопасности  $c \in L$ . Каждый экземпляр  $r''_c$  отношения  $r''$  представляет собой множество различных кортежей в форме  $(a_1, c_1, a_2, c_2, \dots, a_n, c_n, tc)$ , где  $a_i \in D_i$  или  $a_i = \text{null}$ ;  $a_i \neq \text{null} \Rightarrow c_i \leq c, i = 1, \dots, n$ ;  $c_1, c_2, \dots, c_n \in L$ ;  $tc = c_1 \cdot c_2 \cdot \dots \cdot c_n$ .

Метка уровня безопасности любого атрибута кортежа не может быть равной null, даже если атрибут  $a_i = \text{null}$ . Каждый экземпляр отношения служит для представления реальной сущности на некотором уровне безопасности  $c$ .

2. Пусть  $r'$  — отношение со схемой  $R'$ . Для каждого отношения  $r'$  существует коллекция экземпляров отношения — по одному для каждого уровня  $c \in L$ . Каждый экземпляр  $r'_c$  отношения  $r'$  представляет собой множество различных кортежей в форме  $(a_1, c_1, a_2, c_2, \dots, a_n, c_n)$ , где  $a_i \in D_i, c_i \leq c$ . Каждый экземпляр  $r'_c$  служит для представления набора видимых первичных ключей на некотором уровне безопасности  $c$ .

**Свойство 2.** Видимые первичные ключи. Для схемы отношения  $R'$  «видимым» первичным ключом  $AK_{R'}$  будет совокупность всех атрибутов. При этом для всех экземпляров  $r'_c$  отношения  $r'$  и кортежа  $t \in r'_c$  должны выполняться следующие требования:

- 1)  $t(A_i) \neq \text{null}, i = 1 \dots n$ , где  $n$  — количество атрибутов в отношении;
- 2)  $t(C_i) = t(C_j)$ .

Первое требование вытекает из требований стандартной реляционной модели и гарантирует, что не будет null-атрибутов для первичного ключа (в схему отношения  $R'$  в первичный ключ входят все атрибуты отношения). Второе требование гарантирует, что все атрибуты  $A_i$  схемы  $R'$  будут иметь один уровень безопасности. Это обеспечит видимость всего ключа целиком для некоторого уровня безопасности  $c \in L$ .

**Следствие 1.** Из свойства 2 и определения 2 вытекает, что для внешней ссылки  $FK_{R'}$  из схемы отношения  $R''$  на схему отношения  $R'$  выполняются следующие условия:

- 1)  $A_i \in FK_{R'} \Rightarrow t(A_i) \neq \text{null}$ ;
- 2)  $A_i, A_j \in FK_{R'} \Rightarrow t(C_i) = t(C_j)$ .

**Свойство 3.** Сущностная целостность. Отношение  $r''$  удовлетворяет требованию сущностной целостности, если для всех экземпляров  $r''_c$  отноше-

ния  $r''$  и кортежа  $t \in r''_c$  справедливо:  $A_i \in FK_{R'} \Rightarrow t(C_i) \geq t(C_{FK_{R'}})$ .

Это свойство обеспечивает доминирование атрибутов отношения  $r''$ , не входящих в ссылку на первичный ключ отношения  $r'$ , над атрибутами этой ссылки, что, в свою очередь, исключает возможность существования такого  $r''_c$ , в котором бы была не видна ссылка на отношение  $r'$ .

**Свойство 4.** null-целостность. Отношение  $r''$  удовлетворяет null-целостности, если для каждого экземпляра отношения  $r''_c$  справедливы следующие условия:

1) для всех  $t \in r''_c, t(A_i) = \text{null} \Rightarrow t(C_i) = t(C_{FK_{R'}})$ ;

2) будем говорить, что некоторый кортеж  $t$  поглощает кортеж  $s (t, s \in r''_c)$ , если для каждого атрибута  $A_i$  справедливо, что  $t(A_i, C_i) = s(A_i, C_i)$  или  $t(A_i) \neq \text{null}$ , а  $s(A_i) = \text{null}$ . Второе условие заключается в том, что  $r''_c$  не должно содержать двух различных кортежей, таких, что один поглощает другой.

Первое из этих условий обеспечивает сокрытие от менее привилегированных пользователей информации о возможном существовании данных на более высоких уровнях безопасности. Второе правило исключает «лишние» кортежи.

**Свойство 5.** Межэкземплярная целостность. Отношение  $r''_c$  удовлетворяет свойству межэкземплярной целостности, только если для всех уровней безопасности  $c' \leq c, c' \in L$  справедливо, что  $r''_{c'} = \sigma(r''_c, c')$ , где  $\sigma$  — функция фильтрации. Эта функция получает экземпляр  $r''_{c'}$  из  $r''_c$  по следующим правилам:

1) для каждого кортежа  $t \in r''_c, t(C_{AK}) \leq c'$  существует кортеж  $t' \in r''_{c'}, t'(AK, C_{AK}) = t(AK, C_{AK})$  и для всех  $A_i \in AK$  выполняется:

$$t'(A_i, C_i) = \begin{cases} t(A_i, C_i), & \text{если } t(C_i) \geq c' \\ < \text{null}, t(C_{PK}) >, & \text{если нет;} \end{cases}$$

других кортежей в  $r''_{c'}$  не существует;

2) для получения конечного результата  $r''_{c'}$  приводится в соответствие с требованиями null-целостности.

Свойство межэкземплярной целостности инвариантно по отношению к отношениям со схемами  $R''$  и  $R'$ .

**Свойство 6.** Целостность при множественности экземпляров первичного ключа (polyinstantiation integrity).

1. Схема отношения  $R''$  удовлетворяет целостности при множественности экземпляров первичного ключа, если для всех  $A_i \in R''$  выполняется:  $FK_{R'}, C_i \rightarrow A_i$ . Здесь символ « $\rightarrow$ » обозначает функциональную зависимость значения атрибута  $A_i$  от совокупности  $FK_{R'}, C_i$ . Это свойство определяет первичный ключ многоуровневой схемы отношения  $R''$  —  $PK_{R''}$ .

Первичным ключом в схеме  $R''$  является:

$$FK_{R'} \cup C_{1, R''} \cup C_{2, R''} \cup \dots \cup C_{n, R''}.$$

2. Первичным ключом в схеме  $R'$  является:

$$AK \cup C_{1, R'} \cup C_{2, R'} \cup \dots \cup C_{n, R'}.$$

### Манипулирование данными в расширенной реляционной модели

Рассмотрим основные операции — «добавление данных», «изменение» и «удаление» в предложенной модели. Для наглядности будем использовать вместо реляционной алгебры нотацию языка SQL. При этом примем, что для пользователя, выполняющего операции манипулирования данными, разделение исходного отношения  $r$  со схемой  $R$  на два отношения  $r'$  и  $r''$  (со схемами  $R'$  и  $R''$  соответственно) может быть скрыто. Пользователь с некоторым уровнем полномочий  $c \in L$  будет работать с совокупностью отношений  $r'_c$  и  $r''_c$  как с одним отношением, пользуясь представлением  $r_c: r_c = r'_c : r''_c$ .

**Добавление данных.** Операцию добавления данных пользователем, имеющим полномочия уровня  $c \in L$ , в общем случае можно представить следующим образом:

```
Insert
Into  $r_c[(A_i, A_j, \dots)]$ 
Values ( $a_i, a_j, \dots$ )
```

Здесь квадратные скобки обозначают необязательные элементы, а многоточие — многократное повторение. Только атрибуты данных  $A_i$  могут быть явно определены. Метки безопасности  $C_i$  принимают значение  $c$  — уровня безопасности пользователя, от имени которого производится операция добавления.

Предположим, что существует кортеж  $t$ , такой, что  $t(A_k) = a_k$ , если  $A_k$  включен в список атрибутов, определяемых секцией «Into»;  $t(A_k) = \text{null}$ , если  $A_k$  не включен в список;  $t(C_m) = c$ ,  $m = 1 \dots n$ . В этом случае добавление данных возможно, только если выполняются следующие два условия:

- 1)  $t(AK) \neq \text{null}$ , где  $AK$  — все атрибуты, составляющие видимый первичный ключ кортежа  $t$ ;
- 2) для всех кортежей  $u$ ,  $u \in r'_c$ :  $u(PK) \neq t(PK)$ , где  $r'_c$  — экземпляр отношения  $r'$  для  $c$ -пользователя.

Само же добавление происходит по следующему алгоритму:

- 1) проверка возможности добавления данных;
- 2) добавление кортежа в отношение  $r'$  для хранения ключа новой записи;
- 3) добавление кортежа в отношение  $r''$  и создание для него внешнего ключа на отношение  $r'$ .

**Модификация данных.** Выражение для модификации данных в некотором отношении  $r_c$  имеет следующую форму:

```
Update  $r_c$ 
Set  $A_i = s_i, A_j = s_j, \dots$ 
[Where  $p$ ]
```

Здесь  $s$  — скалярное выражение, определяющее значение некоего атрибута;  $p$  — предикат, определяющий те кортежи, которые подлежат модификации. Предикат может включать условия на метки безопасности в дополнение на условие к атрибутам данных. Но секция «Set» может иметь дело толь-

ко с атрибутами данных. Измененные в результате выполнения запроса атрибуты будут иметь метки безопасности равные метке безопасности пользователя, осуществившего запрос.

В традиционной реляционной модели выполнение запроса на модификацию данных сводится к установлению для соответствующих атрибутов указанных в запросе значений. В расширенной модели процесс модификации отличается, поскольку необходимо предотвратить несанкционированные потоки данных.

Рассмотрим этот процесс. Примем  $S$  — множество кортежей, удовлетворяющих условию предиката  $p$ :

$$S = \{t \in r_c, t \text{ удовлетворяет } p\}.$$

Результат выполнения операции модификации можно представить как сумму результатов от модификации каждого кортежа в  $S$ . Модификацию кортежа  $t$  можно представить как замещение этого кортежа идентичным кортежем  $t'$  за исключением того, что атрибуты, указанные в секции «Set» операции модификации (обозначим их как «SET»), принимают новые значения:

$$r'(A_k, C_k) = \begin{cases} t(A_k, C_k), & A_k \notin \text{SET} \\ \langle s_k, c \rangle, & A_k \in \text{SET}. \end{cases}$$

Для предотвращения утечки информации о модификации кортежа  $t$  к пользователям, имеющим меньший уровень привилегий, необходимо ввести дополнительный кортеж  $t''$ . Такая необходимость возникнет в случае:  $\exists k, t(A_k) \neq \text{null}, t(C_k) < c$ , где  $c$  — уровень полномочий пользователя, производящего модификацию данных. Кортеж  $t''$  при этом должен содержать исходное значение атрибута  $A_k$ :

$$r''(A_k, C_k) = \begin{cases} t(A_k, C_k), & t(C_k) < c \\ \langle \text{null}, t(C_{PK}) \rangle, & t(C_k) = c. \end{cases}$$

Таким образом, каждый кортеж  $t$ ,  $t \in S$  при модификации должен быть заменен кортежем  $t'$  и при необходимости кортежем  $t''$ . Полученный результат должен удовлетворять перечисленным выше ограничениям.

Рассмотрим, как отражается модификация данных «с»-пользователем на представлении отношения для более высокоуровневых пользователей. Здесь возможны две ситуации.

Первая связана с тем, что «с»-пользователь модифицирует некоторый кортеж  $u$ , в котором существует атрибут  $u(A_k) \neq \text{null}, u(C_k) > c$ . В силу межэкземплярной целостности «с»-пользователь видит этот кортеж в отфильтрованном виде, в котором кортежу  $u$  соответствует кортеж  $t$ :  $t \in S, t(A_k) = \text{null}, t(C_k) = c$ . Для предотвращения потери конфиденциальной информации заменить кортеж  $u$  кортежами  $t'$  и  $t''$  так, как описано выше, но уже формируемые с точки зрения  $c' > c$ .

Вторая ситуация также связана со свойством межэкземплярной целостности. Предположим, что в секции «SET» существует атрибут  $A_k$  такой, что

$t(A_k) \neq \text{null}$  и  $t(C_k) = c$ . При модификации этого атрибута его значение в кортеже  $t$  будет заменено на новое —  $s_k$ . Теперь рассмотрим  $r_{c' > c}$ . В силу свойства межэкземплярной целостности может быть несколько кортежей  $u \in r_{c' > c}$ , которые имеют тот же самый первичный ключ, что и кортеж  $t(u(AK, C_{AK}) = t(AK, C_{AK}))$  и то же самое значение атрибута  $A_k$  ( $u(A_k, C_k) = t(A_k, C_k)$ ). Для поддержания межэкземплярной целостности при множественности экземпляров первичного ключа необходимо во всех таких кортежах заменить значение атрибута  $A_k$  на новое —  $s_k$ . Имеем:  $\forall A_k \in \text{SET}, A_k \neq \text{null} \Rightarrow \exists U = \{u \in r_{c' > c}: u(AK, C_{AK}) = t(AK, C_{AK}), u(A_k, C_k) = t(A_k, C_k)\} \Rightarrow$  заменяем кортеж  $u$  на новый кортеж  $u'$ , идентичный кортежу  $u$  за исключением того, что  $u'(A_k, C_k) = \langle s_k, c \rangle$ .

**Удаление данных.** Выражение для удаления данных имеет следующую основную форму:

```
Delete
From  $r_c$ 
[Where  $p$ ]
```

Здесь, как и в предыдущем случае,  $p$  — предикат, позволяющий идентифицировать те кортежи, которые подлежат удалению. В случае расширенной реляционной модели, даже если секция «Where» отсутствует, всегда добавляется одно условие:  $t(TC) = c$ , где  $c$  — уровень полномочий пользователя, от имени которого осуществляется удаление.

Рассмотрим простой пример. Предположим, что пользователь с полномочиями «О» хочет удалить запись о проекте с кодом «ТП18» из отношения «Проекты» (табл. 8, 9). Для этого он выполняет запрос «delete from „Проекты“ where Проекты.Код = „ТП18“». СУБД должна в этом случае удалить кортеж (ТП18 О, Луна О, Жилой дом О) из отношения «Проекты», а затем удалить кортеж (ТП18 О) из отношения «Проекты». Тогда запись о проекте «Луна» будет полностью удалена из отношения «Проекты».

Но при удалении могут возникнуть и более сложные ситуации. Например, пользователь с уровнем полномочий «О» хочет удалить запись о проекте с кодом «BZM00» из отношения «Проекты» (табл. 8, 9). При этом он видит только одну запись (BZM00 О, Волна О, Пристань О). Вторую запись с несекретным ключом «BZM00» он не видит в силу требований null-целостности. Если пользователь удалит эту первую запись, то он не должен увидеть вторую запись. Иначе возникнет канал утечки о существовании скрытой информации о проекте «BZM00». Но эта вторая запись изначально даже не входила в представление для пользователя «О» уровня ( $r'_0$ ).

В этом случае необходимо физически удалить кортеж (BZM00 О, Волна О, Пристань О) из отношения «Проекты». А во втором кортеже с несекретным ключом «BZM00» повысить уровень безопасности ключа на один уровень (т. е. до уровня «К»). С точки зрения политики безопасности, это вполне допустимо, поскольку такая операция будет рассматриваться как запись данных с более низкого уровня на более высокий. Но, с точки зрения ссы-

лочной целостности, это невозможно, пока не создан кортеж «BZM00 К» в отношении «Проекты». Только после создания такой записи и повышения уровня безопасности ключевой связи в отношении «Проекты» можно удалить кортеж «BZM00 О» в отношении «Проекты». Теперь операцию удаления можно считать корректно совершенной.

Если бы кортеж (BZM00 О, Волна О, Пристань О) не существовал изначально в отношении «Проекты», то пользователь уровня секретности «О» видел бы только второй кортеж (BZM00 О, null О, null О). Но просто так удалять такой кортеж тоже нельзя. Иначе будет потеряна конфиденциальная информация (BZM00 О, Прометей К, Строительство казарм К), которая доступна пользователю с уровнем секретности «К». Поэтому в данной ситуации надо создать кортеж «BZM00 К» в отношении «Проекты» и затем повысить уровень секретности атрибута «Код» со значением «BZM00» в отношении «Проекты». Только после этого можно удалить кортеж «BZM00 О» из отношения «Проекты».

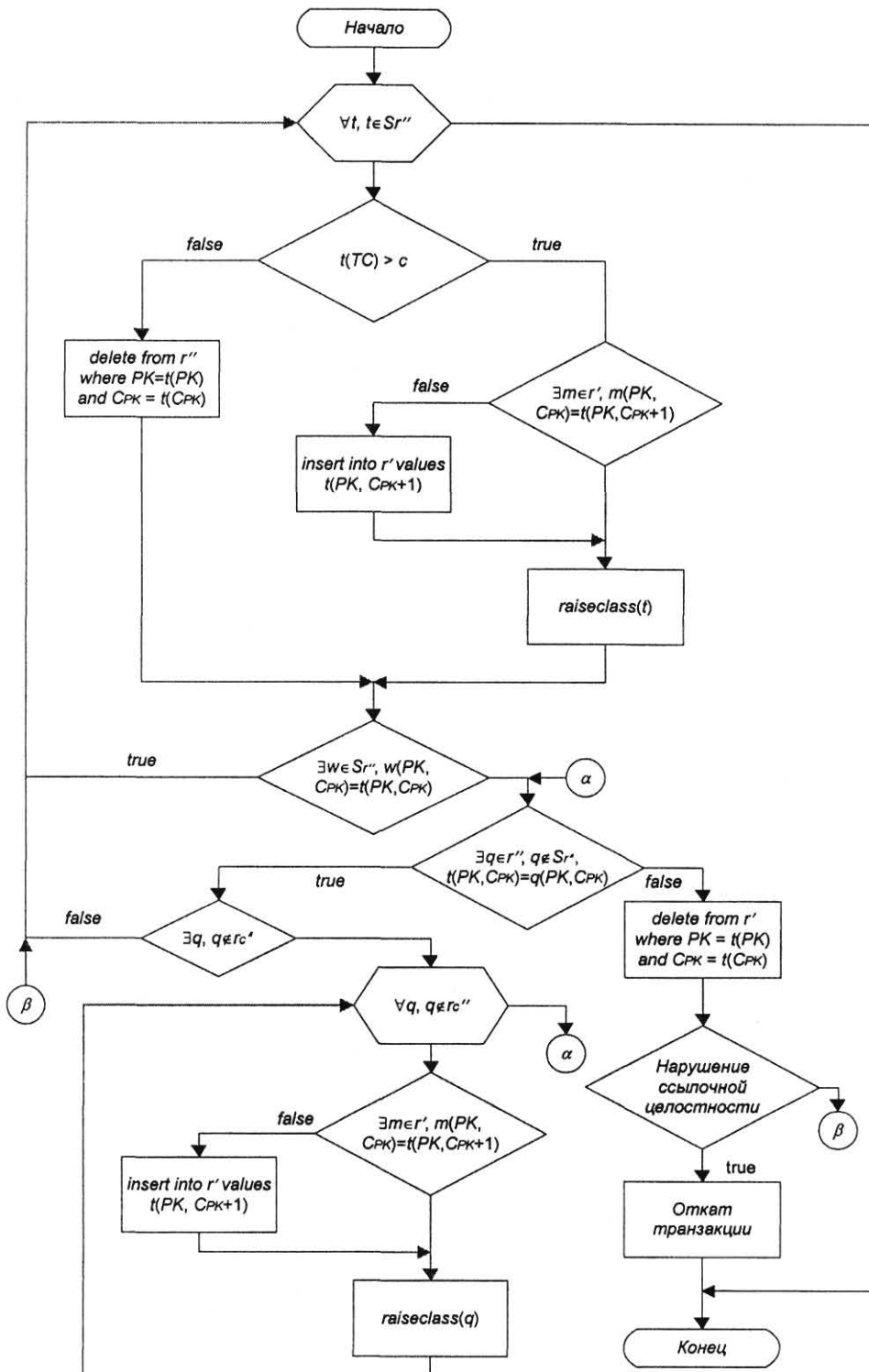
Еще один вариант удаления кортежа из отношения «Проекты»: допустим, что пользователь с полномочиями уровня «К» хочет удалить кортеж (BZM00 О, Прометей К, Строительство казарм К). В этом случае надо лишь физически удалить этот кортеж из отношения «Проекты».

Следует отметить, что если с отношением «Проекты» связаны другие отношения (собственно в нашем примере так и есть — имеется связь с отношением «Сотрудники»), то возникнет ситуация с нарушением ссылочной целостности при попытке удаления кортежа «BZM00 О». В этом случае процесс удаления должен быть отменен, а состояние базы данных должно вернуться в исходное — в то, которое было до начала удаления (откат транзакции).

Формально алгоритм удаления кортежа из отношения  $r$  в рассматриваемых расширениях реляционной модели представлен на рис. 1. Алгоритм учитывает все возможные случаи при удалении кортежа из отношения. В этом алгоритме для каждого кортежа из  $S_{r''} = \{t \in r''_c, t \text{ удовлетворяет } p\}$  выполняются следующие действия, перечисленные ниже.

1. Проверяется метка безопасности удаляемого кортежа. Если она меньше или равна уровню полномочий пользователя, то производится физическое удаление кортежа из отношения  $r''$ . Если нет, то выполняется «алгоритм повышения уровня безопасности кортежа», который заключается не в физическом удалении кортежа из отношения, а в увеличении на один уровень секретности всех меток безопасности всех его атрибутов.

2. Проверяется существование среди удаляемых кортежей таких, в которых «видимый» первичный ключ и его метка безопасности идентичны «видимому» первичному ключу и метке безопасности кортежа, удаленному на шаге 1. Если такие кортежи есть, то они будут обработаны в следующих итерациях, если нет, то проверяется необходимость удаления первичного ключа из отношения  $r'$ .



■ Рис. 1. Алгоритм удаления кортежа из отношения  $r' : r''$

3. Проверяется необходимость удаления ключа из отношения  $r'$ . Она заключается в проверке существования таких кортежей в отношении  $r''$ , которые не входят в число удаляемых кортежей, но

у которых первичный ключ и его метка безопасности идентичны первичному ключу и метке безопасности кортежа, удаляемого на первом шаге. Если таких кортежей нет вообще, значит необходимо уда-



лить первичный ключ из отношения  $r'$ . Если такие кортежи есть, то надо проверить, есть ли необходимость в удалении первичного ключа.

4. Такая необходимость есть только в случае, когда среди найденных на шаге 3 кортежей есть такие, которые не относятся к множеству  $r_c$ . Это те кортежи, которые были вытеснены в силу требования null-целостности из представления отношения для пользователя с уровнем полномочий «с». Для таких кортежей запускается «алгоритм повышения уровня безопасности кортежа». После этого необходимо вернуться на шаг 3 для повторной проверки.

Алгоритм повышения уровня безопасности кортежа применяется для корректного «удаления» кортежей из отношения  $r' : r''$ . Алгоритм состоит из следующих шагов.

1. Проверяется в отношении  $r'$  существование ключа с меткой безопасности на уровень большей, чем метка безопасности обрабатываемого кортежа.

2. Если такого ключа не существует, необходимо создать его.

3. Для обрабатываемого кортежа выполняется функция повышения уровня безопасности кортежа `raiseclass()`.

Функцию `raiseclass` можно представить следующим образом:

$$\begin{aligned} \text{raiseclass: } \forall i, i = 1 \dots n, t(C_i) = c \Rightarrow \\ \Rightarrow t(C_i) = \text{nextclass}(t(C_i)). \end{aligned}$$

Здесь  $n$  — количество атрибутов в схеме  $R$ ;  $t \in r$ ; функция `nextclass()` получает следующий уровень на решетке уровней безопасности.

### Практическое использование

Многие функции, которые описаны в модели, должны возлагаться непосредственно на СУБД (например, функция фильтрации, повышения уровня безопасности, манипулирования данными). В связи с этим возникает вопрос о возможности применения модели в существующих СУБД, в стандартных реляционных базах данных.

Очевидно, что модель представима для существующих реляционных баз данных. Для этого необходимо рассматривать все элементы схем отношений  $R'$ ,  $R''$  (и метки безопасности и атрибуты) как обычные атрибуты стандартной реляционной модели. Но доступ к данным в этом случае необходимо осуществлять только через хранимые процедуры, которые будут реализовывать описанную выше логику вместо СУБД. Доступ непосредственно к базе данных другими путями необходимо запретить. Это вполне можно осуществить средствами распределения полномочий современных СУБД.

Тем не менее разработка модели базы для поддержания предложенной нотации достаточно трудоемка. Необходимо не только продумать, как разделить таблицы, но и описать для каждой таблицы хранимую процедуру доступа к её данным. Поэто-

му для упрощения разработчику должны быть предоставлены специальные средства проектирования данных, которые могут выглядеть как расширения моделей «Сущность—Связь», и соответствующие средства для генерации физической структуры базы данных.

### Заключение

В статье предложены расширения реляционной модели для обеспечения безопасности баз данных в соответствии с мандатной политикой доступа. Эти расширения, в отличие от существующих, позволяют организовать ссылочную целостность между отношениями. Разработаны алгоритмы по манипулированию данными в предлагаемой модели. Дальнейшие разработки будут направлены на создание модели для инфологического проектирования структуры базы данных на основе предложенных расширений.

### Литература

1. **Federal** Criteria for Information Technology Security // National Institute of Standards and Technology & National Security Agency. — Version 1.0, 1992.
2. **Common** Criteria for Information Technology Security Evaluation // National Institute of Standard and Technology & National Security Agency (USA). Communication Security Establishment (Canada). Communications Electronics Security Group (United Kingdom). Bundesamt fur Sichertit in der Informationstechnik (Germany). Service Center de la Securite des Systemes d'Information (France). National Communications Security Agency (Netherlands). — Version 2.1, 1999.
3. **Harrison M., Ruzzo W., Uhlman J.** Protection in operating systems // Communication of the ACM. — 1976. — Vol. 19. — N 8. — P. 461–471.
4. **Bell D. E., LaPadula L. J.** Secure Computer System: Unified Exposition and Multics Interpretation. MTR — 2997 Rev. 2. MITRE Corp., Bedford, Mass, 1976.
5. **Chen P. P-S.** The Entity-Relationship Modell. — Toward a Unified View of Data. ACM TODS 1:1. — March, 1976.
6. **Codd E. F.** A relational model of data for large shared data banks // Communication of the ACM. — 1970. — Vol.13. — N 6. — P. 377–387.
7. **Denning, Lunt T., Schell R.** A Multilevel Relational Data Model // IEEE Symp. Security and Privacy, 1987. — P. 220–234.
8. **Nelson, Doug, Chip Paradise.** Using Polyinstantiation to develop an MLS Application // Proc. IEEE 7<sup>th</sup> Annual Security Application Conf., 1991. — P. 12–22.
9. **Sushil J., Sandhu.** Enforcing Primary Key Requirements in Multilevel Relations // Proc. 4<sup>th</sup> RADC Workshop on Multilevel Database Security. Little Compton, 1991.
10. **Sandhu, Sushil J.** Honest Databases That Can Keep Secrets // Proc. 14<sup>th</sup> NIST-NCSC Nat'l Computer Security Conf. Washington, 1991. — P. 267–282.
11. **Sandhu, Ravi S., Sushil J.** Eliminating Polyinstantiation Securely // Computers and Security. — Vol. 11. — 1992. — P. 547–562.

УДК 621.396.67

# ОСОБЕННОСТИ РАСПРОСТРАНЕНИЯ РАДИОВОЛН МИЛЛИМЕТРОВОГО ДИАПАЗОНА, ПЕРСПЕКТИВЫ ИХ ИСПОЛЬЗОВАНИЯ В СОВРЕМЕННЫХ РАДИОТЕХНИЧЕСКИХ СИСТЕМАХ

**В. Н. Красюк,**

д-р техн. наук, профессор

**О. Ю. Платонов,**

ассистент

**А. Ю. Мельникова,**

ассистент

Санкт-Петербургский государственный университет  
аэрокосмического приборостроения (ГУАП)

*Освоение ММД длин волн обусловлено очевидными преимуществами ММД при использовании его в РТС ближней локации. Несомненный интерес вызывают исследования, связанные с изучением особенностей распространения радиоволн ММД в атмосфере, особенно в сложных метеорологических условиях, в условиях тумана и задымленности. В статье приводятся результаты сравнительного анализа основных параметров РЛС, работающих в длинноволновой и коротковолновой частях ММД длин волн (8 и 3 мм). Выделены несколько перспективных направлений развития РТС, использующих ММД длин волн.*

*Evident advantages of millimeter wavelength range, using in the radio-technical systems of near location, had been condition of millimeter range wide mastering. Investigations associating with research of features of the millimeter radio-waves propagation in atmosphere, especially in complete meteorological conditions, such as a fog and a smoke, excite certain interest. The comparative analysis results of basic characteristics of the radar, operating in the different parts of the millimeter wavelength range (8 mm and 3 mm), has been suggested in this article. Therefore, may be selected some perspective directions of radio-technical systems, using of millimeter wavelength range, development.*

Активное освоение миллиметрового диапазона (ММД) длин волн, наметившееся в последние годы, обусловлено целым рядом факторов, выгодно отличающих эту область радиоволн от сантиметрового и дециметрового диапазонов.

В ММД существенно выше разрешающая способность РЛС по угловым координатам, при этом габариты антенных систем (АС) могут быть значительно уменьшены. Использование радиоволн ММД позволяет, в силу существенной зависимости их коэффициентов отражения и поглощения от природы и структуры отражающей поверхности, определять свойства этой поверхности, включая ее фазовые состояния (вода, лед, снег) и рельеф (степень волнения, тернистости и др.). Таким образом, в ММД длин волн становится возможным неконтактное измерение характеристик отражающих поверхностей. В ММД длин волн мал уровень атмосфер-

ных и промышленных помех. РТС ММД менее восприимчивы к преднамеренным помехам со стороны противника, так как возможность работы в широкой полосе частот позволяет путем использования широкополосной ЧМ и кодирования сигналов повысить помехозащищенность и скорость обработки принимаемой информации. ММД не перегружен, работающие в нем средства связи имеют хорошую электромагнитную совместимость со средствами связи других диапазонов. Радиоволны ММД со сравнительно небольшим затуханием проходят через плазму. На миллиметровых волнах (ММВ) обеспечивается больший доплеровский сдвиг частоты, что существенно облегчает распознавание и классификацию движущихся и маневрирующих целей.

По сравнению с оптическим диапазоном радиоволны ММД также отличаются рядом преимуществ.

Практически не уступая в своей коротковолновой части по разрешающей способности инфракрасному (ИК) излучению, ММВ обеспечивают большую дальность обнаружения. РТС ММД нормально функционируют в условиях тумана, сильной запыленности, густой облачности (например, в тумане с видимостью до 100 м затухание в ИК- и оптическом диапазонах возрастает до 100 дБ/км и более, в то же время затухание ММВ в этих условиях не превышает нескольких дБ/км) [1]. РТС, работающие в ММД, обладают точностью, близкой к точности электронно-оптических средств и в то же время обеспечивают всепогодность и круглосуточность применения, что характерно для РТС, работающих в диапазоне СВЧ. На условия распространения волн ММД не влияет смена суток и сезонов года.

Несмотря на отсутствие резкой границы в закономерностях распространения между сантиметровыми волнами (СМВ) и волнами ММД, некоторые явления, связанные не только с распространением, но и с излучением и поглощением радиоволн в ММД, качественно отличаются от аналогичных явлений, наблюдаемых на более длинных волнах. Это обстоятельство требует нового подхода к анализу работы РТС в ММД радиоволн.

Для ММВ характерно интенсивное поглощение электромагнитной энергии в атмосфере, а также наличие резонансного поглощения кислородом на длинах волн 5,0, 2,5 и 1,7 мм [2]. Таким образом, в ММД атмосфера имеет ряд окон прозрачности и пиков поглощения, средние частоты и длины волн которых представлены в табл. 1 [3]. Там же представлены границы поддиапазонов, соответствующих окнам прозрачности и окнам поглощения, выделенные согласно решению Всемирной административной конференции по радиосвязи.

Радиосвязь, как правило, осуществляется в окнах прозрачности. Больше по сравнению с СМВ поглощение ММВ в гидрометеорах приводит к снижению дальности связи, что требует повышения энергетического потенциала радиолиний для компенсации затухания. В то же время, с точки зрения защищенности передаваемой информации от разведки противника, целесообразно использование окон поглощения. Так, в диапазоне частот в районе 60 ГГц наблюдается самое большое поглощение ММВ кислородом, составляющее 16–17 дБ/км при ясной погоде. Эта особенность по-

■ **Таблица 1.** Границы поддиапазонов, соответствующих окнам прозрачности и окнам поглощения волн ММД в атмосфере

Поддиапазон	Окно прозрачности		Участок поглощения	
	Средняя частота, ГГц	Средняя длина волны, мм	Средняя частота, ГГц	Средняя длина волны, мм
30–51,4	35	8,6	–	–
51,4–66	–	–	60	5
66–105	94	3,2	–	–
105–134	–	–	120	2,5
134–170	140	2,1	–	–
170–190	–	–	180	1,7
190–275	230	1,3	–	–

■ **Таблица 2.** Погонное затухание радиоволн ММД в атмосфере

Погонное поглощение радиоволн ММД в газах атмосферы $\gamma_d$ , дБ/км							
Частота, ГГц	Поглощение в парах воды при плотности, г/м <sup>3</sup>		Поглощение другими газами атмосферы	Суммарное поглощение при влажности, г/м <sup>3</sup>			
	7,5	20,0		7,5	20,0		
10	$5,3 \cdot 10^{-3}$	$7 \cdot 10^{-3}$	$8 \cdot 10^{-3}$	0,013	0,015		
35	0,05	0,13	0,05	0,1	0,18		
94	0,2	0,52	0,045	0,25	0,57		
140	1,0	2,7	0,03	1,0	2,7		
Значения параметров $K_d$ и $\alpha$ для расчета погонного затухания в дожде							
Частота, ГГц	$K_d$ , дБ · ч/м <sup>2</sup>				$\alpha$		
	Поляризация						
	горизонтальная	вертикальная	горизонтальная	вертикальная			
10	0,0101	0,0089	1,28	1,26			
15	0,0367	0,0347	1,15	1,13			
30	0,187	0,167	1,02	1,00			
80	0,975	0,906	0,769	0,769			
90	1,06	0,999	0,753	0,754			
Значение коэффициента $K_T$ при температуре 18 °С, дБ · м <sup>3</sup> /(г · км)							
Длина волны, см	0,2	0,5	0,7	1,0	1,25	2,0	3,0
$K_T$	7,14	1,65	0,876	0,438	0,280	0,112	0,050
Зависимость оптической видимости от влажности тумана							
Влажность $M_T$ , г/м <sup>3</sup>	2		1	0,5	0,2	0,1	
Оптическая видимость, м	30		50	80	200	300	
Классификация типов снега и погонное ослабление $\gamma_d$ на частоте 35 ГГц, дБ/км							
Тип снега	Плотность снега (дождя), г/м <sup>3</sup>	Содержание воды, г/м <sup>3</sup>		$\gamma_d$			
Сухой	0,08	0,006		0,05			
Слегка влажный	0,09	0,008		0,5			
Влажный	0,1	0,01		1,3			
Мокрый	0,27	0,077		2,3			
Обводненный	0,5	0,25		3,5			
Дождь	1,0	1,0		1,1			

зволяет практически исключить возможность обнаружения и перехвата радиосигналов, если станция перехвата расположена лишь на несколько сотен метров дальше от передатчика, чем приемник. Повышенное затухание в пиках поглощения позволяет передавать информацию на соответствующих им частотах при низком уровне взаимных помех от различных служб и организовывать скрытую связь вдоль поверхности Земли на небольших расстояниях. Кроме того, частоты, соответствующие пикам поглощения в атмосфере, могут использоваться на межспутниковых линиях связи большой протяженности за пределами атмосферы. В этом случае атмосфера выполняет роль заграждающего фильтра по отношению к помехам Земли.

В табл. 2 представлены данные о поглощении радиоволн ММД водяным паром, содержащимся в атмосфере [1]. Плотность водяного пара 7,5 г/м<sup>3</sup> считается стандартной для континентальных районов с умеренным климатом, плотность 20,0 г/м<sup>3</sup> ха-

рактерна для нижних слоев тропосферы над океанами и морями субтропической, тропической и экваториальных зон, а также для прибрежных районов суши в этих широтах. Из табл. 2 видно, что поглощение ММВ в газах атмосферы, прежде всего в парах воды, требует заметного повышения потенциала РЛС, работающих в коротковолновой части ММД по сравнению с РЛС сантиметрового диапазона. Например, при дальности действия 5 км энергетический потенциал станции на частоте 140 ГГц в зависимости от условий работы должен на 10–27 дБ превосходить потенциал сантиметровых радиолокаторов.

Поглощение в газах атмосферы является прогнозируемым фактором, причем для расчета полного ослабления на трассе с достаточной точностью можно пользоваться климатическими данными. При этом необходимо оценивать наихудшие условия работы, т. е. ориентироваться на времена года с наибольшей влажностью атмосферы.

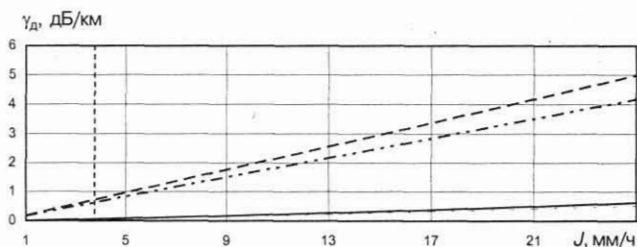
В отличие от поглощения в газах ослабление ММВ, вызываемое осадками, может прогнозироваться только в вероятностном смысле. В настоящее время разработаны методы расчета вероятностных характеристик ослабления радиосигналов на трассах различной протяженности, в том числе на трассах Земля–Космос. Погонное ослабление радиоволн в однородном дожде во всем частотном диапазоне рассчитывается по формуле:

$$\gamma_d = K_d J^\alpha, \quad (1)$$

где  $\gamma_d$  — погонное затухание, дБ/км;  $J$  — интенсивность осадков, мм/ч;  $K_d$  — параметр, зависящий от длины волны, температуры, поляризации, дБ · ч/м<sup>2</sup>;  $\alpha$  — безразмерная величина, значение которой так же, как и  $K_d$ , зависит от длины волны, температуры, поляризации и приводится в табл. 2 [1].

На рис. 1 представлены результаты расчетов погонного затухания радиоволн ММД в осадках различной интенсивности, выполненных в соответствии с (1).

На рис. 1 приняты следующие обозначения: — зависимость погонного затухания от интенсивности дождя  $\gamma_d(J)$  при горизонтальной поляризации радиосигнала на частоте 10 ГГц; - - - - аналогичная зависимость для частоты радиосигнала 30 ГГц; точечной (.....) линией представлена зависимость  $\gamma_d(J)$  при вертикальной поляризации радиосигнала на частоте 10 ГГц; - · - · - · - анало-



■ Рис. 1. Погонное затухание радиоволн в зависимости от интенсивности дождя

гичная зависимость на частоте 30 ГГц. Вертикальными линиями обозначены расчетные интенсивности осадков 1 мм/ч (—) и 4 мм/ч (- - -), которые часто используют при практических оценках затухания радиоволн. Критерием такого выбора является условие, при котором реальное ослабление на трассе не превышает расчетного более 1 % времени в среднем в течение года. При этом интенсивность 1 мм/ч соответствует 99 % обеспеченности работоспособности радиолинии в климатических зонах со среднегодовым количеством осадков 300–500 мм, характерным для большей части Центральной Европы. Интенсивность, равная 4 мм/ч, соответствует такой же обеспеченности для районов со среднегодовым количеством осадков 2000–3000 мм (Юго-Восточная Азия, экваториальные зоны Африки и Латинской Америки, некоторые прибрежные районы субтропиков).

В настоящее время значительное внимание в исследованиях по влиянию осадков на работу радиолиний уделяется вопросам учета неоднородности дождя на трассе распространения, что особенно важно при расчете радиорелейных линий и линий Земля–Космос. Кроме того, при анализе работы радиолокационных систем ММД обычно можно ограничиваться предположением об однородности дождя в пределах дальности действия РЛС. Основанием для этого служит следующее. Как видно из рис. 1, даже при слабых дождях создаваемое ими ослабление достаточно велико. Оценки показывают, что при современном энергетическом потенциале РЛС ММД их дальность действия в таких осадках не превышает 5–10 км. В сильных дождях с интенсивностью более 10 мм/ч во всем ММД погонное ослабление превосходит 5–7 дБ/км, и в этом случае дальность действия РЛС не превышает 1–3 км. Следовательно, предполагая дождь однородным в пределах всей дальности действия РЛС, незначительно завышается суммарное ослабление сигнала.

Ослабление радиоволн ММД в тумане зависит от количества жидкой воды в единице его объема, т.е. водности этого гидрометеорологического образования, а также от его температуры. Величина погонного ослабления в этом случае определяется по формуле

$$\gamma_d = K_T M_T,$$

где  $M_T$  — водность, г/м<sup>3</sup>;  $K_T$  — удельный погонный коэффициент ослабления, дБ · м<sup>3</sup>/(г · км), значения которого при температуре 18 °С представлены в табл. 2.

Следует отметить, что коэффициент  $K_T$  имеет сильную температурную зависимость: на длине волны 5 см он изменяется примерно в три раза в интервале температур от 0 до 40 °С, убывая с увеличением температуры. Водность тумана ориентировочно можно определить по оптической видимости в нем, пользуясь данными, также приведенными в табл. 2. Вероятность появления туманов в равнинной местности в холодное время года составляет 3–5 %, а в теплые месяцы — 0,6–2,0 %, причем

приземные туманы могут захватывать большие районы протяженностью 100–1000 км. Высока вероятность возникновения туманов над океанами и морями в высоких широтах, где она может достигать 40 %.

Многообразие форм выпадающего снега и сильная зависимость поглощения радиоволн ММД от влажности снега затрудняют разработку теоретических моделей или подбор эмпирических зависимостей. Классификация типов выпадающего снега и погонное ослабление им радиоволн на частоте 35 ГГц представлены в табл. 2 [1]. Из данных таблицы видно, что в сухом снегу ослабление составляет менее 0,05 дБ/км, т. е. сравнимо с поглощением радиоволн ММД в газах атмосферы, в то время как во влажном снегу ослабление составляет 2,3–3,5 дБ/км, что превосходит затухание радиоволн в дожде той же интенсивности.

Радиоволны ММД с небольшим затуханием проходят через плазму, поэтому используются для связи с ракетами, проходящими через ионизированную атмосферу. В [4] приведены результаты расчета потерь энергии электромагнитных волн за счет поглощения и однократного отражения от границы плазма — воздух. Расчеты приведены для случая нормального прохождения через слой плазмы толщиной 10 см с параметрами: концентрацией электронов  $N_e = 2,8 \cdot 10^{12}$  1/см<sup>3</sup> и частотой соударений  $5,0 \cdot 10^9$  1/с для следующих длин волн: 20 м, 4,0 м, 0,5 м, 0,05 м и 0,008 м. Результаты расчетов приведены в табл. 3. В указанной таблице приняты следующие обозначения:  $\lambda$  — длина волны, м;  $f$  — частота радиосигнала, МГц;  $\alpha$  — погонное затухание радиоволн в плазме, дБ/м;  $L_{\text{погл.}}$  — коэффициент поглощения радиоволн в плазме, дБ;  $L_{\text{отр.}}$  — коэффициент отражения в плазме, дБ;  $L$  — коэффициент суммарных потерь дБ.

Из табл. 3 видно, что потери радиоволн при прохождении через плазму зависят от частоты сигнала. При заданных параметрах плазмы потери, практически исключают возможность прохождения радиоволн, наблюдаются в диапазоне дециметровых и сантиметровых волн. В то же время в ММД суммарные потери сравнительно невелики. Наибольшие потери на всех указанных длинах волн вносит поглощение; сравнительно небольшие потери имеют место за счет отражения радиоволн плазмой.

Перечисленные особенности распространения радиоволн ММД позволяют сделать вывод о перспективах их использования в РТС самого разного назначения. В связи с этим можно выделить не-

сколько основных направлений развития РТС ММД. Несомненны преимущества применения ММД в системах ближней локации с высокой разрешающей способностью (вплоть до радиовидения на волнах с длинами 2–3 мм). Эффективным представляется применение ММВ в системах обнаружения и обигания препятствий особенно в условиях плохой видимости (сильная задымленность, туман, облачность), при полетах ЛА на малой высоте с большой скоростью и т. п. Перспективной является возможность осуществления дистанционного мониторинга земной поверхности при наблюдении с ЛА с целью получения информации о влажности почвы, растительном покрове, масляных (нефтяных) пятнах на водной поверхности, состоянии снежного и ледового покрова и т.п. [5]. Нельзя не упомянуть о возможности применения радиоволн ММД в качестве эффективного не летального (ожогового) оружия.

При использовании радиоволн ММД в РТС, на первый взгляд, очевидным преимуществом обладает диапазон волн со средней длиной волны 8 мм (35 ГГц), обеспечивающий наименьшее затухание в атмосфере (см. табл.1). Однако, если исходить из одинаковых массогабаритных характеристик антенны, в диапазоне длин волн, близких к 3 мм, обеспечивается существенно больший коэффициент усиления антенны, большая эффективная величина рассеяния (ЭПР) отражающего объекта, в несколько раз большая разрешающая способность. Кроме того, в ММД вблизи длины волны 3 мм при одной и той же относительной полосе пропускания можно получить большую абсолютную полосу усиливаемых частот. Участок ММД со средней длиной волны 3 мм, уступая более длинноволновым участкам ММД по характеристикам, связанным с потерями в атмосфере (см. рис. 1, табл.2), по ряду других параметров, определяющих дальность, имеет важные преимущества по сравнению с участком ММД вблизи окна прозрачности, соответствующего 8 мм.

Проведем сравнительную количественную оценку некоторых основных радиолокационных параметров в ММД длин волн.

1. При одинаковом эффективном диаметре антенны  $D$  коэффициент усиления  $G \sim (D/\lambda)^2$  при  $\lambda \approx 3$  мм почти на порядок больше, чем на участке ММД, соответствующем 8 мм [5].

2. Для плоской отражающей поверхности при угле падения сторонней волны  $\phi > \pi/4$  ЭПР  $\sigma(\phi)$  пропорциональна  $1/\lambda^2$ . При скользющем угле падения и вертикальной поляризации падающей волны ЭПР пропорциональна  $1/\lambda^4$ . Это позволяет получить на длине волны 3 мм значение ЭПР на порядок больше по сравнению с ЭПР, полученной на длине волны 8 мм.

ЭПР прямоугольной пластины с размерами  $a, b$  можно определить из следующего выражения:

$$\sigma(\phi) = \frac{4\pi a^2 b^2}{\lambda^2} \left[ \frac{\sin(ka \sin(\phi))}{ka \sin(\phi)} \right], \quad (2)$$

■ Таблица 3. Потери энергии электромагнитных волн за счет поглощения и однократного отражения от границы плазма–воздух

$\lambda$ , м	$f$ , МГц	$\alpha$ , дБ/м	$L_{\text{погл.}}$ , дБ	$L_{\text{отр.}}$ , дБ	$L$ , дБ
20,0	15,0	$2,5 \cdot 10^2$	25,0	19	44,0
4,0	75,0	$4,2 \cdot 10^2$	42,0	14	56,0
0,5	600	$1,67 \cdot 10^3$	167,0	13	180,0
0,05	6000	$3,4 \cdot 10^3$	340,0	13	353,0
0,008	37 500	$1,7 \cdot 10^1$	1,7	0	1,7

где  $k$  — волновое число;  $\phi$  — угол падения сторонней волны, отсчитываемый в азимутальной плоскости относительно нормали к поверхности пластины.

Результаты расчетов, выполненные в соответствии с (2) для длин волн 8 и 3 мм, представлены на рис. 2. Сплошной линией на рисунке обозначена диаграмма рассеяния (ДР) в азимутальной плоскости от прямоугольной отражающей пластины при падении сторонней волны с длиной 8 мм; штриховой линией — ДР в азимутальной плоскости такой же пластины при падении волны с длиной 3 мм.

3. Доплеровский сдвиг  $F_d$  частоты при скорости цели 30 км/ч (8,333 м/с) может быть рассчитан по формуле [6]

$$F_d = f_0 \left[ \frac{1 - V_r/c}{1 + V_r/c} \right],$$

где  $V_r$  — радиальная скорость цели;  $f_0$  — частота падающей волны;  $c$  — скорость света.

Результаты расчета доплеровского сдвига частоты представлены в табл. 4.

Факторы, перечисленные в пп. 1–3, компенсируют уменьшение дальности на участке ММД вблизи длины волны 3 мм, вызванное увеличением удельных потерь на 3–4 дБ при сложных метеорологических условиях.

4. Разрешающая способность по дальности  $\Delta R$  обеспечивается в основном параметрами зондирующего импульса и, казалось бы, от несущей частоты радиосигнала не зависит. Однако при внутриимпульсной ЛЧМ она определяется следующим образом:

$$\Delta R = \frac{c}{2 \Delta F},$$

где  $c$  — скорость света;  $\Delta F$  — абсолютная девиация ЛЧМ-сигнала.

Очевидно, что при  $\lambda = 3$  мм большее значение  $\Delta R$  достигается при меньшей относительной девиации частоты, чем при  $\lambda = 8$  мм. Так, при  $\Delta F_{\text{отн}} = 0,5\%$  на несущей частоте 94 ГГц ( $\Delta F = 400$  МГц) реализуется  $\Delta R = 37,5$  см, в то время как при таком же значении  $\Delta F_{\text{отн}}$  на несущей частоте 35 ГГц может быть получено значение  $\Delta R = 85,7$  см.

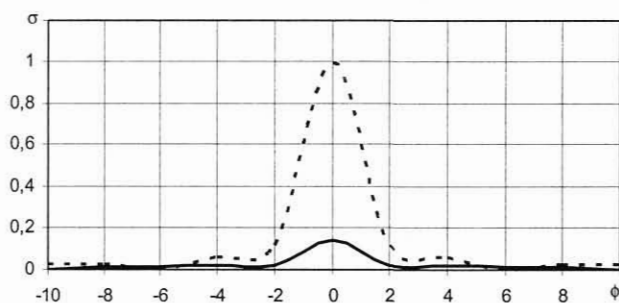


Рис. 2. Диаграмма рассеяния прямоугольной пластины, ориентированной ортогонально направлению падения волны

Таблица 4. Зависимость доплеровского сдвига частоты от длины падающей радиоволны

$\lambda$ , м	0,03	0,008	0,003	0,002
$F_d$ , Гц	$1 \cdot 10^{10}$	$3,75 \cdot 10^{10}$	$1 \cdot 10^{11}$	$1,5 \cdot 10^{11}$

5. Разрешающая способность по азимуту определяется главным образом шириной диаграммы направленности антенны  $\Delta\Phi$ , которая, в свою очередь, зависит от отношения эффективного диаметра антенны  $D$  к длине волны  $\lambda$ . Преимущество трехмиллиметрового диапазона перед восьмимиллиметровым при одинаковом значении  $D$  в этом случае очевидно. При одинаковом значении  $\Delta\Phi$  в трехмиллиметровом диапазоне достигается значительное (приблизительно в 3 раза) уменьшение  $D$ .

Основные параметры РЛС ММД, работающих на длинах волн 8 и 3 мм, приведены в табл. 5.

При проводимой сравнительной оценке следует принять во внимание параметры, характеризующие мощность излучения и шумы приемных устройств. В этих случаях, безусловно, участок ММД со средней длиной волны 8 мм имеет преимущество.

При переходе к окну прозрачности в атмосфере, соответствующему средней длине волны 2 мм, выигрыш в значениях  $G$  и  $\sigma$  составляет лишь 2,25 раза, тогда как увеличение потерь в сложных метеословиях составляет примерно 6–8 дБ, не считая значительного затухания в волноводном тракте. Следовательно, переход к еще более коротковолновому участку ММД не оправдан.

Рассмотрев особенности распространения волн ММД и сравнив РТС ММД с оптическими системами и РТС, работающими на СВЧ, можно сделать следующий вывод: эффективность использования РТС ММД и выбор рабочей частоты определяются

Таблица 5. Основные параметры РЛС ММД, работающих на длинах волн 8 и 3 мм [5]

Параметр	Длина волны, мм	
	8	3
Диаметр антенны, м, при одинаковом отношении $D/\lambda$	2,0	0,75
Диаметр антенны, м, при ширине ДН по уровню половинной мощности 0,5 град	1,9	0,5
Относительная ЭПР (при одинаковой геометрической площади отражения)	1,0	7,8
Поперечная разрешающая способность, мин	22 ( $D/\lambda \sim 225$ )	8 ( $D/\lambda \sim 150$ )
Разрешающая способность по дальности, м	0,86	0,38
Доплеровский сдвиг частоты, ГГц	37,5	100,0
Массогабаритные характеристики СВЧ-части, отн. ед.	1,0	0,25

не только конкретными задачами, но и условиями применения.

Как правило, функционирование РТС ММД осуществляется на длинах волн, соответствующих окнам прозрачности, т. е. 8,6 и 3,2 мм. При сильной запыленности местности, в пустынных районах Африки, Средней Азии и Ближнего Востока, более эффективным представляется использование длинноволновой области ММД, поскольку на более коротких волнах, тем более в ИК- и оптическом диапазонах, велико рассеяние электромагнитного излучения на частицах пыли, песка и т. п. По этим же причинам затруднено использование волн ИК и оптического диапазонов в условиях сильной задымленности местности, где предпочтительно применение ММД. Вследствие существенного поглощения волн ММД водяным паром, содержащимся в атмосфере, и осадками затруднено применение волн ММД над поверхностью океанов и морей а также над прибрежными районами субтропических, тропических и экваториальных зон Юго-Восточной Азии, Африки и Латинской Америки. В этих районах возможно использование длинноволновой области ММД, а также коротковолновой части диапазона СВЧ. В районах с сухим континентальным климатом либо в районах, характеризующихся умеренной интенсивностью осадков (Центральная Европа, Сибирь и т. п.), в системах ближней локации различного назначения предпочтительно использование РТС, работающих в коротковолновой области ММД. В этой же области ММД представляется эффективным функционирование средств обеспечения ледовой разведки при навигации судов в условиях Заполярья, Арктики и Антарктиды.

Для защиты передаваемой информации целесообразно использовать окна поглощения. Это позволяет передавать информацию при низком уровне взаимных помех и организовывать скрытую связь на небольших расстояниях. Кроме того, частоты, соответствующие пикам поглощения в атмосфере, могут использоваться на межспутниковых линиях связи большой протяженности. В этом случае атмосфера выполняет роль заграждающего фильтра по отношению к помехам Земли.

Работа выполнена в рамках гранта Т02-02.5-3650 Минобразования РФ (конкурс 2002 г.) по фундаментальным исследованиям в области технических наук.

### Литература

1. **Кулемин Г. П., Разказовский В. Б.** Рассеяние миллиметровых радиоволн поверхностью Земли под малыми углами. — Киев: Наук. думка, 1987. — 232 с.
2. **Белоцерковский Г. Б.** Антенны. — М.: Оборонгиз, 1962. — 491 с.
3. **Гассанов Л. Г., Липатов А. А., Марков В. В., Могильченко Н. А.** Твердотельные устройства СВЧ в технике связи. — М.: Радио и Связь, 1988. — 288 с.: ил.
4. **Красюк В. Н., Михайлов В. Ф.** Бортовые антенны гиперзвуковых летательных аппаратов: Учебное пособие. — СПб.: СПбГААП, 1994. — 216 с.
5. **Гельвич Э. А.** Преимущества использования 3-мм диапазона длин волн для обеспечения безопасности полетов и судовождения // Электронная техника. Сер. СВЧ-техника. — Вып. 1(467). — 1996. — с. 55-57.
6. **Финкельштейн М. И.** Основы радиолокации: Учебник для вузов. — М.: Радио и Связь, 1983. — 535 с.

УДК 621.396.67

# ОСОБЕННОСТИ ПРИМЕНЕНИЯ МИКРОПОЛОСКОВЫХ АНТЕНН НА ЛЕТАТЕЛЬНЫХ АППАРАТАХ ИЗ КОМПОЗИТОВ С МАЛЫМИ ХАРАКТЕРИСТИКАМИ РАССЕЯНИЯ

**В. Н. Красюк,**

д-р техн. наук, профессор

**В. В. Горбацкий,**

ассистент

Санкт-Петербургский государственный университет  
аэрокосмического приборостроения (ГУАП)

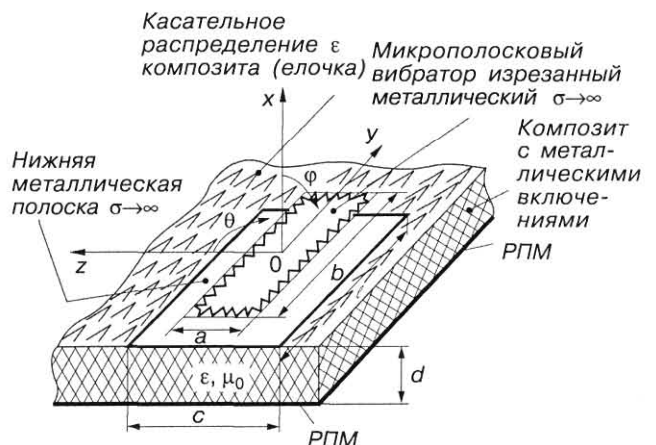
На примере расчета характеристик излучения ленточного микрополоскового вибратора и экспериментальных исследований описаны изменения в его конструктивном исполнении, которые дают возможность использовать его с композиционными материалами и уменьшить величину рассеяния на 8–13 дБ по сравнению с обычным полуволновым вибратором.

On an example of calculation of radiation characteristics of the tape microstrip vibrator and experimental researches the changes in its design are adduced which enable to use it with composite materials and to reduce value of dissipation on 8–13 dB as contrasted to by normal half-wave dipole.

Возросшая потребность в легких низкосилуэтных конформных и недорогих антеннах, которые можно размещать на самолетах, выполненных из композитов с малыми рассеивающими свойствами, определила в технике антенн СВЧ создание микрополосковых антенн с малой эффективной площадью рассеяния (ЭПР) на подложках из композиционных материалов (рис. 1). Микрополосковые антенны (МПА) обеспечивают высокую повторяемость размеров, низкую стоимость, малые металлоемкость, габаритные размеры, массу, рассеяние от сторонних РЛС [3, 6]. Для сравнения: турникетная антенна на полуволновых вибраторах с рефлектором имеет массу 200–300 г, аналогичная МПА с теми же электродинамическими характеристиками — на порядок меньше (приблизительно 30 г). Не менее показательны и габаритные параметры МПА, которые являются двумерными по своей конструкции, например, стандартный полуволновой микрополосковый излучатель занимает площадь примерно  $0,25(\lambda \cdot \lambda)/\epsilon$  ( $\lambda$  — длина волны,  $\epsilon$  — относительная диэлектрическая проницаемость подложки). Площадь, занимаемая схемой питания (в плоскостном исполнении), равна  $\approx 0,04(\lambda \cdot \lambda)/\epsilon$ . Фазовращатель в фазированной антенной решетке (ФАР) обычно занимает площадь  $0,1(\lambda \cdot \lambda)/\epsilon$ .

МПА способны излучать поле с линейной, круговой и эллиптической поляризацией, допускают удоб-

ные конструктивные решения для обеспечения работы в двух- или многочастотных режимах, легко позволяют объединить многие элементарные излучатели в ФАР и разместить их на поверхностях сложной формы. Кроме того, МПА обладают высокими аэродинамическими, механическими и тем-



■ Рис. 1. Ленточный микрополосковый вибратор с конструктивными изменениями при использовании композита и получения малых характеристик рассеяния



пературными характеристиками. Они также могут быть использованы на самолетах, выполненных из композитов с малыми характеристиками рассеяния. Исследование этого и является основной задачей данной статьи.

Эффективность излучения и простота выполнения схемы возбуждения ставит полосковый вибратор на одно из первых мест по использованию при построении малогабаритных антенных систем. Вибраторы применяют и в качестве элементов микрополосковых антенных решеток. Известно большое число конкретных модификаций микрополосковых вибраторов (МПВ), однако для большинства из них излучающим элементом является отрезок открытой микрополосковой линии (МПЛ) — ленточный вибратор.

Проанализируем электрические характеристики ленточного вибратора, который можно рассматривать как модельную конструкцию для многих разновидностей МПВ.

Рассмотрим излучение МПВ (см. рис. 1) в виде отрезка МПЛ длиной  $b$ , шириной  $a$ , расположенного на слое экранированного  $s$  и  $l$  диэлектрика

$$J(y) = a_y \varphi(y)/a = a_y \cos(\beta y/b)/a, \quad (1)$$

где  $a_y$  — единичный вектор;  $\beta$  — постоянная распространения волны в МПЛ, отрезок которой образует МПВ.

Выражение (1) можно рассматривать как первое приближение для функции, описывающей истинное распределение тока на поверхности вибратора [1–4].

Диаграммы направленности (ДН) МПА могут быть определены стандартным методом: по известной силе тока на излучателе (1) рассчитывают электромагнитные поля, затем, если это удастся, производят разделение полей на распространяющиеся и затухающие, для первых устанавливается зависимость от угловых координат, т. е. ДН. Вычисление полей в дальней зоне ( $R \gg \lambda$ ) производится обычно с использованием методов перевала или стационарной фазы. Здесь не рассматривается техника этих вычислений. Применения этого метода к родственным задачам можно найти, например, в работах [1–5].

Пользуясь методиками расчета характеристик полосковых антенн, приведенными в работах [1–4], запишем выражения для расчета диаграмм направленности с учетом зависимости  $\varepsilon = f(y)$  (наличие композита):

В плоскости  $E$  ( $\theta = \pi/2$ ;  $xoy$ ):

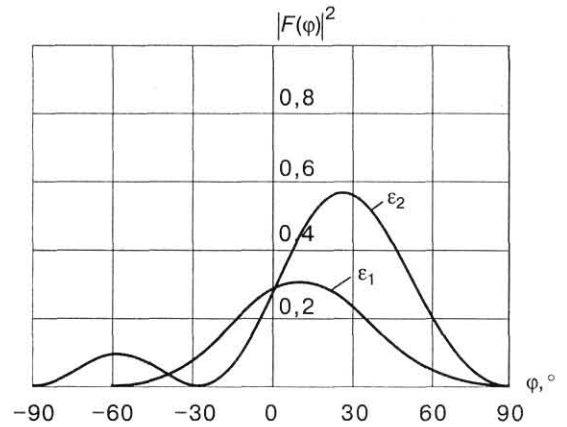
$$|F(\varphi)|^2 = \left( \beta \frac{\cos(0,5k_0 b \sin \varphi) - \cos(0,5\beta b)^2}{[\beta^2 - (k_0^2 \sin \varphi)^2] \sin(0,5\beta b)} \right)^2 \times \frac{(\xi \cos \varphi)^2}{\xi^2 + (\varepsilon \cos \varphi \operatorname{ctg} \xi k_0 d)^2}, \quad (2)$$

$\xi = \sqrt{\varepsilon - \sin^2 \varphi}$ ,  $\varepsilon = \frac{\varepsilon_a}{\varepsilon_0}$ ,  $k_0 = \frac{2\pi}{\lambda}$ ,  $\varepsilon = \text{const}$  по  $z$ ,  $\varepsilon = f(y)$  («елочка»).

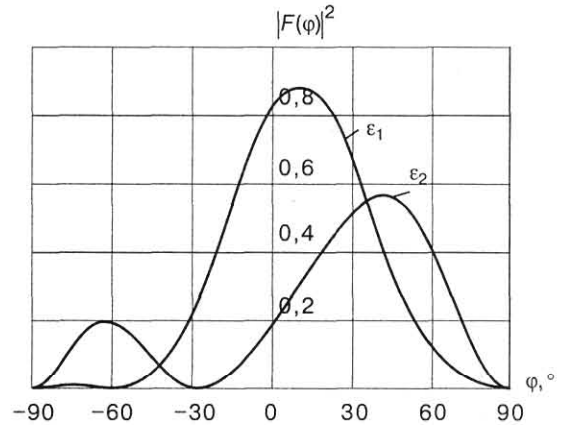
В плоскости  $H$  ( $\varphi = 0$ ;  $zox$ ):

$$|F(\theta)|^2 = \beta^2 \left( \frac{\sin(0,5k_0 a \cos \theta)}{0,5k_0 a \cos \theta} \right)^2 \times \frac{\sin^2 \theta}{\sin^2 \theta + (\xi \operatorname{ctg} \xi k_0 d)^2}, \quad (3)$$

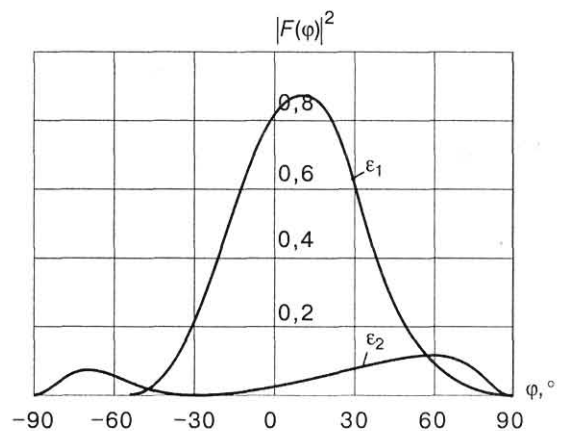
$$\xi = \sqrt{\varepsilon - \cos^2 \theta}, \quad \varepsilon = \text{const} \text{ по } z \text{ и } y.$$



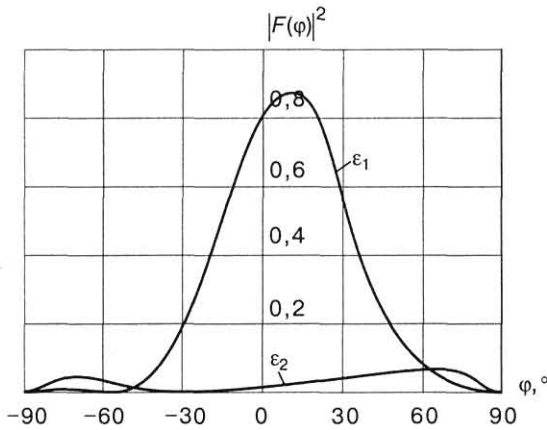
■ Рис. 2. ДН ленточного МПВ в  $E$ -плоскости при  $d = 0,1$  см;  $\varepsilon_1 = 2,54$ ;  $\varepsilon_2 = 5$



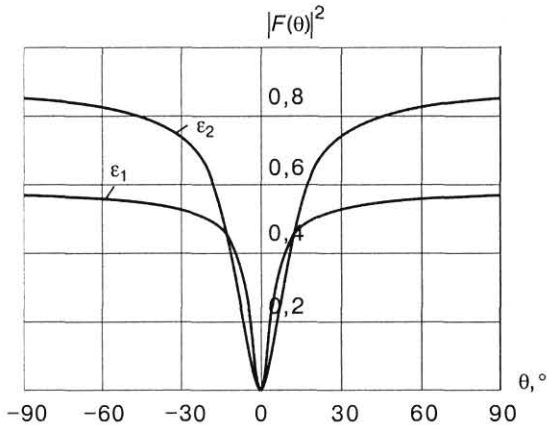
■ Рис. 3. ДН ленточного МПВ в  $E$ -плоскости при  $d = 0,5$  см;  $\varepsilon_1 = 2,54$ ;  $\varepsilon_2 = 5$



■ Рис. 4. ДН ленточного МПВ в  $E$ -плоскости при  $d = 1,5$  см;  $\varepsilon_1 = 2,54$ ;  $\varepsilon_2 = 5$



■ Рис. 5. ДН ленточного МПВ в *E*-плоскости при  $d = 2$  см;  $\epsilon_1 = 2,54$ ;  $\epsilon_2 = 5$



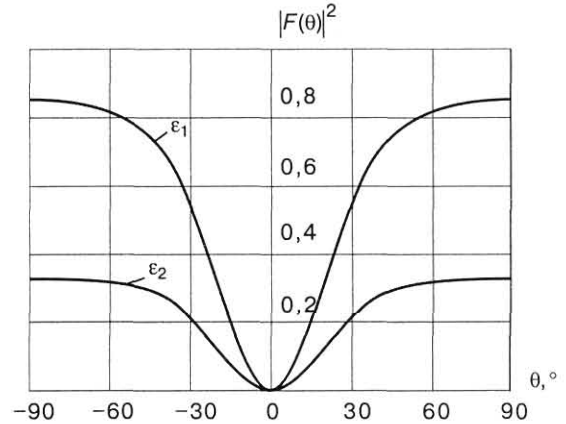
■ Рис. 6. ДН ленточного МПВ в *H*-плоскости при  $d = 0,1$  см;  $\epsilon_1 = 2,54$ ;  $\epsilon_2 = 5$

На рис. 2–9 приведены диаграммы направленности ленточного микрополоскового вибратора (ЛМПВ) в *E*- и *H*-плоскостях; диаграммы рассчитаны по приведенным формулам при следующих данных:  $\lambda = 3$  см;  $a = 0,5$  см;  $b = 1,5$  см;  $d_1 = 0,1$  см;  $d_2 = 0,5$  см;  $d_3 = 1,5$  см;  $d_4 = 2$  см;  $\epsilon_1 = 2,54$ ;  $\epsilon_2 = 5$ ;  $\epsilon = e^{0,07y}$  по направлению оси  $y$ .

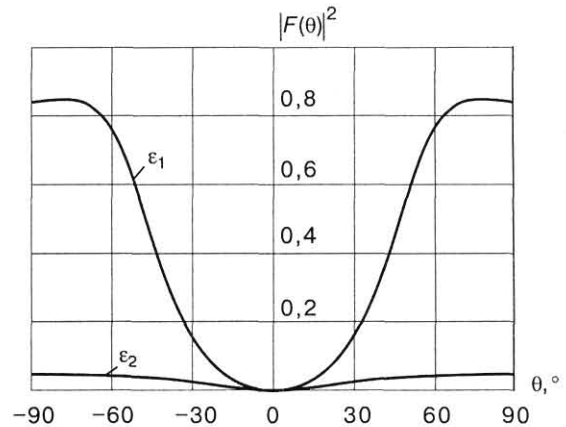
Как видно из рис. 2–5, ДН ЛМПВ в *E*-плоскости имеет ярко выраженный максимум. Причем при  $d = 0,1$  см значение максимума ДН при  $\epsilon = 5$  почти в два раза превышает максимальное значение ДН при  $\epsilon = 2,54$ ,  $\epsilon = f(y)$ . Такая же картина наблюдается и в *H*-плоскости (см. рис. 6), при аналогичных значениях диэлектрической постоянной и толщины диэлектрического композита  $\epsilon \neq f(y)$ . Очевидно, что для данных размеров ЛМПВ при  $\epsilon = 5$  и  $d = 0,1$  см волна, излученная вибратором в композит, отразившись от нижней проводящей полоски ( $c > a$ ,  $l > b$ ) складывается в фазе с волной, излученной вибратором в свободное пространство, т. е. возникает резонанс излучения.

При сравнении графиков видно, что изменение толщины композита сильнее проявляется на ДН излучателя при больших значениях  $\epsilon$ . Если при  $d = 0,5$  см значения ДН излучателя для разных значений диэлектрической постоянной соизмеримы, то

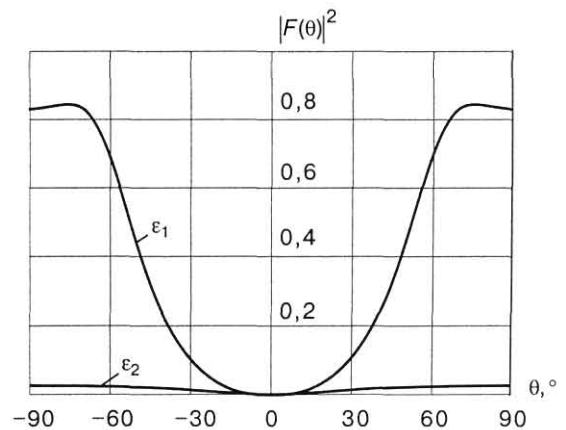
при  $d = 2$  см максимум ДН ЛМПВ для  $\epsilon = 2,54$  почти в 10 раз превышает максимальное значение ДН вибратора при  $\epsilon = 5$  (см. рис. 4–5). Такая же ситуация и в *H*-плоскости. Причиной столь резкого изменения амплитуды являются поверхностные волны, об условиях возбуждения и существования которых изложено в работах [3, 5].



■ Рис. 7. ДН ленточного МПВ в *H*-плоскости при  $d = 0,5$  см;  $\epsilon_1 = 2,54$ ;  $\epsilon_2 = 5$



■ Рис. 8. ДН ленточного МПВ в *H*-плоскости при  $d = 1,5$  см;  $\epsilon_1 = 2,54$ ;  $\epsilon_2 = 5$



■ Рис. 9. ДН ленточного МПВ в *H*-плоскости при  $d = 2$  см;  $\epsilon_1 = 2,54$ ;  $\epsilon_2 = 5$

Ширина ДН по уровню 0,707 ЛМПВ в  $E$ -плоскости практически не зависит от значений  $\epsilon$  и  $d$ . В  $H$ -плоскости ширина ДН существенно зависит от значения  $d$  (для сравнения:  $2\theta = 140^\circ$  при  $d = 0,1$  см, а при  $d = 2$  см  $2\theta = 80^\circ$  (см. рис. 6–9)).

Возможность использования микрополосковых антенн для уменьшения характеристик рассеяния от сторонних РЛС, а также при их установке на летательных аппаратах из композитов следует из предварительных расчетов и измерений, проведенных в БЭК.

1. ЭПР полуволнового вибратора (направление от сторонней РЛС нормальное):  $\sigma_1 = 0,86\lambda^2$ . Для длины сторонней волны  $\lambda = 3$  см  $\sigma_1 = 7,74$  см<sup>2</sup>;  $\lambda = 30$  см  $\sigma_1 = 774$  см<sup>2</sup>.

2. ЭПР ленточного микрополоскового полуволнового вибратора, площадь которого  $S_1 = 0,25 \frac{\lambda^2}{\epsilon}$ , вместе со схемой питания площадью  $S_2 = 0,04 \frac{\lambda^2}{\epsilon}$  будет равна

$$\sigma_2 = 4\pi \frac{S^2}{\lambda^2}.$$

Можно показать, что в приближении нормального отражения сторонней волны от металлической площадки площадью  $S = S_1 + S_2$ :

$$\sigma_2 = \frac{4\pi}{\lambda^2} \left[ 0,25^2 \frac{\lambda^4}{\epsilon^2} + 0,04^2 \frac{\lambda^4}{\epsilon^2} \right] \approx 1,25 \frac{\lambda^2}{\epsilon^2}.$$

Для длины сторонней волны  $\lambda = 3$  см,  $\epsilon = 3$   $\sigma_2 = 1,25$  см<sup>2</sup>;  $\lambda = 30$  см,  $\epsilon = 3$   $\sigma_2 = 125$  см<sup>2</sup>.

3. Определим, во сколько раз примерно уменьшилась ЭПР полуволнового вибратора в обычном и микрополосковом исполнении (относительная диэлектрическая проницаемость  $\epsilon = 3$ ): ~8 дБ.

Используя полученные расчеты и графики (см. рис. 2–9), можно рекомендовать следующие конструктивные изменения в микрополосковом исполнении вибратора (см. рис. 1), которые экспериментально подтвердили уменьшение ЭПР до 13 дБ по сравнению с обычным вибратором (при этом интегральная площадь ДН уменьшилась на 8 %, измерение проводилось в БЭК на длине волны 3 см):

а) использовать композиционные материалы с  $\epsilon \approx 3$ ,  $\text{tg } \sigma < 0,001$  в качестве подложки  $\epsilon = f(y)$ ;

б) запитку излучателя производить как можно ближе к полоске, что уменьшает потери за счет потерь в композите и РПМ экрана;

в) использовать РПМ вместо экрана композита, оставляя металлическую нижнюю полоску для запитки и излучателя на  $0,2\lambda$  шире верхней;

г) производить запитку по направлению «елочки» композита, для стороннего поля РЛС,  $\vec{E}$  которого перпендикулярен к «елочке» композита;

д) плоский вибратор устанавливать по направлению «елочки» (оси  $y$ ,  $\epsilon = f(y)$ );

е) использовать РПМ снаружи на расстоянии около  $1,2\lambda$  от кромок нижней полоски;

ж) использовать в подложке металлические включения размером не более  $\frac{\lambda}{30}$  и изрезанность  $N$  внешней металлической полосы вибратора  $> N \frac{b}{12}$ ;

з) использовать тот факт, что наличие продольного распределения  $\epsilon$  приводит в  $E$ -плоскости к смещению диаграммы направленности в положительном направлении оси  $y$ .

Работа выполнена в рамках гранта Т02-02,5-3650 Минобразования РФ (конкурс 2002 г.) по фундаментальным исследованиям в области технических наук.

## Литература

1. Нефедов Е. И., Фиалковский А. Т. Полосковые линии передачи: Электродинамические основы автоматизированного проектирования интегральных схем СВЧ. — М.: Наука, 1980. — 170 с.
2. Гвоздев В. И., Нефедов Е. И. Объемные интегральные схемы СВЧ. — М.: Наука, 1985. — 230 с.
3. Панченко Б. А., Нефедов В. И. Микрополосковые антенны. — М.: Радио и связь, 1986. — 120 с.
4. Коваленко Ю. Ф., Нефедов В. И., Советов В. Н. Микрополосковые антенны СВЧ. Обзор, модели, анализ, синтез. — В кн.: Машинное проектирование устройств СВЧ. — Тбилиси, 1979. — 240 с.
5. Красюк В. Н. Антенны СВЧ с диэлектрическим покрытием. — Л.: Судостроение, 1986. — 210 с.
6. Красюк В. Н., Михайлов В. Ф. Бортовые антенны гиперзвуковых летательных аппаратов. — СПб.: СПбГААП, 1994. — 216 с.

УДК 519.6; 621.391

# АЛГОРИТМЫ ОБРАБОТКИ СИГНАЛОВ В МЕДИЦИНСКОЙ ДИАГНОСТИКЕ С ИСПОЛЬЗОВАНИЕМ ОПОРНОГО СЛУЧАЙНОГО ПРОЦЕССА

**Э. П. Тихонов,**

канд. техн. наук, доцент

Санкт-Петербургский государственный  
электротехнический университет (ГЭТУ)

*В настоящее время для решения сложных задач медицинской диагностики, в том числе редуцированной проблемы Рэлея, например в цифровой рентгенографии и томографии, широко используются математические методы решения обратных задач, реализуемые программно на современных компьютерах. В качестве альтернативы в работе предлагается метод измерения с опорным случайным сигналом, который позволяет решать аналогичные задачи на аппаратно-программном уровне с достижением, по мнению автора, лучших метрологических характеристик.*

*Mathematical methods of the decision reverse problems are applied to the decision of problems of medical diagnostics. Alternative technologies in work are offered the method of measurement with bearing casual signal. The method examined in work allows to solve similar problems at a achievement the best metrological characteristics*

## Исходные сведения и постановка задачи

Основу современных средств медицинской диагностики составляют бесконтактные методы измерения, использующие эффекты взаимодействия зондирующих физических полей (сигналов) с исследуемым объектом. К зондирующим физическим полям относятся, например, ультразвуковые, световые, электромагнитные, рентгеновские, радиоизотопные поля [1]. В зависимости от вида зондирующего сигнала (ЗС) изменяются закономерности его взаимодействия с исследуемым объектом и соответственно методы, алгоритмы и устройства диагностики, но сохраняется общее положительное свойство — получение информации бесконтактным способом. Некоторые трудности возникают при выявлении степени изменения ЗС в зависимости от патологического состояния диагностируемых объектов, в том числе внутренних органов тела человека. Указанные трудности вызваны следующими причинами:

- нелинейным характером взаимодействия ЗС с исследуемым объектом;
- низкой чувствительностью ЗС к изменению свойств и состояния диагностируемого объекта;
- особенностью преобразования в датчиках энергии ЗС в электрическую энергию;

— низким уровнем отношения мощности сигнала к мощности помехи.

Особую роль в медицине играют бесконтактные методы диагностики на основе использования в качестве ЗС рентгеновского излучения [1, 2], причем прогрессирующее положение занимают цифровая рентгенография и томография [2, 3].

В качестве параметра исследуемого локализованного в пространстве и во времени объекта или его части в цифровой рентгенографии и томографии рассматривается изменение плотности в дискретных подобластях  $V_{ij}$  объекта, на которые условно разбивается тем или иным способом наблюдаемая область или весь объект  $V$  исследования; при этом выполняется требование

$$V = \sum_{i=1}^n \sum_{j=1}^m V_{ij}.$$

Таким образом, как бы осуществляется объемная дискретизация исследуемого объекта и тем самым дискретизация его физического параметра — плотности, причем информация, которую несет рентгеновский ЗС об измеряемой плотности, проходя через подобласть  $V_{ij}$ , усредняется по объему с торцами, ограниченными площадями  $S_{V_{ij}}$  и высотой (толщиной)  $d_{ij}$ . Центры площадей  $S_{V_{ij}}$  имеют

координаты  $x_i$  и  $y_j$ . В результате взаимодействия рентгеновского излучения с биологическим веществом в указанных объемах происходит ослабление исходной интенсивности ЗС в соответствии с законом Бера.

В современных цифровых рентгеновских установках, в отличие от томографических, зондирующий луч формируется специальным коллиматором и попадает на линейку, состоящую из конечного числа детекторов (рис. 1), которые перемещаются одновременно с рентгеновской трубкой с постоянной скоростью только в одной плоскости в направлении, например, оси  $y$  [2, 3]. В этом случае направление рентгеновского луча совпадает с осью  $z$  и, согласно упомянутому закону, наблюдаемая интенсивность выходного сигнала от каждой подобласти  $V_{ij}$  изменяется в соответствии с законом [1, 2]

$$I(x, y) = I_0(x, y) e^{-\int_{d_{ij}} \bar{\rho}(z) dz}$$

где  $I_0(x, y)$ ,  $I(x, y)$  — функции, описывающие интенсивности рентгеновского луча, проходящего через центры  $S_{V_{ij}}$  на входе и выходе исследуемой подобласти;  $\bar{\rho}(z)$  — усредненная по площади  $S_{V_{ij}}$  искомая функция плотности.

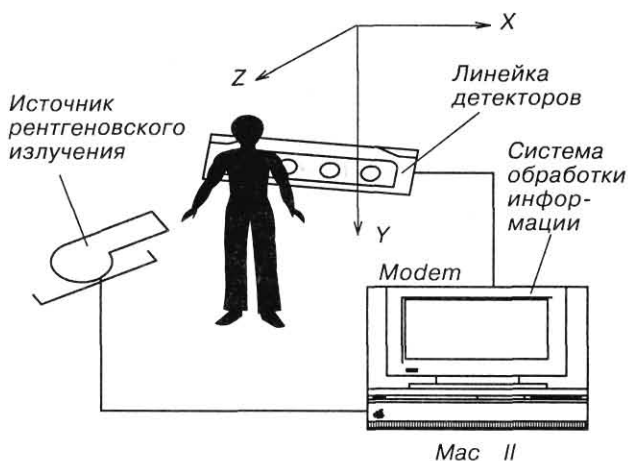
В томографах помимо поступательного перемещения с постоянной скоростью источника и приемники (детекторы) рентгеновского ЗС поворачиваются дискретно на угол  $\varphi$ . Причем сканирование подобным образом выполняется полностью для всего исследуемого объекта.

Функция плотности  $\bar{\rho}(z)$  зависит от исследуемого объекта и содержит информацию о свойствах объекта, выраженных в равенстве

$$\bar{\rho}(z) = \mu(z)\rho(z),$$

где  $\mu(z)$  — массовый коэффициент ослабления рентгеновского излучения;  $\rho(z)$  — плотность вещества, подвергающегося рентгеновскому облучению.

В медицинской диагностике наибольшую плотность имеют костные ткани, массовый коэффици-



■ Рис. 1. Схема цифровой рентгеновской установки

ент ослабления которых в 68 раз превышает аналогичный параметр мягких тканей, поэтому в костных тканях рентгеновский луч имеет наибольшее поглощение [1]. Как будет показано ниже, этот факт имеет решающее значение для предлагаемого и исследуемого в работе алгоритма измерения плотности.

В цифровых рентгеновских установках и томографах происходит накопление информации за время экспозиции  $T$  об измеряемой плотности по интенсивности  $I(x, y)$ , которая зависит также от четвертой координаты — времени, т. е.

$$R(x, y, T) = \int_0^T I(x, y, t) dt,$$

где  $R(x, y, T)$  — результат накопления информации на выходе исследуемой подобласти за время  $T$ .

В реальных цифровых рентгеновских установках накопление информации осуществляется в темпе с преобразованием энергии рентгеновских лучей в электрические сигналы. В зависимости от типа детектора электрический сигнал принимает тот или иной вид. Например, при использовании в качестве детекторов пропорциональных многопроволочных камер рентгеновские кванты преобразуются за счет ударной ионизации в электрические импульсы, которые после нормализации поступают в каждом канале на вход накапливающих двоичных счетчиков. На счетчиках осуществляется преобразование за время экспозиции  $T$  средней частоты следования случайного потока импульсов, пропорционального интенсивности рентгеновского излучения и имеющего пуассоновское распределение вероятностей, в цифровой двоичный код. Далее этот код поступает на ЭВМ, где осуществляются дальнейшая обработка полученной информации и отображение результатов обработки на дисплее или других носителях информации. При этом интеграл для определения  $R(x, y, T)$  преобразуется в сумму, а результаты измерения дискретизируются (квантуются) не только в пространстве, а и во времени с интервалом дискретизации, равным  $T$ .

Если же в качестве датчиков используются фотодиоды со слоем люминофора для преобразования рентгеновского излучения в электрический сигнал, то возможны и другие формы представления электрического сигнала, например, в виде реализации непрерывного случайного процесса. Количественные условия формирования реализации случайного процесса из импульсного исходного сигнала описаны в работе [4], а информация об интенсивности рентгеновского излучения содержится в его математическом ожидании. В этом случае целесообразно исходный случайный процесс представить в виде модели случайного квазистационарного процесса, для которого принимается гипотеза о постоянстве математического ожидания и других его вероятностных характеристик в течение времени экспозиции.

Если же за время экспозиции принятая гипотеза не выполняется, то следствием этого является возникновение дополнительной, т. е. динамической

составляющей погрешности измерения интенсивности рентгеновского излучения. Этот недостаток присущ всем подобным «накопительным» статистическим алгоритмам, усредняющим исходную информацию за фиксированное время  $T$ . Таким образом, фиксация времени  $T$ , как правило, приводит к возникновению динамической составляющей погрешности при изменении плотности в течение времени экспозиции  $T$ .

Для рентгеновской томографии удобно пользоваться компактной формулой для непрерывного представления функции плотности в виде [1, 2]

$$I(l, \varphi) = I_0(l, \varphi) e^{-\int_{L(l, \varphi)} \hat{\rho}(x, y) ds'} \quad (1)$$

где  $l$  — координата детектора;  $\varphi$  — угол поворота источника и детектора;  $I(l, \varphi)$ ,  $I_0(l, \varphi)$  — интенсивности рентгеновский ЗС на входе и выходе исследуемого объекта;  $\hat{\rho}(x, y)$  — плотность исследуемой подобласти объекта, находящейся в поле зрения луча ЗС, с координатами  $x$  и  $y$ ; при этом луч ЗС распространяется по прямой линии вида

$$x \cos(\varphi) + y \sin(\varphi) = 1.$$

При записи уравнения (1) в форме

$$\theta(l, \varphi) = \int_{L(l, \varphi)} \hat{\rho}(x, y) ds \quad (2)$$

получают так называемое преобразование, или уравнение Радона, где

$$\theta(l, \varphi) = -\ln[I(l, \varphi) / I_0(l, \varphi)].$$

Решение данного уравнения позволяет найти искомую функцию поглощения, непосредственно связанную простой функциональной зависимостью с плотностью исследуемого объекта. Учитывая связь функции поглощения с функцией плотности, далее по тексту функция  $\theta(l, \varphi)$  будет называться либо функцией поглощения, либо функцией плотности.

Трудности и методы решения уравнения Радона хорошо известны [2,3], так как оно относится к классу так называемых некорректно поставленных задач, при которых погрешности измерения оказывают существенное влияние на результаты решения. Несмотря на успешное применение известных математических методов решения некорректных задач, вопросы, связанные с повышением достоверности получения исходной информации, входящей в правую часть уравнения, по результатам измерения остаются актуальными. Это связано с тем, что качество решения данного уравнения Радона в значительной степени зависит от погрешности представления левой части уравнения (2), которую можно получить только в результате измерения. Попытка непосредственного уменьшения погрешности измерения путем программного решения обратной задачи уже применительно к результатам измерения, связанным математически с измеряемым параметром посредством уравнения Фредгольма I рода, приводит к определенным

положительным результатам [1]. Однако представляется, что возможно эффективное решение проблемы Рэлея, но уже с привлечением аппаратно-программных средств.

В данной статье на основе метода измерения со случайным опорным сигналом, предложенного и разработанного в работах [5, 6] с использованием подхода, рассмотренного в работах [7, 8] для бесконтактного измерения паросодержания, предлагается итерационный алгоритм измерения функции поглощения  $\theta(l, \varphi)$  и исследуются его свойства. Как показала практика, применение алгоритма с опорным сигналом уменьшает не только инструментальные погрешности, но и динамическую составляющую погрешности измерения, и непосредственно в процессе измерения линеаризует левую часть уравнения (2).

### Итерационный алгоритм с опорным сигналом

Суть предлагаемого алгоритма состоит в следующем. Используя информацию о ЗС на выходе исследуемого объекта при максимальном (или минимальном) поглощении рентгеновского излучения, в измерительный алгоритм дополнительно вводится случайный опорный сигнал, представляющий собой пуассоновский процесс [7, 8]. Благодаря введению такого случайного опорного процесса удается связать интенсивность рентгеновского излучения на выходе исследуемого объекта с соответствующим цифровым эквивалентом на выходе измерительного устройства посредством линейной градуировочной характеристики. Преимущества подобного метода в отличие от чисто математического способа линеаризации заключается в следующем. Опорный сигнал может формироваться в дополнительном канале, у которого максимальная или минимальная плотность нормированы путем соответствующего поглощения рентгеновского (или другого, например, ультразвукового) излучения введенными в измерительный канал фантомами. Причем параметры фантома могут подстраиваться по параметрам исследуемого объекта в процессе калибровки. Поскольку данный опорный канал непосредственно входит в измерительную линейку, то он позволяет исключить или, по крайней мере, значительно ослабить целый ряд влияющих на результат измерения отрицательных факторов и осуществить линеаризацию градуировочной характеристики, связывающей результат измерения с измеряемым параметром — искомой плотностью [7].

В математической форме данный алгоритм для каждого канала при фиксированном векторе  $z = (l, \varphi)$  имеет вид

$$k_j[(n+1)\Delta t] = k_j(n\Delta t) - \gamma_j \{ \eta_j(n\Delta t) \exp[-\theta_j(z) - \alpha k_j(n\Delta t)] - \eta_{j0}(n\Delta t) \exp[-\beta \theta_{j \max} + \alpha k_j(n\Delta t)] \}, \quad (3)$$

где  $k_j(n\Delta t)$  и  $k_j[(n+1)\Delta t]$  — цифровой эквивалент искомой функции плотности на  $n$ -м и  $n+1$ -м так-

тах итерации для  $i$ -го канала (результат измерения искомой плотности);  $\Delta t$  — временной интервал, через который опрашивается выход  $i$ -го датчика в  $i$ -м канале;  $\eta_i$  — случайная пуассоновская величина, соответствующая появлению электрического импульса в момент опроса  $i$ -го канала;  $\eta_{i0}$  — случайная пуассоновская величина, соответствующая появлению электрического импульса в опорном канале, совпадающем с моментом опроса  $i$ -го канала;  $\gamma_i$  — заданная постоянная величина или постоянная уравновешивания для  $i$ -го канала;  $\theta_i(z)$  — искомая функция плотности;  $\theta_{i\max}$  — максимальное значение плотности, которое может принимать функция плотности при данном рентгеновском обследовании для  $i$ -го канала, сформированное на фантоме;  $\alpha$  и  $\beta$  — заданные постоянные.

Для рассматриваемого алгоритма дополнительно выполняются следующие условия и ограничения:

- $n = 1, 2, \dots, N$ , где  $N$  — число итераций;  $i = 1, 2, \dots, m$ , где  $m$  — число каналов;
- $0 \leq \theta_i(z) \leq \theta_{i\max}$ ;  $0 \leq k_i(n\Delta t) \leq \theta_{i\max}$ ;  $\gamma_i = \theta_{i\max}/N_0$ , где  $N_0$  — заданное положительное число, соответствующее числу квантования максимального значения измеряемой плотности;

- начальное значение  $k_i(n\Delta t)$  равно нулю при числе итераций  $n$ , равном нулю, а для каждой новой позиции линейки датчиков сохраняется предыдущее значение  $k_i(\cdot)$ .

Отметим, что постоянную  $\gamma_i$  целесообразно выбирать  $\gamma = \text{const}$  независимо от  $i$ -го канала из условия

$$\gamma = \frac{\max_i \theta_{i\max}}{N_0}.$$

Условия функционирования рассматриваемого алгоритма можно изменить, если связать момент изменения искомого параметра  $k(n\Delta t)$  с моментом появления импульса на выходе детектора  $i$ -го канала с одновременной проверкой наличия импульса в опорном канале.

При исследовании подобных итерационных алгоритмов необходимо проверить следующие три условия, определяющие возможность использования данных алгоритмов для измерения искомого параметра. Эти условия определяются выполнением следующих требований:

- сходимостью в установленном смысле измеряемого параметра  $k(n\Delta t)$  к его установившемуся значению;
- числом тактов, определяющих время сходимости или переходный процесс от начального к установившемуся (измеренному) значению параметра;
- конечностью финальной дисперсии или дисперсией, характеризующей флуктуацию измеряемого параметра относительно установившегося значения и тем самым характеристику случайной составляющей погрешности измерения при числе тактов  $n \rightarrow \infty$ .

Если за время наблюдения выполняются условия стационарности случайного потока, то можно установить [5, 6], что искомый параметр в алгорит-

ме (2) при  $n > N$  (где  $N$  некоторое положительное число) и надлежащем выборе постоянной  $\gamma$  стремится к такому значению  $k_{i0}$ , при котором выполняется в среднем равенство

$$M_{\eta} \{ \eta_i(n\Delta t) \exp[-\theta_i(z) - \alpha k_{i0}(n\Delta t)] \} = M_{\eta} \{ \eta_{i0}(n\Delta t) \exp[-\beta \theta_{i\max} + \alpha k_{i0}(n\Delta t)] \},$$

где  $M_{\eta} \{ \eta_i(n\Delta t) \exp[-\theta_i(z) - \alpha k_{i0}(n\Delta t)] \} =$

$$= I_i \exp[-\theta_i(z) - \alpha k_{i0}(n\Delta t)]$$

и  $M_{\eta} \{ \eta_{i0}(n\Delta t) \exp[-\beta \theta_{i\max} + \alpha k_{i0}(n\Delta t)] \} =$

$$= I_{i0} \exp[-\beta \theta_{i\max} + \alpha k_{i0}(n\Delta t)],$$

т. е. выполняется сходимость по распределению [9].

Если обеспечить идентичность интенсивностей  $I_i$  и  $I_{i0}$ , то при  $n \geq N$  выполняется в среднем условие

$$k_{i0}(z) = \frac{\beta \theta_{i\max} - \theta_i(z)}{2\alpha}, \quad (4)$$

которое при соответствующем выборе постоянных  $\alpha$  и  $\beta$  определяет искомую градуировочную характеристику измерительного устройства, функционирующего в соответствии с исследуемым алгоритмом. Например, для  $\alpha = 0,5$  и  $\beta = 1$  получаем в среднем равенство

$$k_{i0}(z) = \theta_{i\max} - \theta_i(z),$$

где  $z = n\Delta t$ .

Алгоритм (3) можно представить в виде следующей модификации:

$$k_i[(n+1)\Delta t] = k_i(n\Delta t) - \gamma_i \{ \eta_i(n\Delta t) \exp[-\theta_i(z) + \alpha k_i(n\Delta t)] - \eta_{i0}(n\Delta t) \exp[-\alpha k_i(n\Delta t)] \}. \quad (5)$$

Для данного алгоритма при тех же условиях и  $\alpha = 0,5$  выполняется в среднем равенство

$$k_{i0}(n\Delta t) = \theta_{i0}(z). \quad (6)$$

Попутно заметим, что, выбирая соответствующим образом постоянную  $\alpha$ , можно достичь эффекта усиления результатов измерения и тем самым повысить чувствительность измерительного устройства, так как

$$\frac{dk}{d\theta} = (2\alpha)^{-1},$$

и для  $\alpha < 0,5$  следует  $\delta > 1$ .

### Анализ погрешности

Особенность алгоритмов (3) и (5) состоит в том, что по заданному значению  $\gamma$  можно подобрать такое значение  $\varepsilon(\gamma) > 0$ , при котором для всех  $n > N$  выполняются условия:

$$P\left\{\left|k_i(n\Delta t) - \frac{\beta\theta_{i\max} - \theta_i(n\Delta t)}{2\alpha}\right| \geq \varepsilon(\gamma)\right\} = 0;$$

$$P\{|k_i(n\Delta t) - \theta_i(n\Delta t)| \geq \varepsilon(\gamma)\} = 0,$$

где  $P\{\dots\}$  — вероятность выполнения соответствующих неравенств.

Это значит, что при постоянном конечном значении параметра  $\gamma$  можно гарантировать сходимость по вероятности  $k_i(n\Delta t)$  только в некоторую  $\varepsilon(\gamma)$ -окрестность искомого значения  $k_{0i}(n\Delta t)$ . Следовательно,  $\varepsilon(\gamma)$ -окрестность определяет абсолютное значение методической погрешности измерения искомого параметра посредством рассматриваемых алгоритмов, которая фактически эквивалентна погрешности оценки математического ожидания, возникающей из-за конечности объема выборки.

Величину  $\varepsilon(\gamma)$  можно оценить как для асимптотического случая, т. е. когда  $n$  стремится к  $\infty$ , так и при конечном числе тактов  $n$ , если воспользоваться, например, неравенством Чебышева

$$P\left\{\left|k_i(n\Delta t) - \frac{\beta\theta_{i\max} - \theta_i(n\Delta t)}{2\alpha}\right| \geq \varepsilon(\gamma)\right\} \leq \frac{D\left\{k_i(n\Delta t) - \frac{\beta\theta_{i\max} - \theta_i(n\Delta t)}{2\alpha}\right\}}{\varepsilon^2(\gamma)}. \quad (7)$$

Для того чтобы установить меру отклонения цифрового эквивалента или результата измерения  $k_i(n\Delta t)$  при конечном  $n$  от искомого значения  $k_{0i}(n\Delta t)$ , необходимо найти дисперсию, которая характеризует методическую случайную составляющую погрешности измерения искомого параметра при  $n \leq N_0$ .

Переходный процесс, характеризующий сходимость в среднем  $k_i(n\Delta t)$  к  $k_{0i}(n\Delta t)$ , и его дисперсия для алгоритма (5) в первом приближении определяются по следующим формулам:

$$k_i[n\Delta t] = (1 - v\lambda_i)^n [k_i(0) - k_{0i}] + k_{0i}; \quad (8)$$

$$D\{k_i(n\Delta t)\} \approx D_i^* [1 - (1 - v\lambda_i)^{2n}], \quad (9)$$

где  $\lambda_i = \gamma_i / l_i$ ;  $k_i(n\Delta t)$  — значение измеряемого параметра на  $n$ -м такте итерации;  $k_{0i}$  — искомое установившееся значение измеряемого параметра;  $v = \Delta t \exp(-\theta(z))$ ;  $D_i^*$  — финальная дисперсия или ее предельное значение,

$$D_i^* = \frac{\gamma}{2\alpha \exp(-\theta_i(z)) - \gamma\alpha^2(1 + 2l_i \Delta t)}. \quad (10)$$

Подставляя значение  $\gamma$  и пренебрегая вторым слагаемым в знаменателе, получаем следующую оценку финальной дисперсии:

$$D_i^* \leq \frac{\max_i \theta_i \exp\left(\max_i \theta_i(z)\right)}{2N_0\alpha}. \quad (11)$$

Пусть абсолютное значение погрешности  $\varepsilon(\gamma)$  составляет долю  $v < 1$  от  $\max_i \theta_i$ , тогда формула (11) приобретает вид

$$D_i^* \leq \frac{\exp\left(\max_i \theta_i(z)\right)}{2N_0\alpha v}.$$

В соответствии с неравенством (7) оценка вероятности превышения абсолютной погрешностью  $\Delta(\gamma)$  измерения искомой плотности заданной величины  $\varepsilon(\gamma)$  равна

$$P\{\Delta(\gamma) \leq \varepsilon(\gamma)\} = \frac{\exp\left(\max_i \theta_i(z)\right)}{2N_0\alpha v}.$$

Полученное значение вероятности характеризует фактически доверительный интервал искомой методической погрешности, надежность которого определяется значением указанной вероятности [9]. Эта вероятность обычно задается априори, и по ее значению и по известному максимуму измеряемого параметра  $\max_i \theta_i$  из выражения (12) можно найти параметр  $N_0$ .

Формулы (8) — (11) показывают, что при надлежащем выборе параметра  $N_0$ , т. е. таком его значении, при котором выполняется условие  $\lambda < 1$  для всех  $i$ , условие сходимости текущего значения измеряемого параметра  $k(n\Delta t)$  к его установившемуся значению выполняется. При этом сходимость алгоритма обеспечивается при конечной финальной дисперсии  $D_i^*$ , значение которой оценивается в соответствии с формулами (10) и (11) и полностью контролируется параметром  $N_0$ . Заметим, что полученные формулы для оценки параметра  $N_0$  дают завышенный результат из-за использования при выводе этих формул неравенства Чебышева.

### Анализ динамического режима измерения

Преимуществом данного итерационного алгоритма является также тот факт, что его работоспособность не только сохраняется при изменении за время экспозиции искомой функции плотности, но и позволяет повысить точность измерения по сравнению с известным алгоритмом обычного усреднения. Этот факт является важным хотя бы потому, что диагностируется живой объект, полную неподвижность которого за время экспозиции обеспечить невозможно.

Действительно, пусть искомая функция плотности за наблюдаемый отрезок времени изменяется в первом приближении линейно

$$\theta(n\Delta t) \approx \theta(0) + \theta'(0)n\Delta t,$$

тогда для алгоритма (1) при  $\beta = 1$  для искомого значения параметра получим

$$k[(n+1)\Delta t] = (1 - v\gamma)^n \left[ k(0) - \frac{\theta_{\max} - \theta(0)}{2\alpha} \right] +$$



$$+ \frac{\theta_{\max} - \theta(0) - \theta'(0)\{n\Delta t - T_3[1 - (1 - \nu\gamma)^n]\}}{2\alpha}$$

Из последнего выражения следует, что при достаточно большом числе тактов  $n$  искомый параметр для линейно изменяющейся функции плотности запаздывает относительно истинного значения на время  $T_3 = \Delta t/\nu\gamma$ , которое не влияет на результат решения уравнения Радона.

Положительная особенность предлагаемого алгоритма проявляется прежде всего в высокой помехоустойчивости относительно аппаратных искажений, обусловленных рассеянием рентгеновского луча, взаимным влиянием детекторов и нестабильностью коэффициента регистрации  $\epsilon$  излучения детектором. Погрешность, возникающая из-за аппаратных искажений, можно представить в виде дополнительного к основному излучению для данного канала фонового излучения  $I_{\phi}(l, \varphi)$ . Это фоновое излучение при обычном накоплении (интегрировании) приводит к возникновению дополнительной составляющей погрешности, так как алгоритм (1) преобразуется к виду

$$k_i[(n+1)\Delta t] = k_i(n\Delta t) - \gamma\{\epsilon_1[\eta_i(n\Delta t)\exp[-\theta_i(z)] + \eta_{\phi}(n\Delta t)]\exp[-\alpha k_i(n\Delta t)] - \epsilon_2[\eta_{i0}(n\Delta t) + \eta_{\phi}(n\Delta t)]\exp[-\beta\theta_{i\max} + \alpha k_i(n\Delta t)]\},$$

где  $\epsilon_1$  и  $\epsilon_2$  — коэффициенты регистрации детекторами в  $i$ -м и опорном каналах;  $\eta_{\phi}(n\Delta t)$  — фоновое излучение.

Нетрудно показать [8], что максимальная относительная дополнительная погрешность, возникающая из-за неидентичности каналов, определяется по формуле

$$\delta_{\max} = \delta_{\epsilon} \left(1 - \frac{1}{\theta_{i\max}}\right) + \frac{\eta_{\phi}}{\eta_{i0}},$$

где  $\delta_{\epsilon}$  — относительная вариация коэффициента регистрации,  $\delta_{\epsilon} = (\epsilon_2 - \epsilon_1)/\epsilon_1$ .

Так как  $1/\theta_{i\max} \ll 1$  и  $\eta_{\phi} \ll \eta_{i0}$ , то  $\delta_{\max} \approx \delta_{\epsilon}$ , и при хорошей технологии выполнения детекторов эта погрешность может достигать очень малых значений. Заметим, что данный способ исключения дополнительных аппаратных погрешностей при томографии более эффективен, чем рассматриваемый, например, в работе [2] численный метод решения проблемы Рэля, которая предшествует основному решению уравнения Фредгольма I рода.

### Результаты моделирования

Анализируемый подход был использован в реальных условиях для бесконтактного измерения радиозотопным методом плотности двухфазной среды [7, 8], который в целом хорошо себя зарекомендовал. Техническое решение радиозотопного измерения с использованием опорного сигнала было предложено, например, в работах [10, 11] и ре-

ализовано в измерительно-вычислительных комплексах, которые успешно эксплуатировались в Сибирском энергетическом институте СО АН (г. Иркутск) и в филиале ВНИИАЭС — ЭНИС (г. Электрoгорск).

Однако для наглядного представления процесса измерения применительно к рассматриваемому случаю, с учетом того, что теоретические результаты по исследованию переходных процессов получены при определенных допущениях и ограничениях, целесообразно провести более углубленное исследование на основе имитационного моделирования.

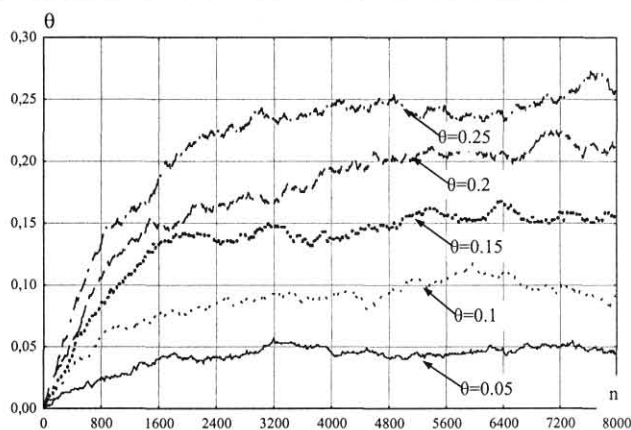
В соответствии с программой, разработанной в математическом пакете «Статистика» (см. Приложение), на рис. 2, 3 приведены результаты моделирования алгоритма для измерения значений функции плотности  $R$ . Значения  $R$  соответственно равны 0,05; 0,10; 0,15; 0,20; 0,25 и за время измерения не меняются. В программе установлены следующие значения параметров:  $\gamma = 0,001$ ; объем выборки или число итераций  $N = 8000$ ; интенсивность  $I = 0,01$ ; искомое значение функции плотности изменяется от 0,05 до 0,25 с шагом дискретизации 0,05. На рис. 2 приведены финальные значения указанных на рис. 1 параметров, т. е. без учета данных о переходном процессе, результатов измерения.

Графики, приведенные на рис. 2, имеют следующие значения:

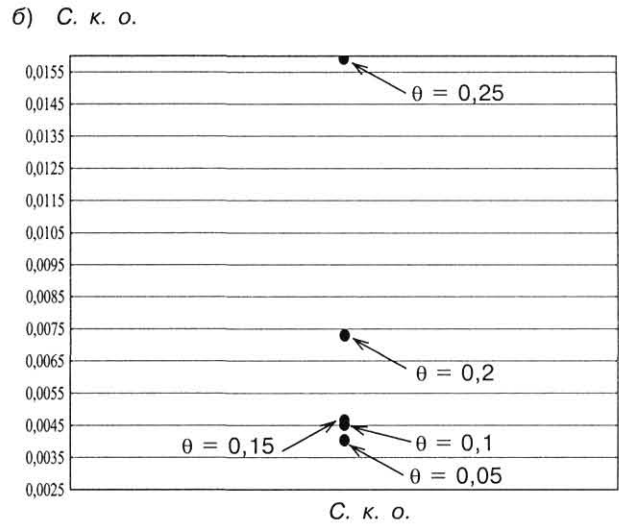
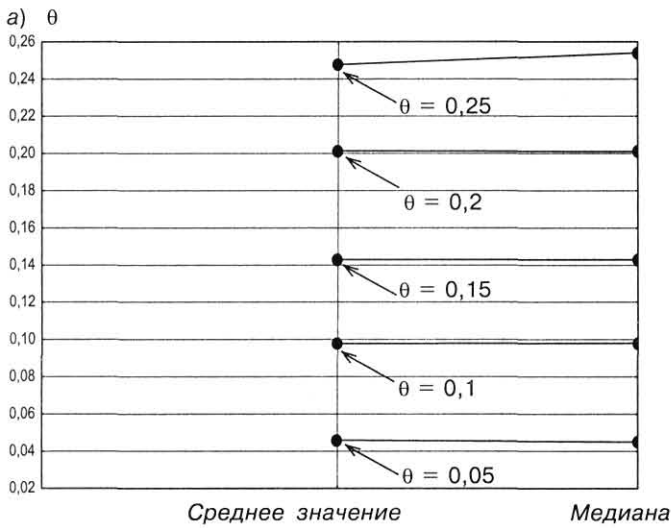
Значения $R$	0,05	0,1	0,15	0,2	0,25
MEAN (среднее)	0,05422	0,10109	0,15079	0,19882	0,23521
MEDIAN (медиана)	0,05400	0,10200	0,15100	0,20000	0,23200
SD (с. к. о.)	0,00665	0,00560	0,00589	0,00951	0,00745

Для того чтобы оценить полученные теоретически формулы для оценки свойств рассматриваемых алгоритмов, на рис. 4–8 приведены результаты сравнения аналитических расчетов и имитационного моделирования алгоритма (3) в соответствии с программой, приведенной в Приложении.

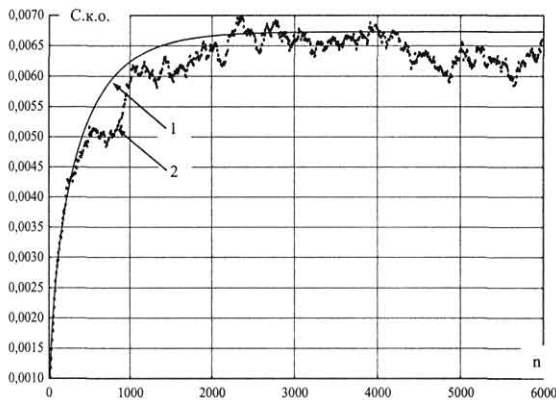
Как следует из рис. 2–8, полученные в результате моделирования значения функции плотности практически совпадают с их априорно заданными



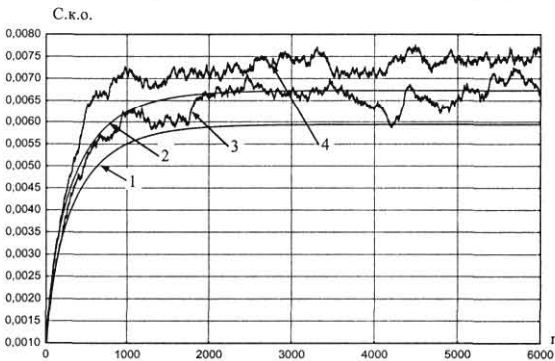
■ Рис. 2. Результаты моделирования алгоритма для измерения значений функции плотности



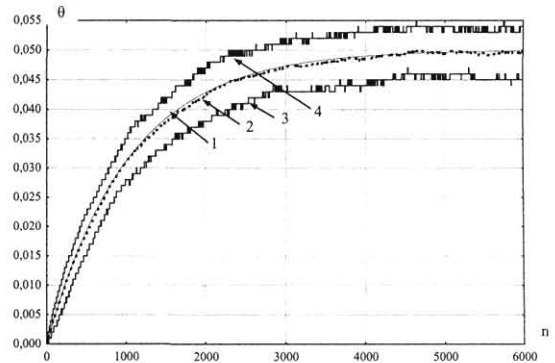
■ **Рис. 3.** Средние значения и медианы измеряемых параметров, полученные по результатам моделирования без учета переходных процессов (а) и финальные среднеквадратические отклонения (с. к. о.), характеризующие случайную составляющую погрешности измерения искомым функцией плотности для этих же измеряемых параметров и при тех же условиях (б)



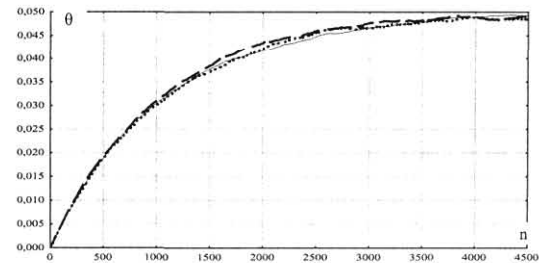
■ **Рис. 4.** Графики, иллюстрирующие результаты расчета (1) и машинного эксперимента (2) по определению с. к. о. случайной составляющей погрешности измерения функции плотности (искомая плотность равна 0,05)



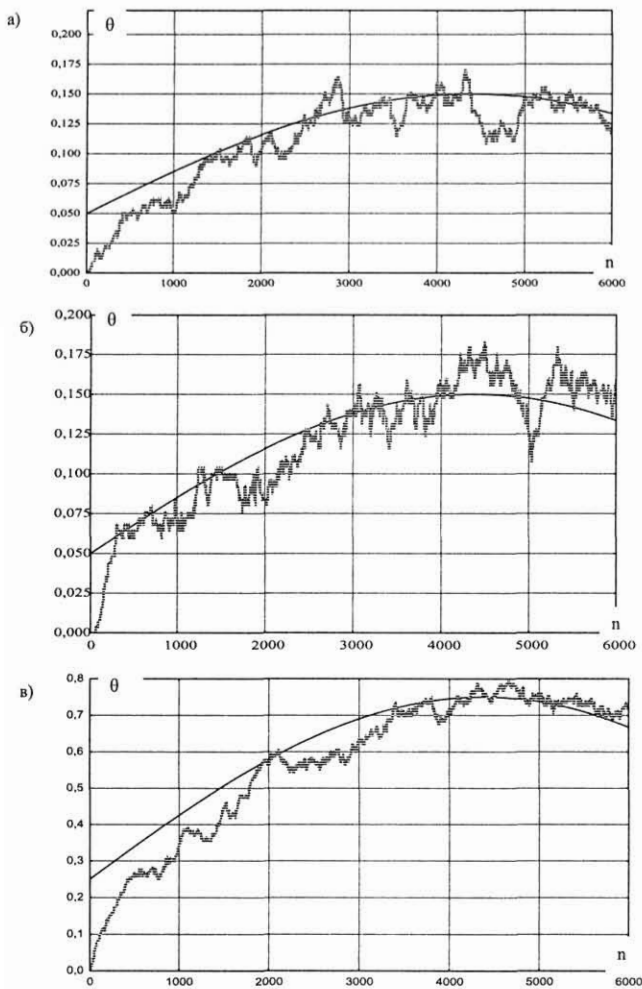
■ **Рис. 5.** Расчетные (1), (2) и экспериментальные (3) и (4) графики для различных интенсивностей (для 1, 2, 3, 4 они соответственно равны 0,01, 0,02, 0,02, 0,03) случайных пуассоновских импульсов, иллюстрирующие с учетом переходного процесса изменение с. к. о. измеряемой плотности, равной 0,05.



■ **Рис. 6.** Графики, иллюстрирующие результаты расчета среднего значения (1) измеряемого параметра  $\theta$ , равного 0,05, и машинного эксперимента (2) с учетом переходного процесса, а также значения 25% (3) и 75% (4) квантилей, определяющие случайный разброс измеряемого параметра, возникающего из-за случайного характера зондирующего рентгеновского сигнала



■ **Рис. 7.** Графики, иллюстрирующие с учетом переходного процесса усредненные данные машинного эксперимента для одного и того же значения измеряемого параметра, равного 0,05 при различных интенсивностях  $I$  случайных пуассоновских импульсов ЗС:  
 — —  $I = 0,02$ ; - - - -  $I = 0,03$ ; ..... —  $I = 0,04$



■ Рис. 8. Графики, полученные при различных постоянных усреднениях ( $\gamma$ ) с учетом переходного процесса, результатов измерения изменяющегося по синусоидальному закону исходного параметра (а — шаг равен 0,03; б, в — шаг равен 0,04; на рис. в значение измеряемого параметра увеличено на 0,6 по сравнению с его значениями, указанными на рис. а и б):  
 — входной синусоидальный сигнал плотности;  
 - - - измеренный синусоидальный сигнал плотности

значениями, что подтверждает работоспособность предложенных алгоритмов для измерения функции плотности с использованием ЗС в целях медицинской диагностики.

### Заключение

В отличие от известных математических методов решения обратных задач, например, для повышения точности результатов измерения, в предлагаемом подходе фактически решается обратная

задача на аппаратно-программном уровне. При этом решение задачи осуществляется путем аппаратного выполнения функции сравнения результатов преобразования как образцовой, так и измеряемой физической величины идентичными операторами.

Действительно, в момент уравнивания выполняется равенство

$$A(-x - k) = A^*(k - x_{\max}),$$

или

$$A(-x + k) = A^*(-k),$$

где  $A$  и  $A^*$  — идентичные операторы преобразования в измерительном и опорном каналах соответственно;  $x$  — значение измеряемого параметра из интервала  $(0, x_{\max})$ ;  $k$  — уравниваемая физическая величина;  $x_{\max}$  — максимальное значение образцовой меры, идентичность которой максимальному значению измеряемой величины устанавливается априори.

Из исходного уравнения сразу же следует решение

$$-x - k = A^{-1} \cdot A^*(k - x_{\max}).$$

Если операторы  $A \approx A^*$ , то  $A^{-1} \cdot A^* \approx 1$  и  $x = (x_{\max} - k)$ .

Имея программный доступ к регулировке операторов  $A$  и  $A^*$ , достаточно добиться их идентификации в процессе проведения калибровки непосредственно перед проведением измерений в одной точке измеряемой зависимости (в точке максимума или минимума). Тем самым достигается практически полная идентификация операторов  $A$  и  $A^*$  за счет устранения составляющих инструментальной погрешности, связанных с дрейфом и старением элементов блоков и узлов, в том числе источника ЗС, образующих измерительную цепь.

Для уменьшения случайной составляющей погрешности, возникающей из-за ведения дополнительного канала со случайным опорным сигналом, можно осуществить раздельно операцию накопления информации по опорному каналу в процессе калибровки непосредственно перед проведением операции измерения и реализовать, например, алгоритм (5) в виде

$$k_j [(n + 1) \Delta t] = k_j (n \Delta t) - \gamma_j \{ \bar{\eta}_j (n \Delta t) \exp[-\theta_j(z) + \alpha k_j (n \Delta t)] - \bar{\eta}_{j0} \exp[-\alpha k_j (n \Delta t)] \},$$

где  $\bar{\eta}_{j0}$  — результат усреднения информации, поступающей по опорному каналу после калибровки.

Таким образом, в рассматриваемой статье предлагается аппаратно-программный метод решения проблемы Рэлея, возникающей при использовании различных зондирующих сигналов в бесконтактных измерениях, в том числе в цифровой рентгенографии и томографии.

## Приложение

Основная программа для моделирования процесса измерения искомой плотности в соответствии с алгоритмом (2) имеет вид:

```
for j:=1 to M do
  Data(1,j):=0;
  for j:=4 to M do
    for i:=1 to N do
      Data(i+1,j):=Data(i,j)+D*((0.5*(1+Sign(Rnd(L)-
        -D1*exp(-R1+0.5*Data(i,j)))))-(0.5*(1+Sign(Rnd(L1)-
        -D1*exp(-0.5*Data(i,j))))));
```

Программа для моделирования изменяющейся за время экспозиции функции плотности:

```
D:=0.001; N:=7000; M:=10; L:=10; D1:=9.9; R1:=0.1;
L1:=10;
for j:=1 to M do
  Data(1,j):=0;
  for j:=2 to M do
    for i:=1 to N do
      Data(i+1,j):=Data(i,j)+D*j*((0.5*(1+Sign(Rnd(L)-
        -D1*exp(-R1*(0.5+sin(0.2*4*Pi*i/N))+0.5*Data(i,j))))-
        -(0.5*(1+Sign(Rnd(L1)-D1*exp(-0.5*Data(i,j))))));
    for i:=1 to N do
      Data(i+1,1):=R1*(0.5+sin(0.2*4*Pi*i/N));
```

В программах введены следующие параметры и постоянные:  $\gamma = D = 0,001$ ; число итераций  $N = 6000$  или (в зависимости от рис. 8000); объем выборки  $M = 200$  или (в зависимости от рис. 8000);  $L = 10$  — параметр, характеризующий равномерное распределение вероятностей, используемое для имитации пуассоновского потока квантов рентгеновского излучения;  $D1 = 9,8$  (или другие указанные на рисунках значения) — параметр, определяющий интенсивность потока квантов рентгеновского излучения;  $R1 = 0,05$  (или другие значения, указанные на рисунках) — значение измеряемой функции плотности.

Кроме того, помимо данной программы, использовались встроенные в математический пакет «СТАТИСТИКА» операторы и программы для вычисления соответствующих вероятностных характеристик результатов эксперимента.

## Литература

1. **Ремизов А. Н.** Медицинская и биологическая физика: Учебник для вузов. — М.: Высш. шк., 1999. — 616 с.
2. **Сизиков В. С.** Математические методы обработки результатов измерений: Учебник для вузов. — СПб: Политехника, 2001. — 240 с.
3. **Троицкий И. Н.** Статистическая теория томографии. — М.: Радио и связь, 1989. — 240 с.
4. **Рытов С. М.** Введение в статистическую радиофизику. Ч. 1. Случайные процессы. — М.: Наука, 1976. — 494 с.
5. **Тихонов Э. П.** Измерения с опорным случайным процессом // Метрология. — 1985. — № 10. — С. 20–29.
6. **Тихонов Э. П., Селиванова М. П.** Основные принципы, области применения и перспективы развития измерений с опорным случайным процессом // Измерения, контроль, автоматизация. — 1990. — № 3. — С. 3–6.
7. **Селиванова М. П., Тихонов Э. П.** Сравнительный анализ и выбор оптимального алгоритма радиоизотопного измерения плотности // Автметрия. — 1990. — № 3. — С. 69–72.
8. **Селиванова М. П., Тихонов Э. П.** Метод измерения плотности двухфазных сред в динамике // Автоматика. — 1989. — № 5. — С. 79–83.
9. **Ширяев А. Н.** Вероятность. — М.: Наука, 1980. — 570 с.
10. **А.с. SU 1384001 А1, СССР, МКИ G01 N 23/02 /** Радиоизотопное устройство измерения плотности в динамике / Э. П. Тихонов, А. Н. Гусев.
11. **А.с. SU 1517534 А1, СССР, МКИ G01 N 23/02 /** Радиоизотопное устройство измерения плотности в динамике / Э. П. Тихонов, Г. Г. Живилов.

УДК 602-507

## НОВАЯ ИНИЦИАТИВА В ПРОГРАММИРОВАНИИ. ДВИЖЕНИЕ ЗА ОТКРЫТУЮ ПРОЕКТНУЮ ДОКУМЕНТАЦИЮ

**А. А. Шалыто,**

*д-р техн. наук, профессор*

*Санкт-Петербургский государственный университет  
информационных технологий, механики и оптики*

*Обосновывается целесообразность разработки открытой проектной документации на программное обеспечение.*

*Here we proved advantages of developing of open project documentation for software.*

Простота требует проектирования и хорошего вкуса.

*Л. Торвальдс*

Глубоко ошибается тот, кто думает, что изделиями программистов являются программы, которые он пишет. Программист обязан создавать заслуживающие доверия решения и представлять их в форме убедительных доводов, а текст написанной программы является лишь сопроводительным материалом, к которому эти доказательства применимы.

*Э. Дейкстра*

Недавно я стал свидетелем того, как один выдающийся программист (участник двух финалов командных чемпионатов мира по программированию, проводимых организацией «Association for Computing Machinery» — ACM), в течение весьма продолжительного времени (15 мин) не мог понять программу из *шести строк* на языке Си, про которую было известно, что она решает классическую задачу (другое решение которой программист знал). После этого мы вышли в Интернет и сравнительно быстро нашли ту работу, где был весьма внятно описан алгоритм, использованный в указанной программе.

### **Почему исходные тексты не решают проблему понимания программ?**

Данный факт в очередной раз подтвердил мое мнение, что «Движение за открытые исходные тексты» («Open Source Software») не обеспечивает даже в этом достаточно простом (относительно реальных проектов) случае решения проблемы понимания программ. К счастью, так думаю не только я один. Н. Безруков, ведущий аналитик корпорации

BASF, профессор университета Fairleigh Dickinson (NJ), Web-мастер [www.softpanorama.org](http://www.softpanorama.org) — Open Source Software, считает [1]: «Центральный вопрос в практике программирования — вопрос о понимании программных текстов. Всегда хорошо иметь исходники, но проблема состоит в том, что зачастую их недостаточно. Чтобы понять некоторую нетривиальную программу, обычно требуется дополнительная документация. Эта потребность растет экспоненциально с ростом объема кода. Анализ текстов программ, направленный на восстановление первоначальных проектных решений, принятых разработчиками, и понимание программ являются двумя важными ветвями технологии программирования, существование которых неразрывно связано с недостаточностью исходных текстов для понимания программ. В качестве примера попробуйте понять структуру нетривиального компилятора при условии, что вы не располагаете определением того языка, который им компилируется.

Каждый, кто участвовал в крупном проекте по реконструкции ПО, навсегда запомнит то *чувство беспомощности и потерянности*, которое испыты-

ваешь, когда впервые видишь гору плохо документированных (но не всегда плохо написанных) исходных текстов. Доступность исходных текстов не слишком помогает, когда отсутствует доступ к ключевым разработчикам. Если программа написана на сравнительно низкоуровневом языке типа Си и плохо документирована, то все основные проектные решения обычно растворяются в деталях кодирования и требуют реконструкции. В таких ситуациях ценность документации более высокого уровня, такой как спецификация интерфейсов и описание архитектуры, может *превышать ценность самого исходного текста*.

Осознание неадекватности исходных текстов для понимания программ привело к попыткам объединить код и документацию более высокого уровня. Одна из наиболее известных попыток решения указанной проблемы была предпринята Д. Кнудом в книге «Грамотное программирование» [2]. Возможно, самой известной запрещенной в истории компьютерных наук была книга «Commentary on Unix: With Source Code» [3], содержащая высокоуровневое описание исходных текстов UNIX наряду с используемыми алгоритмами. Она нелегально копировалась и распространялась более 20 лет с момента ее первой публикации в 1977 г.

Если вы не принимали участия в работе над проектом, начиная с его ранних стадий, то сложность и объем фактически «закрывают» от вас исходные тексты, при условии, что отсутствует документация высокого уровня. Понимание «доисторического» кода в случае отсутствия первоначальных разработчиков или адекватной документации, позволяющей разобраться с соответствующими архитектурными решениями, является, вероятно, одним из труднейших видов программистской деятельности.

И еще. «Существуют ли какие-нибудь достойные свободные программы, вид которых не вызывает отвращения? До сих пор, несмотря на обилие свободного кода, нормальных программ среди него ужасно мало» [4].

О том, к чему это приводит, хорошо сказал великий русский математик, академик Л. С. Понтрягин: «Только хорошо выполненная работа дает радость! Выполненная небрежно, она вызывает отвращение и постепенно вырабатывает в человеке аморальное отношение к труду» [5].

### Почему программы не проектируются?

Итак, без исходных текстов плохо, но и с ними также нехорошо. Чего же не хватает для полного счастья? Ответ прост: *проектной* документации, выполненной весьма подробно и аккуратно, в которую *программная* документация входит как одна из составляющих.

Мне на ум приходят только три вещи, которые создаются без разработки проектной документации: дети, картины и программы. При этом использование документации не зависит от серийности — платье и в домашних условиях обычно шьется не кое-как, а по выкройке.

Мосты, дороги и небоскребы без документации обычно не строятся, а вот о программах этого не скажешь. Кстати, в археологии тоже нельзя копать, где и как Бог на душу положит, так как при этом могут возникать серьезные неприятности.

В программировании сложилась ситуация, определяемая вторым законом Вейнберга: «Если бы строители строили дома так, как программисты пишут программы, достаточно было бы одного-единственного дятла, чтобы разрушить цивилизацию» [6].

Почему на аппаратуру выпускается море подробной и внятной проектной документации, в которой специалист средней квалификации может сравнительно быстро разобраться и изменить ее даже через много лет после выпуска, а для программ такая документация либо вовсе отсутствует, либо она пишется чисто формально и для ее корректировки (если разработчик отсутствует) требуется специалист более высокой квалификации?

Видимо, причина этого заключается в следующем. Во-первых, аппаратуру разрабатывают одни люди (организации, подразделения), а изготавливают другие. Поэтому при некачественной документации разработчик проведет на «заводе» остаток жизни, чего ему делать явно не хочется. При создании программ имеет место другая ситуация, так как в этом случае обычно и разработчик, и изготовитель программ — одно и то же лицо, и поэтому какую номенклатуру документов ни создавай, их содержимое будет, как правило, весьма поверхностным. Во-вторых, аппаратура — «жесткая», а программа — «мягкая». Это упрощает внесение изменений в программу, но не служит основанием для того, чтобы вовсе не выпускать проектную документацию. Известно, что большинство программистов патологически не желают читать и уж тем более писать документацию [7].

Опыт показывает, что практически ни один молодой программист, даже очень толковый, не умеет писать проектную документацию. Несмотря на то, что многие программисты слушали и сдавали большие и сложные курсы математики, это почти никак не отражается на логике и строгости написания документации. Они, например, никак не могут во всей документации (вне зависимости от ее размера) использовать одно и то же обозначение, и поэтому один и тот же предмет называется, например, то лампа, то лампочка, то lamp, то с большой буквы, то с маленькой. И это еще не предел их фантазии!

Это, видимо, происходит потому, что при программировании компилятор подсказывает несоответствия, в то время как при написании проектной документации подсказывать некому.

Вопрос о качестве документации на программное обеспечение приобретает все большее социальное значение. Так, в статье [8], опубликованной в газете «Известия», первым среди вопросов, «о чем мы должны подумать задолго до того, как кончится нефть», является качество указанной документации. «Наиболее запомнившееся и поразившее меня при выполнении совместного проекта с фирмой IBM отличие состояло в том, как *они* относятся к доку-

ментации. У *них* было принято, и это соответствовало практике, — что написано в документации, то так и есть, так и работает. У *нас* — никогда! У *них* из этого проистекала корпоративная уверенность в себе — на все вопросы *они* находили ответ в документации, утвержденной начальством. У *нас* не только документация плоха, но и мысли изложить толком никто не умеет. Я много раз сравнивал их доклады и *наши* „доклады“. Однажды мы пригласили одного из лучших переводчиков. Он потом извинялся, что не смог толком перевести на английский, но я его успокоил, что и на русском-то ничего не понял... Этому нужно научиться. Эту часть работы следует любить и уважать». Несомненно, что и у *них* с документацией не так все хорошо, как говорит автор, но тенденция просматривается...

Разработка программ все больше напоминает шоу-бизнес с его погоней за прибылью. Все делается в дикой спешке, без раздумий о том, что будет с продуктом через некоторое время, особенно длительное. Как и в шоу-бизнесе, в программировании используются критерии «выгодно и невыгодно», а не «хорошо и плохо». В большинстве случаев хороша не та технология, которая действительно хороша, а та, которая выгодна.

Нежелание писать проектную документацию, вероятно, связано и с тем, что, чем более закрыт (не документирован) проект, тем более незаменим его автор.

Подобный стиль работы, к сожалению, распространяется и на создание программного обеспечения для особо ответственных систем. Поэтому не удивителен случай, который стал недавно широко известен. В центре Парижа из-за отказа компьютерной системы управления в новом дорогом автомобиле чуть не погибли люди, так как кондиционер перестал работать, двери не открывались, а бронированные стекла даже снаружи удалось разбить далеко не сразу.

Это во многом связано с тем, что в большинстве случаев программы пишутся, а не проектируются. «При проектировании любая техника сложнее CRC-карточек [9] или диаграмм использования [10] считается слишком сложной и не применяется. Программист всегда может отказаться от любой технологии, сказав начальнику, что он не укладывается в срок» [11]. Все это приводит к тому, что даже «пользователи не считают ошибочное программное обеспечение чем-то из ряда вон выходящим» [7].

Заметим, что такая ситуация в программировании имела место далеко не всегда — при использовании «больших» машин программы либо проектировались, либо писались очень тщательно, так как после нахождения ошибки следующий «заход» на машину происходил обычно не ранее, чем через сутки. Таким образом, технический прогресс привел к менее ответственному отношению к программированию.

### Что дает проектная документация?

При наличии качественной проектной документации программист не сможет «управлять» ме-

неджерами. После его увольнения на продолжение проекта можно нанять человека с более низкой квалификацией и зарплатой, а не с более высокой, как это обычно бывает. В конечном счете, в цивилизованной стране средний программист не должен получать больше среднего школьного учителя.

Можно ли учиться проектированию и реализации программ по книгам? Да, но по разным: проектированию — по одним [12], реализации — по другим [13]. Книг, в которых прослеживаются оба этих этапа, почему-то нет. Их отсутствие могут восполнить *открытые* проекты.

Наконец-то появилось третье издание книги [14], содержащей более 1150 страниц, в которой проектирование и реализация рассматриваются совместно на примере моделирования работы лифта для двухэтажного здания, на каждом этаже которого может находиться не более одного человека. Однако, во-первых, эта книга одна, во-вторых, этот пример один. Это напоминает библиотеку, содержащую только одну книгу. Поэтому задача создания библиотеки с большим числом открытых проектов является актуальной.

Программное обеспечение с подробной открытой проектной документацией, в которой программная документация является лишь составной частью, может рассматриваться в качестве новой разновидности паттернов проектирования [15], позволяющих достаточно просто оценить как достоинства, так и недостатки выполненных проектов. Рефакторинг (изменение структуры программы без изменения ее функциональности) [16] или перепроектирование программы в этом случае может быть выполнено значительно проще, чем при наличии только исходных текстов. Проектная документация должна содержать, в частности, формальную спецификацию, по крайней мере, на логику разрабатываемой программы, так как «то, что не специфицировано формально, не может быть проверено, а то, что не может быть проверено, не может быть безошибочным» [17], а также в связи с тем, что, «если нет спецификации, то нет и ошибок» [18].

Кроме того, проектная документация должна содержать «протоколы (истории вычислений), которые являются конструкциями, вскрывающими механизм работы программы, и поэтому среди теоретиков программирования складывается представление, что множество протоколов лучше характеризует программу, нежели сам исходный программный текст» [19].

Отметим также, что без качественной проектной документации одно из основных достоинств объектно-ориентированного программирования — повторное использование кода — может приводить к большим неприятностям [20].

И самое важное. Проектная документация необходима, так как из теории алгоритмов известно (теорема Райса), что, в общем случае, по тексту программы невозможно алгоритмически доказать истинность никаких нетривиальных свойств функции, вычисляемой при помощи данной теории.

Для понимания программ (в отличие от их исполнения), следуя Тьюрингу, требуется «проницательность и изобретательность». Но так как анализ текстов программ все равно не поддается автоматизации, а анализ «вручную» требует колоссальных затрат времени, то кажется, что по одному лишь тексту программы невозможно сделать выводы о ее свойствах.

Однако это не так. Математики нашли решение аналогичной проблемы еще в античности. Это запись доказательства понятным человеку языком, что и отличало греческую математическую школу от древнеегипетской. В последней в качестве решения геометрической задачи представлялся чертеж, снабженный «пояснительной» надписью «Смотри!». Это сродни представлению текста программы для определения того, что она делает.

Для того чтобы другие могли понять не только как работает наша программа, но и что она делает и каким образом она это делает, им необходимо представлять подробное описание (на понятном человеку даже «средней» квалификации формализованном языке) как процесса создания программы, так и ее свойств (статических и динамических), что и является, по сути, проектной документацией. Она и служит подтверждением того, что представленная программа вычисляет данную функцию — соответствует техническому заданию.

### Движение за открытую проектную документацию

Исходя из изложенного и не найдя в Интернете исчерпывающей документации практически ни на один программный проект, мы (автор и профессор В. Г. Парфенов, декан факультета информационных технологий и программирования С.-Петербургского государственного института точной механики и оптики (технического университета)) решили «вызвать огонь на себя» и объявили в ноябре 2002 г. в Санкт-Петербурге на открытии полуфинальных соревнований Северо-Восточного Европейского региона командного чемпионата мира по программированию ACM об образовании «Движения за открытую проектную документацию» (Stream for Open Project Documentation), в поддержку которого был создан сайт <http://is.ifmo.ru>.

Для придания импульса этому движению я начал педагогический эксперимент со студентами кафедры «Компьютерные технологии» СПбГИТМО (ТУ), суть которого состояла в том, что студенты разделились примерно на 40 групп (от одного до трех человек), каждая из которых должна была выполнить заинтересовавший ее проект на основе автоматного программирования (программирование с явным выделением состояний) [21], но так, чтобы работа завершилась размещением проекта на указанном сайте.

В отличие от многих студенческих работ, «выложенных» в Интернет в соответствии с лицензией AS/IS («как есть» — без ответственности за ошибки) [22], мы очень старались. Выполнение каждого проекта занимало у студентов несколько десятков

часов, из них не менее десяти часов мы проводили вместе.

Это привело к тому, что эксперимент, начавшийся в осеннем семестре 2002–2003 учебного года, завершится (почти в полном составе от стартовавших) лишь в осеннем семестре следующего учебного года, когда появятся новые «экспериментаторы».

Перечислим некоторые проекты, реализованные или реализуемые в настоящее время:

- построение визуализаторов алгоритмов для обучения дискретной математике и программированию;

- автоматная реализация интерактивных сценариев образовательной анимации с использованием *Macromedia Flash*;

- обучающая и тестирующая программа с примером настройки для изучения английского языка;

- совместное использование теории построения компиляторов и автоматного программирования;

- скелетная анимация;

- XML-формат для описания внешнего вида видеопроигрывателя ([www.crystalplayer.com](http://www.crystalplayer.com));

- управление различными объектами (дизель-генератор, турникет, кодовый замок, банкомат, светофор, кофеварка, телефон, лифт и т. д.);

- система безопасности банка;

- клиент — серверные приложения;

- построение пользовательских интерфейсов;

- реализация сетевого протокола SMTP;

- классические «параллельные» задачи («Синхронизация цепи стрелков» и «Обедающие философы»);

- моделирование игры «Пятнашки» для робота «LEGO»;

- игры («Robocode» [21], «Terrarium», «CodeRally», «Морской бой», «Lines», «Automatic Bomber», «Tron», «Однорукий бандит» и «Завалинка»).

Заметим при этом, что среди сотен танков для игры «Robocode», как отмечено в работе [24], только танк, разработанный нами, содержит проектную документацию. Такая же ситуация имеет место и для другой, еще более известной программистской игры — «Terrarium». Известны сотни созданных по правилам этой игры существ, но проектную документацию, похоже, разработали только мы. Из изложенного следует, что, если мир начал двигаться в сторону открытых исходных текстов, то можно предположить, что со временем будет разрабатываться и открытая проектная документация. Это позволит отказаться от мучительного чтения чужих программ, заменив его рассмотрением проектной документации, а когда захочется «пошевелить» мозгами, то вместо чтения программ можно будет, например, взяться за решение японских кроссвордов.

### Первые выводы

Один из наших проектов после восьмой (что далеко не рекорд) корректировки (которая привела к тому, что мой соавтор-студент, подобно чеширскому коту, потерял улыбку), по моему мнению, можно было завершить: программа имела хороший пользовательский интерфейс, нормально работа-



ла и была разработана подробная, аккуратно выполненная проектная документация, содержащая в числе прочего и исходный код.

Последнее нас и «погубило», так как открытая проектная документация, как отмечалось выше, весьма наглядно демонстрирует не только достоинства, но и все недостатки программы. Один выдающийся программист (призер чемпионатов мира по программированию) взглянул на этот код, и студент вновь начал переделывать проект. Это (с перерывами) продолжается уже более полугода (выполнено еще несколько корректировок), и я надеюсь, что хотя бы в этом проекте (игра «Морской бой») все будет по-чеховски прекрасно: и лицо (пользовательский интерфейс), и одежда (документация), и душа (исходный код программы), и мысли (работа программы). Отметим, что все это происходит после того, как зачет по проекту был давно получен. На основе данного примера можно представить, как выглядят программы и документация на них, если они выполнены по лицензии AS IS.

### Заключение

Высокая трудоемкость создания качественной проектной документации в программистском шоу-бизнесе вряд ли привьется. Эта технология является «тяжелой», в то время как сейчас все шире пропагандируются «легкие» и «шустрые» (agile) технологии [25], такие как, например, экстремальное программирование.

Однако есть и другие области программирования, в которых без «тяжелых» технологий не обойтись, и появляются новые люди, которым нравятся программы с хорошей проектной документацией. Один из студентов, впервые увидев проектную документацию, разработанную на основе предлагаемого подхода, воскликнул: «Она лучше, чем на телевизоре! Видимо, она такая же, как для подводной лодки».

Даже если разработка проектной документации не привьется широко, то с точки зрения педагогики движение будет весьма полезным для его участников. Оно важно также с познавательной и эстетической сторон для тех, кто просто знакомится с проектами, даже в случае, если они сами ее документацию не разрабатывают — ведь не каждый посетитель музея пишет картины.

«Движение за открытую проектную документацию — это действительно то, чего разработчикам не хватает. К сожалению, практически вся документация в коммерческих проектах является собственностью заказчика, который не торопится ее обновлять. Поэтому в Интернете так мало примеров реальных проектов. Сайтов с исходными текстами — море, а сайтов с примерами проектных решений — единицы. Я тут же залез в документацию в разделе «Проекты» вашего сайта и сравнил свои решения в аналогичных проектах с решениями, предложенными авторами. Жаль, что у меня не было доступа к этим текстам раньше. Я бы потратил на разработку меньше времени!» [26].

Работа выполняется при поддержке Российского фонда фундаментальных исследований по

гранту № 02-07-90114 «Разработка технологии автоматного программирования». Вариант настоящего текста был ранее опубликован в журнале Мир ПК — ДИСК. 2003. № 8.

### Литература

1. **Безруков Н.** Повторный взгляд на «собор» и «базар» // ВУТЕ/Россия. — 2000. — № 8.
2. **Literate Programming.** — Stanford: Center for the Study of Language and Information, 1992.
3. **Lions J.** Lions' Commentary on UNIX: With Source Code. — Annabooks, 1977.
4. **Протасов П.** Снизу // Компьютерра. — 2003. — № 19.
5. **Исследователь «руля»** // Информатика. — 2003. — № 11.
6. **Блох А.** Закон Мерфи // ЭКО. — 1983. — № 1–3.
7. **Демин В.** Проблемы выхода российских разработчиков на Запад // PC WEEK/RE. — 2001. — № 32.
8. **Скрипников А.** Когда кончится нефть // «Известия». 09.09.2003.
9. **Бадд Т.** Объектно-ориентированное программирование. — СПб.: Питер, 1997.
10. **Буч Г., Рамбо Д., Джекобсон А.** UML. Руководство пользователя. — М.: ДМК, 2000.
11. **Фаулер М.** Новые методологии программирования // www.spin.org.ua.
12. **Буч Г.** Создание будущего // Подборка статей на тему «Программы следующего десятилетия» // «Открытые системы». — 2001. — № 12.
13. **Страуструп Б.** Язык программирования C++. — М.: Бином; СПб.: Невский диалект, 2001.
14. **Дейтел Х. М., Дейтел П. Дж.** Как программировать на C++. — М.: Бином, 2003.
15. **Гамма Э., Хелм Р., Джонсон Р., Влиссидис Дж.** Приемы объектно-ориентированного проектирования. Паттерны проектирования. — СПб.: Питер, 2001.
16. **Фаулер М.** Рефакторинг. — М.: Вильямс, 2003.
17. **Зайцев С. С.** Описание и реализация протоколов сетей ЭВМ. — М.: Наука, 1989.
18. **Аллен Э.** Типичные ошибки проектирования. — СПб.: Питер, 2003.
19. **Ершов А. П.** Смешанные вычисления // В мире науки. — 1984. — № 6.
20. **Тэллес М., Хсих Ю.** Наука отладки. — М.: КУДИЦ-ОБРАЗ, 2003.
21. **Шалыто А. А., Туккель Н. И.** Программирование с явным выделением состоянием // Мир ПК. — 2001. — № 8, 9. <http://is.ifmo.ru>, раздел «Статьи».
22. **Романовский И. В.** Дискретный анализ. — СПб.: БХВ-Петербург, Невский диалект, 2003.
23. **Шалыто А. А., Туккель Н. И.** Танки и автоматы // ВУТЕ/Россия. — 2003. — № 2. <http://is.ifmo.ru>, раздел «Статьи».
24. **Озеров А.** Четыре танкиста и компьютер // Магия ПК. — 2002. — № 11. <http://is.ifmo.ru>, раздел «О нас».
25. **Cockburn A.** Agile Software Development. — NJ: Addison-Wesley, 2001.
26. **Трофимов С.** [info@caseclub.ru](mailto:info@caseclub.ru).

## «ОРГАНИЗАЦИЯ ЭВМ И СИСТЕМ»

**Б. Я. ЦИЛЬКЕРА И С. А. ОРЛОВА**

Авторами рукописи являются доктора технических наук Цилькер Борис Яковлевич — профессор кафедры сетевых технологий Рижского института транспорта и связи и Орлов Сергей Александрович — профессор, заведующий кафедрой программного обеспечения того же института.

Предполагаемый тираж 5000 экз., год выпуска — 2003, издательство «ПИТЕР».

В последние годы стали появляться учебники для вузов по целому ряду популярных направлений подготовки дипломированных специалистов, в том числе и по предметам, относящимся к информатике и вычислительной технике. Однако хорошего учебника по организации ЭВМ и систем, одному из самых основных предметов такого важнейшего направления, как «Информатика и вычислительная техника», уже долгие годы нет. Имеются только переводы наиболее известных книг по этой тематике, тогда как желательно иметь учебную литературу, написанную с учетом наших отечественных требований и традиций. Одной из лучших, по общему мнению, является 4-е издание книги известного специалиста Э.Таненбаума — «Архитектура компьютера», однако материал, в ней излагаемый, далек от требований Государственного образовательного стандарта.

Рукопись учебника Цилькера Б. Я. и Орлова С. А. в том виде, в котором она попала на рецензию, имеет большой объем, что имеет как положительные, так и отрицательные моменты. К положительным следует, прежде всего, отнести реализованную авторами возможность широко и, в то же время, глубоко охватить тему. Рукопись содержит достаточно полное и последовательное изложение одноименной дисциплины (ОПД.Ф.07 — «Организация ЭВМ и систем») в соответствии с государственным образовательным стандартом по направлению 654600 («Информатика и вычислительная техника»). В ней рассмотрены все темы, перечисленные в стандарте.

Весь учебный материал хорошо структурирован и довольно удачно распределен по главам (их в учебнике — 16) и разделам. Рассмотрены вопросы архитектуры вычислительных машин и систем, функциональная и структурная организации процессора, организация памяти ЭВМ, основные стадии выполнения команды при различных способах адресации, организация прерываний и ввода-вывода, периферийные устройства, архитектурные особенности организации ЭВМ различных классов, параллельные системы, понятие о многомашинных и многопроцессорных вычислительных системах. Материала действительно много, причем весь он непосредственно относится к изучаемой дисциплине.

В книге 484 рисунка, которые очень помогают в изучении представленного материала. Наличия такого большого количества рисунков свидетельствует об очень хорошей методической проработке учебного материала. Видно, что авторы в учебнике постарались учесть весь свой богатый педагогический опыт. Следует отметить, что учебник написан на достаточно высоком научном уровне, использование формальных моделей корректно, все определения четко сформулированы и сопровождаются разъяснениями, примерами и иллюстрациями.

Учебник написан четким и понятным для студентов языком, содержит минимум ошибок и опечаток, носит

завершенный характер. Материал курса излагается последовательно, следуя перечисленным в стандарте темам, хорошо проиллюстрирован. Каждая глава имеет контрольные вопросы. Следует заметить, что некоторые вопросы, сформулированные в виде заданий, могут быть с успехом использованы для организации самостоятельной работы студентов или проведения лабораторного практикума. Авторам следовало бы выделить такие вопросы отдельно и в явном виде придать им статус заданий, а не контрольных вопросов.

Авторы приводят большой список использованной и дополнительной литературы, однако среди 235 наименований более 190 представляют собой труднодоступные для рядового студента иностранные публикации. Понятно желание авторов привести все использованные источники, однако студенты, как правило, хотя и увидят в списке литературы рекомендуемые к изучению работы, в которых имеются более подробные сведения по тому или иному вопросу. Так, например, жаль, что, несмотря на достаточно подробное изложение исторических аспектов развития вычислительной техники, в обсуждаемом списке нет ссылки на очень интересную и познавательную книгу «С. А. Лебедев» (М.: Физматлит, 2002. — 440 с.), в которой приводятся наиболее полные сведения по развитию отечественной вычислительной техники.

В качестве других замечаний можно отметить следующие. Авторы не совсем корректно с позиций использования общепринятой терминологии описали методы управления вводом-выводом, а тема прерываний (описание устройств и методов обработки прерываний) рассмотрена не в полном объеме и с отходом от принятой терминологии. К сожалению, далеко не везде приводится терминология на английском языке. Не секрет, что знание этих терминов очень помогает в работе. В качестве недостатка можно отметить то, что весь учебный материал не разделен по важности. Другими словами, стоило бы выделить самое главное, а остальное подать как дополнение. Ведь такой большой объем дидактического материала может даже стать проблемой при самостоятельной подготовке студентов, поскольку объем часов, отводимых на изучение этой дисциплины, оговорен в стандарте.

Имеющийся в рукописи указатель явно поможет сориентироваться в книге, однако его незначительное расширение позволило бы пользоваться рассматриваемым учебником и в качестве справочника.

Конечно, в такой большой рукописи не трудно указать и другие, хотя и менее важные, погрешности. Однако считаю их не столь уж существенными, и надеюсь, что авторы сами обратят на них внимание. Тем более что редакция издательства «Питер» заверила в том, что авторы намерены доработать свой учебник с учетом замечаний, полученных и от других рецензентов.

Для рукописи запрашивается гриф «Допущено Министерством образования Российской Федерации в качестве учебника для студентов высших учебных заведений, обучающихся по направлению «Информатика и вычислительная техника», который, несомненно, целесообразно ему присвоить».

*Доктор технических наук, профессор  
Гордеев А. В.*

*Российская Академия наук,  
Комитеты по образованию Санкт-Петербурга и Ленинградской области,  
Академия информатизации образования,  
Санкт-Петербургский государственный университет,  
Санкт-Петербургский государственный университет аэрокосмического приборостроения,  
MICROSOFT и ряд других организаций*

## П Р И Г Л А Ш А Ю Т

принять участие и выступить с докладами  
школьников, студентов, аспирантов, преподавателей и специалистов

**на 23-й Международной конференции**

### **ШКОЛЬНАЯ ИНФОРМАТИКА И ПРОБЛЕМЫ УСТОЙЧИВОГО РАЗВИТИЯ. (EDUCATIONAL INFORMATICS AND SUSTAINABLE DEVELOPMENT PROBLEMS)**

Конференция состоится 16–17 апреля 2004 г. в Санкт-Петербурге

Конференция проводится в соответствии с планом мероприятий Министерства образования РФ и ряда международных организаций.

На конференции предполагается обсудить вопросы развития и применения новых информационных технологий в искусстве, образовании, здравоохранении, экологии, экономике, юриспруденции, промышленности, сельском хозяйстве, местном самоуправлении, в космонавтике и авиации и других сферах. На конференции будут рассматриваться вопросы моделирования различных систем, объектов и процессов, вопросы разработки баз и банков данных, экспертных систем и систем искусственного интеллекта, систем виртуальной реальности, обучающих программ, электронной коммерции, развлекательных и игровых программ и устройств, вычислительных сетей и систем интегрированного сервиса, вопросы музыкальной информатики, интерактивного компьютерного кино и других компьютерных искусств, виртуальных музеев и структур.

На отдельной секции будут рассматриваться вопросы создания электронной базы данных музейных фондов в области научной, научно-технической и образовательной деятельности, находящихся в учреждениях и организациях города, вопросы создания ВИЗИОНАРИУМА, посвященного 60-летию Победы.

В рамках конференции проводится Санкт-Петербургский фестиваль компьютерных искусств – мероприятие, предназначенное для самых широких общественных масс. Любой желающий может в нем участвовать. Жюри фестиваля в течение всего календарного года принимает работы, выполненные с использованием компьютерных технологий в следующих номинациях: компьютерная графика, музыка, анимация и web-дизайн. Итоги фестиваля подводятся каждый год 10 апреля в актовом зале Санкт-Петербургского государственного университета аэрокосмического приборостроения. Авторы лучших работ награждаются призами от компаний Microsoft, Intel, Lynx BCC, Linux Ink, «Титаник» и Musicland. Фестивалю оказывают информационную поддержку издательства «Техноподиум», «КомпьютерPrice» и «КомпьютерМаркет», журнал «Информационно-управляющие системы».

Отбор работ происходит в анонимном виде: на рассмотрение жюри они попадают пронумерованными. Фамилия или псевдоним автора оглашаются только после того, как работа займет призовое место. Авторам некоторых работ, даже не получивших призов, будет предложено дальнейшее сотрудничество. Особенно это касается таких жанров, как музыка и анимация.

Тезисы докладов объемом в 1 страницу принимаются до 1 февраля 2004 г. по адресу: 190000, Санкт-Петербург, ул. Большая Морская, 67. Санкт-Петербургский государственный университет аэрокосмического приборостроения.  
Председателю Оргкомитета конференции  
«Школьная информатика  
и проблемы устойчивого развития»  
профессору М. Б. Игнатьеву.  
Справки по телефону (812) 313-70-44.  
Факс (812) 315-77-78.  
E-mail: eisdp@aanet.ru

Работы на конкурс принимаются по адресу:  
190000, Санкт-Петербург,  
ул. Большая Морская 67, ГУАП,  
кафедра вычислительных систем и сетей,  
координатору фестиваля  
Люденовскому Сергею  
Тел.: (812) 342-32-06  
E-mail: ludinovskov@mail.ru

**АНДРЕЕВ  
Сергей  
Викторович**



Аспирант кафедры управления и информатики в технических системах Санкт-Петербургского государственного университета аэрокосмического приборостроения. Окончил Санкт-Петербургский государственный политехнический университет в 2002 г., факультет интеллектуальных систем и технологий. Области научных интересов: человеко-машинные системы управления движением; оптимизация тактического и стратегического управления автомобилем.

**АСТАПКОВИЧ  
Александр  
Михайлович**



Начальник СКБ СПбГУАП. Окончил Ленинградский политехнический институт в 1977 г. по специальности «Информационно-управляющие системы». Диссертацию по моделированию температурных состояний узлов ядерных энергетических реакторов защитил в 1985 г. Участник 2-й фазы международного проекта «ИТЭР» (исследовательский термоядерный энергетический реактор). Имеет более 90 публикаций, в том числе автор 1 монографии. Область научных интересов: информационно-управляющие системы реального времени.

**БРЖЕЗОВСКИЙ  
Александр  
Викторович**



Доцент кафедры компьютерной математики и программирования Санкт-Петербургского государственного университета аэрокосмического приборостроения (ГУАП). Выпускник ЛИАП 1989 г. Диссертацию на соискание ученой степени кандидата технических наук защитил в 1991 г. Автор 27 научных работ. Области научных интересов: технология разработки программного обеспечения; системы искусственного интеллекта.

**ГОРБАЦКИЙ  
Владимир  
Владимирович**



Ассистент кафедры антенн и эксплуатации РЭА Санкт-Петербургского государственного университета аэрокосмического приборостроения (ГУАП). Окончил ГУАП в 1997 г. по специальности «Радиотехника». Автор 10 научных работ, в том числе 1 монографии и 1 учебного пособия. Области научных интересов: адаптивные алгоритмы измерения параметров движения летательных аппаратов; антенны возвращаемых космических аппаратов; антенны миллиметрового диапазона длин волн.

**ДАШЕВСКИЙ  
Владимир  
Павлович**



Аспирант Санкт-Петербургского института информатики и автоматизации (СПИИРАН). Окончил физико-технический факультет Санкт-Петербургского государственного политехнического университета в 2000 г. Области научных интересов: системы реального времени; цифровая обработка сигналов; распределенные системы.

**ФОМИН  
Александр  
Владимирович**



Ассистент кафедры компьютерной математики и программирования Санкт-Петербургского государственного университета аэрокосмического приборостроения (ГУАП). Окончил ГУАП в 2000 г. Автор десяти научных работ. Области научных интересов: концептуальное моделирование программного обеспечения; технология программирования; компонентное программное обеспечение.

**МЕЛЬНИКОВА**

**Арина  
Юрьевна**



Ассистент кафедры антенн и эксплуатации радиоэлектронной аппаратуры Санкт-Петербургского государственного университета аэрокосмического приборостроения (ГУАП). Окончила ГУАП в 1995 г. по специальности «Радиотехника».

Автор 14 научных и методических работ.

Область научных интересов: разработка методик управления характеристиками рассеяния антенных систем летательных аппаратов

**ПЛАТОНОВ**

**Олег  
Юрьевич**



Ассистент кафедры антенн и эксплуатации радиоэлектронной аппаратуры Санкт-Петербургского государственного университета аэрокосмического приборостроения (ГУАП). Окончил ГУАП 1996 года по специальности «Радиотехника».

Автор 13 печатных работ.

Область научных интересов: разработка методик управления характеристиками рассеяния антенных систем летательных аппаратов

**ТИХОНОВ**

**Эдуард  
Прокофьевич**



Доцент кафедры биомедицинской электроники и охраны среды Санкт-Петербургского государственного электротехнического университета (ГЭТУ). Выпускник Ленинградского института авиационного приборостроения.

Член-корреспондент Метрологической академии.

Автор более 150 научных публикаций, в том числе более 50 авторских свидетельств и патентов.

Область научных интересов: алгоритмы и аналого-цифровые системы обработки информации.

**КРАСЮК**

**Владимир  
Николаевич**



Доктор технических наук, профессор, заведующий кафедрой антенн и эксплуатации РЭА Санкт-Петербургского государственного университета аэрокосмического приборостроения (ГУАП). Окончил Ленинградский институт авиационного приборостроения в 1972 г. по специальности «Радиотехника». Докторскую диссертацию защитил в 1987 г.

Автор более 400 научных работ, из них 14 книг.

Член-корреспондент Международной академии наук высшей школы, Петровской академии наук и искусств. Первый заместитель председателя Научного Совета по проблемам прикладной радиофизики СПНЦ РАН. Председатель Фонда развития новых технологий Русского вольного экономического общества — Невский союз.

Области научных интересов: антенны возвращаемых космических аппаратов, малозаметных летательных аппаратов; антенны миллиметрового диапазона длин волн; распространения волн в сложных средах.

**ШАЛЫТО**

**Анатолий  
Абрамович**



Заведующий кафедрой «Информационные системы» Санкт-Петербургского государственного университета информационных технологий, механики и оптики.

Окончил ЛЭТИ в 1971 г. по специальности «Автоматика и телемеханика».

Ученый секретарь НПО «Аврора».

Диссертацию на соискание ученой степени доктора технических наук защитил в 1999 г. Автор большого числа научных трудов, в том числе 3 книг. Автор более 70 изобретений.

Член редакционной коллегии журнала «Информационно-управляющие системы».

Области научных интересов: системы логического управления; автоматное программирование

*Редакция журнала напоминает, что ответственность за подбор, достоверность и точность фактов, экономико-статистических и технических показателей, собственных имен и прочих сведений, а также за то, что в материалах не содержится сведений, не подлежащих открытой публикации, несут авторы публикуемых в журнале материалов и рекламодатели.*

УДК 623.681

Алгоритм оптимального управления движением в условиях неполной определенности среды  
*Андреев С. В.* — Информационно-управляющие системы, 2003. — № 4. — С. 2–5.

Изучаются различные подходы в области нечетких систем с целью разработки алгоритмов управления движением в условиях не полной определенности среды. Разрабатывается алгоритм оптимизации пути следования объекта управления на некоторой местности, описание которой не полностью определено. Ставится и решается задача автоматизированного принятия решений по выбору оптимального пути, с учетом вероятностных свойств среды. Созданный оптимизатор пути следования, основан на оптимизационном анализе информации о текущей ситуации и принимает решение в зависимости от условий управления.

Список лит.: 6 назв.

УДК 681.325.5: 518.5

Формализм адресно-временных карт для описания алгоритмов функционирования многоканальных систем управления  
*Астапкович А. М.* — Информационно-управляющие системы, 2003. — № 4. — С. 6–14.

Работа посвящена изложению нового способа укрупненного описания алгоритмов функционирования многоканальных систем управления реального времени. Предложенный формализм включает в себя понятия: процессограмма, циклограмма, адресно-временная карта. Рассматриваются алгоритмические аспекты использования предложенного способа описания и символьное представление алгоритмов для наиболее распространенных дисциплин диспетчеризации. Вводятся три типа конфликтов в глобальном конкурентном пространстве.

Список лит.: 6 назв.

УДК 681.32

Методика вероятностного анализа наборов задач в однопроцессорных системах реального времени с фиксированными приоритетами  
*Дашевский В. П.* — Информационно-управляющие системы, 2003. — № 4. — С. 15–23.

Предложена методика вероятностного анализа выполнимости периодических задач в системе реального времени. Методика позволяет гарантировать качество обслуживания каждой задачи в системе, где максимально возможная загрузка процессора превышает единицу, а средняя остается меньше единицы. Представленный алгоритм численного анализа является эффективным как в отношении машинного времени, так и в отношении требуемой оперативной памяти.

Список лит.: 18 назв.

УДК 623.681

The algorithm of optimum control of movement in conditions of incomplete definiteness of environment  
*Andreev S. V.* — IUS, 2003. — № 4. — P. 2–5.

Given article is devoted to studying of indistinct systems with the purpose of development of algorithms of management by movement in conditions not to full definiteness of environment. During research various approaches were investigated in the field of indistinct management. In particular, the methods based on application of expert systems and combinatory tasks, realizing search of optimum managing influence on object were offered.

The algorithm of optimization of a transit of object of management on some district which description is not completely determined is developed. The task of the automated acceptance of decisions is put and solved at the choice of an optimum way, with the account probability properties of environment. Created algorithm of transits, it is based on optimum analysis of the information on the current situation and makes a decision depending on conditions of management.

Refs: 6 titles.

УДК 681.325.5: 518.5

Address-time card formalism for description of the multichannel control system algorithms  
*Astapkovitch A. M.* — IUS, 2003. — № 4. — P. 6–14.

Article concerns the new method of the robust description of the multichannel real-time control system algorithm. Proposed formalism includes: processogram, cyclogramm, address-time card. Algorithm aspects were examined also as symbolic description for the most popular scheduler disciplines. Three types of the conflicts in the global concurrent space were introduced.

Refs: 6 titles.

УДК 681.32

Probability Analysis of Fixed Priority Real-Time Task Sets  
*Dashevsky V. P.* — IUS, 2003. — № 4. — P. 15–23.

This paper describes probabilistic analysis of periodic real-time tasks with varying execution times. The analysis provides performance guaranties for the task sets with total maximum utilization higher than one while the average utilization remains less than one. Each task may me pre-empted by higher priority task and has deadline not grater than its period. Proposed algorithm is memory and time-efficient.

Refs: 18 titles.

УДК 681.324

Расширения реляционной модели для обеспечения безопасности в базах данных  
 Фомин А. В., Бржезовский А. В. — Информационно-управляющие системы, 2003. — № 4. — С. 24–32.

В статье предлагаются расширения реляционной модели для обеспечения безопасности при доступе к данным на основе мандатной модели безопасности Белла и ЛаПадула. В отличие от существующих моделей, предлагаемая модель позволяет обеспечить ссылочную целостность в базах данных «стандартным» образом. Это достигается за счет введения свойства «Разделение», которое заключается в том, что исходная схема отношения реляционной модели должна быть разделена на две: ключевую схему отношений  $R'$ , которая содержит уникальные первичные ключи, и схему данных  $R''$  — собственно данные исходного отношения. Такой подход позволяет исключить дублирование ключей при явлении многозначной реализации.

Список лит.: 11 назв.

УДК 621. 396. 67

Особенности распространения радиоволн миллиметрового диапазона, перспективы их использования в современных радиотехнических системах  
 Красюк В. Н., Платонов О. Ю., Мельникова А. Ю. — Информационно-управляющие системы, 2003. — № 4. — С. 33–38.

Освоение ММД длин волн обусловлено очевидными преимуществами ММД при использовании его в РТС ближней локации. Несомненный интерес вызывают исследования, связанные с изучением особенностей распространения радиоволн ММД в атмосфере, особенно в сложных метеорологических условиях, в условиях тумана и задымленности. В статье приводятся результаты сравнительного анализа основных параметров РЛС, работающих в длинноволновой и коротковолновой частях ММД длин волн (8 мм и 3 мм). Выделены несколько перспективных направлений развития РТС, использующих ММД длин волн.

Список лит.: 6 назв.

УДК 621. 396. 67

Особенности применения микрополосковых антенн на летательных аппаратах из композитов с малыми характеристиками рассеяния  
 Красюк В. Н., Горбачкий В. В. — Информационно-управляющие системы, 2003. — № 4. — С. 39–42.

На примере расчета характеристик излучения ленточного микрополоскового вибратора и экспериментальных исследований приведены изменения в его конструктивном исполнении, которые дают возможность использовать его с композиционными материалами и уменьшить величину рассеяния на 8–13 дБ по сравнению с обычным полуволновым вибратором.

Список лит.: 6 назв.

УДК 681.324

Extension of relational model for database security  
 Fomin A. V., Brzhezovsky A. V. — IUS, 2003. — № 4. — P. 24–32.

Relational model extensions for database security are suggested. These models in contrast to existing models allow supplying with reference integrity by standard manner. This arrives by division original relation scheme on two bounded scheme — «key scheme» and «data scheme». In key scheme there are apparent key of original relation. When polyinstantiation occurs in original relation scheme, in key scheme unique apparent keys keep. This alloy to provide reference integrity on primary key of key scheme. For suggested concept entity integrity, null integrity, interinstance integrity, polyinstantiation integrity are proposed.

Refs: 11 titles.

УДК 621. 396. 67

The features of millimeters radio-waves propagation, perspectives of its application in the modern radio-technical systems

Krasyuk V. N., Platonov O. U., Melnikova A. U. — IUS, 2003. — № 4. — P. 33–38.

Evident advantages of millimeter wavelength range, using in the radio-technical systems of near location, had been condition of millimeter range wide mastering. Investigations associating with research of features of the millimeter radio-waves propagation in atmosphere, especially in complete meteorological conditions, such as a fog and a smoke, excite certain interest. The comparative analysis results of basic characteristics of the radar, operating in the different parts of the millimeter wavelength range (8 mm and 3 mm), has been suggested in this article. Therefore, may be selected some perspective directions of radio-technical systems, using of millimeter wavelength range, development.

Refs: 6 titles.

УДК 621. 396. 67.

Features of application of microstrip antennas on flying devices from composites with small characteristics of dispersion

Krasyuk V. N., Gorbatsky V. V. — IUS, 2003. — № 4. — P. 39–42.

On an example of calculation of radiation characteristics of the tape microstrip vibrator and experimental researches the changes in its design are adduced which enable to use it with composite materials and to reduce value of dissipation on 8–13 dB as contrasted to bynormal half-wave dipole.

Refs: 6 titles.

УДК 519.6; 621.391

Алгоритмы обработки сигналов в медицинской диагностике с использованием опорного случайного процесса

*Тихонов Э. П.* — Информационно-управляющие системы, 2003. — № 4. — С. 43–51.

В настоящее время для решения задач медицинской диагностики, например, в цифровой рентгенографии и томографии, широко используются математические методы решения обратных задач. В качестве альтернативы в работе предлагается метод измерения с опорным случайным сигналом. Рассматриваемый в работе метод позволяет решать аналогичные задачи на аппаратно-программном уровне с достижением, по мнению автора, лучших метрологических характеристик.

Список лит.: 11 назв.

UDK 519.6; 621.391

Algorithms of processing of signals in medical diagnostics with use of bearing casual process  
*Tikhonov E. P.* — IUS, 2003. — № 4. — P. 43–51.

Mathematical methods of the decision reverse problems are applied to the decision of problems of medical diagnostics. Alternative technologies in work are offered the method of measurement with bearing casual signal. The method examined in work allows to solve similar problems at a achievement the best metrological characteristics.

Refs: 11 titles.

УДК 602-507

Новая инициатива в программировании. Движение за открытую проектную документацию

*Шальто А. А.* — Информационно-управляющие системы, 2003. — № 4. — С. 52–56.

Обосновывается целесообразность разработки открытой проектной документации на программное обеспечение

Список лит.: 26 назв.

UDK 602-507

The new initiative in programming. Movement for the open design documentation

*Shalyto A. A.* — IUS, 2003. — № 4. — P. 52–56.

Here we proved advantages of developing of open project documentation for software

Refs: 26 titles.



## **Уважаемые авторы журнала**

### **«ИНФОРМАЦИОННО-УПРАВЛЯЮЩИЕ СИСТЕМЫ»!**

*При подготовке рукописей статей  
редакция просит вас руководствоваться следующими рекомендациями*

**Редакция принимает к публикации статьи в следующие тематические разделы:** обработка информации и управление; информационно-управляющие системы; моделирование систем и процессов; программные и аппаратные средства; защита информации; системный анализ; стохастическая динамика и хаос; управление в социально-экономических системах; управление в медицине и биологии; краткие научные сообщения; рецензии (на книги, журналы, статьи, диссертации); хроника и информация (о семинарах, конференциях, выставках, юбилеях, а также очерки по истории науки и техники).

Объем статьи (текст, таблицы, иллюстрации и библиография) не должен превышать эквивалента в 20 страниц, напечатанных на бумаге формата А4 на одной стороне через 1,5 интервала Word (не более 70 знаков в строке) шрифтом Times New Roman размером 13.

Обязательными элементами оформления статьи являются: индекс УДК, заглавие, инициалы и фамилия автора (авторов), ученая степень, полное название организации, аннотация (5–10 строк).

#### **В редакцию предоставляются:**

- отпечатанный (формат А4) текст статьи, подписанный всеми авторами с указанием даты представления, и иллюстрации, пронумерованные с подрисуночными подписями (в двух экземплярах);
- статья в виде файла Microsoft Word или Adobe PageMaker 6.0 (шрифт Times New Roman размер 12) на дискетах 1,44Mb или CD (иллюстрации в текст не заверстываются и предоставляются отдельными файлами \*.tif, \*.bmp, \*.jpg с максимальным разрешением);
- аннотация (5–10 строк) на русском и английском языках;
- сведения об авторах (фамилия, имя, отчество, ученая степень, ученое звание, место работы, должность, области научных интересов, домашний и служебный адреса и телефоны, факс, e-mail), фото авторов (можно в электронном виде с максимальным разрешением);
- экспертное заключение (при необходимости).

**Список литературы** составляется по порядку ссылок в тексте и оформляется следующим образом:

- для книг и сборников — фамилия и инициалы авторов, полное название книги (сборника), город, издательство, год, общее число страниц;
- для журнальных статей — фамилия и инициалы авторов, полное название статьи, название журнала, год издания, номер журнала, номера страниц;
- ссылки на иностранную литературу следует давать на языке оригинала без сокращений.

#### **Адрес редакции:**

191011, Санкт-Петербург, Инженерная ул., 6.  
ФГУП «Издательство „Политехника“».  
Редакция журнала «Информационно-управляющие системы».  
Тел./факс 312–53–90,  
E-mail: [asklab@aanet.ru](mailto:asklab@aanet.ru), [gfm@polytechics.spb.ru](mailto:gfm@polytechics.spb.ru)