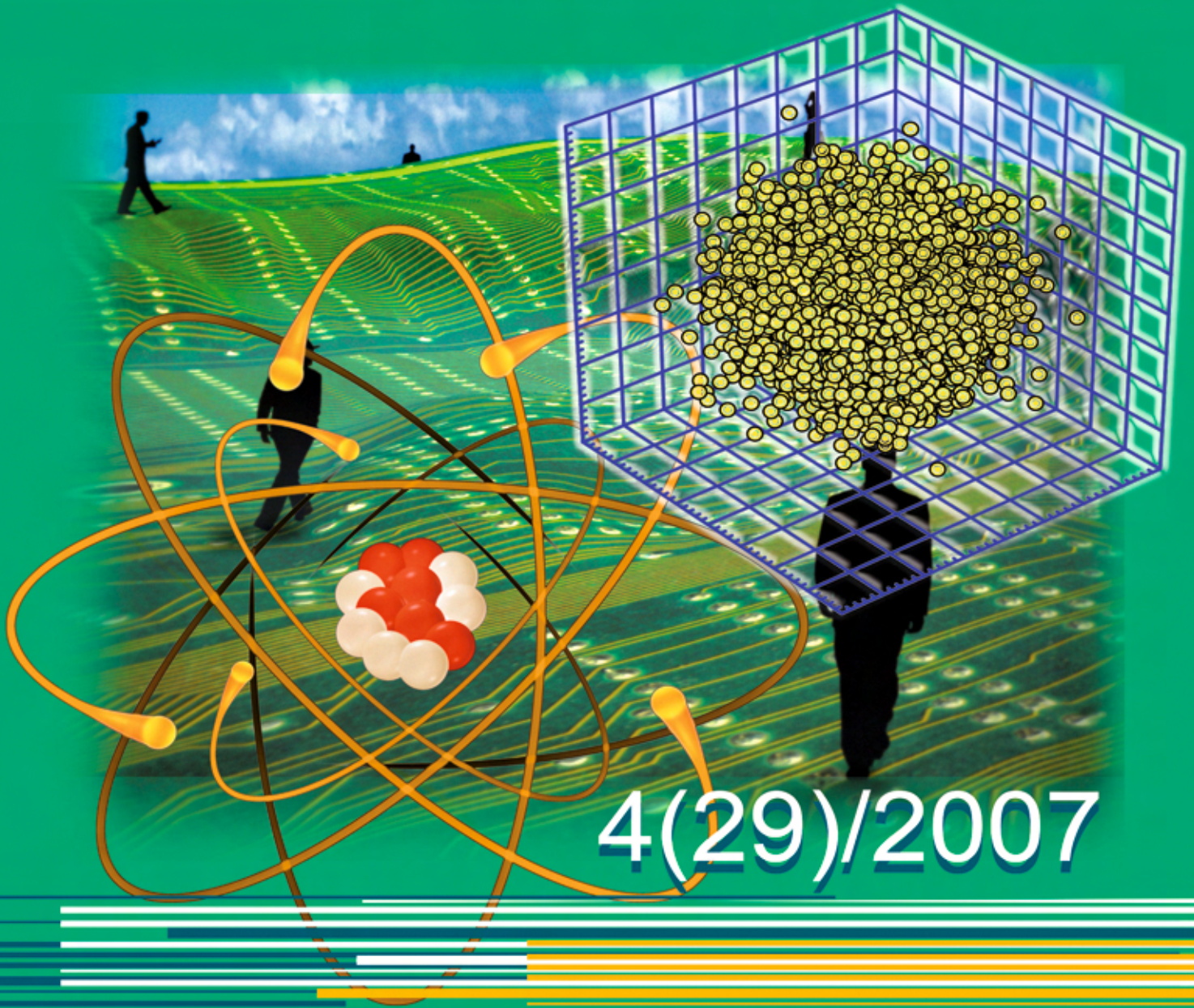


ИНФОРМАЦИОННО- УПРАВЛЯЮЩИЕ СИСТЕМЫ

НАУЧНО-ПРАКТИЧЕСКИЙ ЖУРНАЛ



4(29)/2007

4(29)/2007

ИНФОРМАЦИОННО-УПРАВЛЯЮЩИЕ СИСТЕМЫ

РЕЦЕНЗИРУЕМОЕ ИЗДАНИЕ

Учредитель

ОАО «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, чл.-корр. РАН

Редакционная коллегия:

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

Редактор: А. Г. Ларионова

Корректор: Т. В. Звертановская

Дизайн: М. Л. Черненко, А. Н. Колешко

Компьютерная верстка: Т. М. Каргапольцева

Ответственный секретарь: О. В. Муравцова

Адрес редакции: 190000, Санкт-Петербург,
Б. Морская ул., д. 67, ГУАП, РИЦ
Тел.: (812) 494-70-36
Факс: (812) 494-70-18
E-mail: 80x@mail.ru; ius@aanet.ru
Сайт: www.i-us.ru

Журнал зарегистрирован в Министерстве РФ по делам печати, телерадиовещания и средств массовых коммуникаций.
Свидетельство о регистрации ПИ № 77-12412 от 19 апреля 2002 г.

Журнал входит в «Перечень ведущих рецензируемых научных журналов и изданий, в которых должны быть опубликованы основные научные результаты диссертации на соискание ученой степени доктора и кандидата наук».

Журнал распространяется по подписке. Подписку можно оформить через редакцию, а также в любом отделении связи по каталогам: «Пресса России» – № 42476; «Роспечать» («Газеты и журналы») – № 15385

© Коллектив авторов, 2007

ОБРАБОТКА ИНФОРМАЦИИ И УПРАВЛЕНИЕ

Тихонов Э. П. Алгоритмическое описание и сравнительный анализ свойств сигма-дельта АЦП (Часть 1) 2

Великохатко Ю. П., Миронов А. А. Однозначно раскрашиваемые графы с минимумом ребер 13

Павлов В. С., Савинов М. В. Синтез дискриминатора пеленгационной системы с эллипсообразным расположением точек приема 17

ПРОГРАММНЫЕ И АППАРАТНЫЕ СРЕДСТВА

Степанов О. Г., Шалыто А. А., Шопырин Д. Г. Предметно-ориентированный язык автоматного программирования на базе динамического языка RUBY 22

Новиков Ф. А., Степанян К. Б. Язык описания диаграмм 28

КОДИРОВАНИЕ И ПЕРЕДАЧА ИНФОРМАЦИИ

Андреев С. Д., Семенов С. А., Тюрликов А. М. Методики оценки параметров радиоканала 37

ИНФОРМАЦИОННЫЕ КАНАЛЫ И СРЕДЫ

Савищенко Н. В. Помехоустойчивость модемов с двумерными сигнальными конструкциями по точным формулам вероятности ошибки в канале без замираний и с общими четырехпараметрическими замираниями 44

ХРОНИКА И ИНФОРМАЦИЯ

Ведерников Ю. В., Матросов В. В. Системный анализ. Взгляд на прошлое и настоящее 55

V Санкт-Петербургская межрегиональная конференция «Информационная безопасность регионов России – ИБРР-2007» 58

СВЕДЕНИЯ ОБ АВТОРАХ

59

АННОТАЦИИ

62

ЛР № 010292 от 18.08.98.

Сдано в набор 25.06.07. Подписано в печать 20.08.07. Формат 60×84/8. Бумага офсетная. Гарнитура SchoolBookС. Печать офсетная. Усл. печ. л. 8,0. Уч.-изд. л. 9,0. Тираж 1000 экз. Заказ 392.

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

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

УДК 681.314

АЛГОРИТМИЧЕСКОЕ ОПИСАНИЕ И СРАВНИТЕЛЬНЫЙ АНАЛИЗ СВОЙСТВ СИГМА-ДЕЛЬТА АЦП (Часть 1)

Э. П. Тихонов,

канд. техн. наук, доцент

Санкт-Петербургский государственный электротехнический университет

Предложено аналитическое описание алгоритма работы сигма-дельта АЦП в виде нелинейного отображения, на основании которого осуществлено исследование его характеристик и выполнен аналитическими методами и посредством имитационного моделирования сравнительный анализ особенностей его функционирования.

We propose a non-linear mapping that describes the algorithm of work of the delta-sigma analog-to-digital converter. This model is used to investigate the characteristics of the converter and give a comparative analysis of its features.

Известно, какую роль играют в информационных технологиях средства преобразования форм информации, в том числе различные типы и модификации аналого-цифровых преобразователей (АЦП), составляющие заметную долю продукции на современном рынке электронных компонентов [1–8]. Именно поэтому это направление в информационных технологиях обуславливает теоретический интерес, так как не только способствует дальнейшему совершенствованию АЦП, а и содействует обоснованному выбору типа АЦП с учетом требований решаемой технической задачи. Для этого необходимо знать, по возможности, все нюансы, связанные с особенностями работы соответствующего типа АЦП. Не останавливаясь на анализе различных способов реализации и разнообразных видах алгоритмов аналого-цифрового преобразования, лежащих в основе соответствующих типов АЦП, рассмотренных, например, в работах [1, 2, 7, 9–11], подвергнем анализу особенности алгоритма функционирования широко разрекламированного, высокоточного так называемого сигма-дельта ($\Sigma\Delta$) АЦП [1–3, 7, 8]. В дальнейшем под понятием функционирования подразумевается смена состояний анализируемого прибора или устройства по определенным правилам, законам или алгоритмам в зависимости от значения входного сигнала.

В современной, в основном технической литературе, а также в публикациях фирм-производи-

телей данного типа АЦП (например, фирмы Analog Devices) даются, как правило, технические сведения и отрывочные данные о свойствах и особенностях работы $\Sigma\Delta$ АЦП [2, 4–6]. В то же время вызывает теоретический интерес принцип функционирования, свойства и подробный аналитико-имитационный сравнительный анализ адекватного алгоритма функционирования $\Sigma\Delta$ АЦП, который обеспечивает по сравнению с другими типами АЦП, по данным фирм-производителей, его очень высокую разрядность (до 24) в результатах аналого-цифрового преобразования. Но для такого анализа необходимо в математической форме описать и в явном виде представить соответствующий адекватный алгоритм, отсутствие которого затрудняет полное понимание специфики и сущности работы $\Sigma\Delta$ АЦП даже у самих инженеров-разработчиков. Различные модификации основной схемы $\Sigma\Delta$ АЦП, естественно, будут приводить к некоторой модификации алгоритма функционирования основной схемы $\Sigma\Delta$ АЦП.

Прежде чем составить аналитическое представление алгоритма работы $\Sigma\Delta$ АЦП, рассмотрим основной вариант его структурной схемы с кратким описанием ее принципа действия. На рис. 1 изображена наиболее распространенная структурная схема $\Sigma\Delta$ АЦП [2]. Приведем краткое описание принципа действия схемы. Входной аналоговый сигнал $x_{вх}$ поступает на суммирующий вход ана-

логового сумматора, который совместно с аналоговым интегратором *АИ*, сравнивающим устройством *СУ* и одноразрядным (1-битовым) цифроаналоговым преобразователем (*ЦАП*), состоящим из *D*-триггера с входами *D* и *S* и двухполярного источника образцового напряжения *ИОН*, выполняет так называемую функцию сигма-дельта-модулятора (сумматора приращений). Непосредственно функция дельта-модуляции реализуется *СУ* и *D*-триггером. Выходной импульсный сигнал *D*-триггера с заданной тактовой частотой f_t параллельно поступает на цифровой фильтр и в цепь обратной связи, включающей 1-битовый *ЦАП*, управление полярностью которого осуществляется выходным сигналом *D*-триггера. Таким образом, частота переключения полярности 1-битового *ЦАП*, образующего цепь отрицательной обратной связи, зависит от частоты переключения *D*-триггера, т. е. от тактовой частоты f_t . Выходной сигнал *ЦАП*, уровень которого совпадает с заданным диапазоном преобразования E_0 сигнала $x_{вх}$, фактически непосредственно участвует в организации уравнивающей величины. Сама уравнивающая величина формируется интегратором *АИ* в темпе с поступлением тактовой частоты f_t . Таким образом, чем выше тактовая частота f_t , тем за меньшее время устанавливается приращение сигнала на интеграторе *АИ*. Сигнал E_0 с выхода *ЦАП*, согласно схеме, вычитается из входного сигнала $x_{вх}$, и результат вычитания снова поступает на вход интегратора *АИ*.

Структурная схема, изображенная на рис. 1, по принятой разработчиком классификации, отображает архитектуру $\Sigma\Delta$ АЦП первого порядка. Порядок модулятора определяется числом задействованных в его схеме сумматоров и, как следствие, интеграторов. Сигма-дельта модуляторы k -го порядка содержат k сумматоров и k интеграторов и по логике работы схемы должны обеспечивать большее соотношение сигнал/шум при той же частоте отсчетов, по сравнению с модуляторами первого порядка. Применяют сигма-дельта модуляторы высокого порядка, например, в одноканальном $\Sigma\Delta$ АЦП седьмого порядка, реализованном в микросхеме AD7720, и двухканальном $\Sigma\Delta$ АЦП пятого порядка, выполненном в микросхеме ADMOD79.

В соответствии с релятивным описанием работы АЦП по представленной структурной схеме аналитическая запись алгоритма функционирования $\Sigma\Delta$ АЦП с модуляторами первого порядка имеет адекватное нелинейное описание в виде следующего отображения:

$$E(n\Delta t) = E[(n-1)\Delta t] + \int_0^{\Delta t} \{x + E_0 \text{sign}[x - E[(n-1)\Delta t]]\} dt. \quad (1)$$

Выполняя интегрирование в (1) с учетом того, что выражение в фигурных скобках не зависит от времени (пренебрегая конечной частотной полосой пропускания интегратора), получаем

$$E(n\Delta t) = E[(n-1)\Delta t] + \alpha \{x + E_0 \text{sign}[x - E[(n-1)\Delta t]]\} \quad (2)$$

или

$$E(n\Delta t) = E[(n-1)\Delta t] + \beta \{ \text{sign}[x - E[(n-1)\Delta t]] + \lambda \}, \quad (3)$$

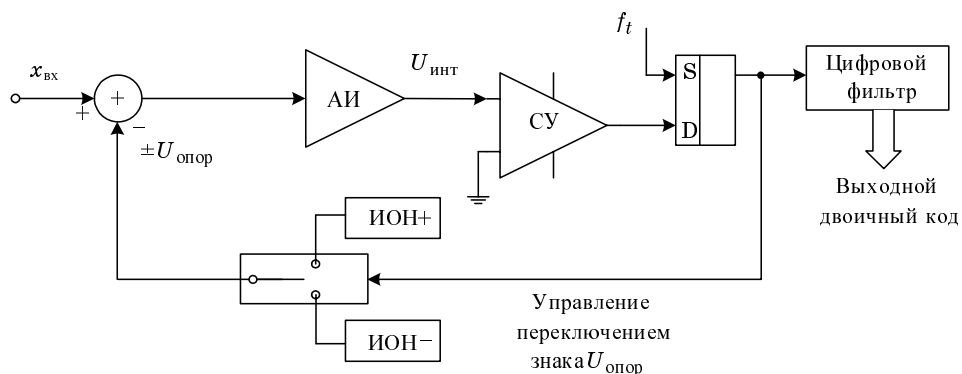
где $E(n\Delta t)$ и $E[(n-1)\Delta t]$ — значение физической величины, например напряжения, которое накапливается на интеграторе за период Δt тактовой частоты f_t ; $\alpha = \int_0^{\Delta t} dt = \Delta t = \frac{1}{f_t}$; $\beta = \alpha E_0$; $\lambda = x/E_0$,

$0 \leq \lambda \leq 1$, если $0 \leq x \leq E_0$.

В общем случае выражения (2) и (3) можно объединить в виде

$$E(n\Delta t) = G\{\alpha, x, E_0, E[(n-1)\Delta t]\}, \quad (4)$$

где $G\{x, E_0, E[(n-1)\Delta t]\}$ — нелинейная функция, описывающая характер отклонения входного сигнала от уравнивающей величины. Конкретный вид данной функции раскрыт выше. Причем слабое влияние полосы пропускания интегратора



■ Рис. 1. Структурная схема сигма-дельта АЦП

на формирование сигнала нелинейной отрицательной обратной связи обусловлено наличием двух нелинейных элементов — сравнивающего устройства и триггера, при этом срабатывание сравнивающего устройства практически не зависит от уровня приращения сигнала на его входе.

С учетом известного равенства, связывающего индикаторную и знаковую функции:

$$h(\theta) = 0,5\{1 + \text{sign}(\theta)\},$$

получаем следующее эквивалентное аналитическое отображение алгоритма работы $\Sigma\Delta$ АЦП:

$$E(n\Delta t) = E[(n-1)\Delta t] + \beta\{2h[x - E[(n-1)\Delta t]] + \chi\}, \quad (5)$$

где $\chi = \lambda - 1$.

Прежде чем перейти к исследованию полученных алгоритмов, целесообразно отметить, что данные алгоритмы в соответствии с принятой в математике терминологией относятся к нелинейным итерационным уравнениям или более кратко — отображениям [11–17]. Следовательно, с технической точки зрения, заданные в явном виде отображения как раз и определяют закономерности функционирования рассматриваемого устройства, прибора или системы, т. е. алгоритмы их работы.

Переход к математической форме описания алгоритма функционирования и идентификация принадлежности его к соответствующему классу математических объектов позволяют применить по ходу изложения статьи с необходимыми дополнениями и пояснениями относительно рассматриваемого случая исследования тот богатый арсенал идей, терминологии, методов исследования и других атрибутов, которые накоплены в математике в области нелинейной динамики. Такой подход к исследованию алгоритмов работы АЦП, несмотря на имеющийся уже достаточно большой исторический период, связанный с созданием и исследованием АЦП и их алгоритмов (см., например, работы [9, 10, 18–24]), далеко не традиционен и не очевиден и именно поэтому требует введения дополнительных понятий и пояснений. В дальнейшем термины «алгоритм», «отображения», «итерационные уравнения в конечных разностях» или просто «уравнения» будут пониматься как синонимы.

Математическое представление (3) алгоритма $\Sigma\Delta$ АЦП первого порядка инициирует предварительное рассмотрение алгоритма следящего АЦП, заданного в виде

$$E(n\Delta t) = E[(n-1)\Delta t] + q\text{sign}[x - E[(n-1)\Delta t]], \quad (6)$$

где $q = 2^{-N} E_0$ — величина кванта. Этот алгоритм подробно исследован автором [24] с учетом воздействия помех относительно традиционными мето-

дами. Отметим, что алгоритм (6) представляет собой типичный пример одномерного нелинейного отображения, куда входит значение входного сигнала в виде некоторого параметра. При визуальном сравнении алгоритмов (3) и (6) возникает невольное ощущение, что появление алгоритма $\Sigma\Delta$ АЦП является продуктом эволюции алгоритма следящего АЦП. Фактически алгоритм (3), если принять точку зрения теории генетических алгоритмов [25], является мутантом алгоритма следящего АЦП, поскольку выполняет функцию аналого-цифрового преобразования на другом качественном уровне — параметрического и временного усреднения с признаками следящего алгоритма аналого-цифрового преобразования и экспоненциального алгоритма усреднения [26]. Особенностью и общностью всех алгоритмов аналого-цифрового преобразования, обладающих всеми признаками нелинейных отображений, является следующее:

- наличие нелинейной функции сравнения;
- область изменения параметра строго ограничена диапазоном преобразования E_0 ;
- число итераций, т. е. число операций сравнения, конечно и определяется способом выполнения операции сравнения (параллельным, параллельно-последовательным и последовательным) и числом разрядов;
- решение уравнений, описывающих алгоритмы, при заданном числе итераций должно оказаться в некоторой окрестности установленного значения параметра (значения входного сигнала), т. е. фактически данная окрестность представляет собой аттрактор [11–17].

С точки зрения метрологии, указанная в последнем пункте окрестность как раз и определяет погрешность аналого-цифрового преобразования. Поэтому очень важно количественно связать свойства алгоритма с некоторыми характеристиками данной окрестности или аттрактора, подразумевая при этом, что специфика построения алгоритма аналого-цифрового преобразования предусматривает решенным вопросом его сходимости в заданную окрестность. Однако, как это будет показано ниже, для алгоритма $\Sigma\Delta$ АЦП эта задача далеко не тривиальна и, как представляется, именно при исследовании сходимости данного алгоритма применение идей нелинейной динамики наиболее плодотворно.

Для лучшего представления о возможностях использования идей нелинейной динамики исследуем свойства следящего алгоритма (6), начав с предварительного анализа переходного процесса, связанного с поиском его решения при отсутствии помех. Пусть для начального значения уравновешивающей величины выполняется равенство $E(0) = 0$, тогда при $n = 1$ и $x > 0$ имеем $E(1) = q$. Для $n = 2$ и $x > q$ имеем $E(2) = 2q$ и т. д. Пусть для $n = k$ выполняется равенство $E(k) = kq$, а для $k + 1$ получаем $x < E(k + 1) = (k + 1)q$. Следовательно, со-

гласно алгоритму, при следующих итерациях уравновешивающая величина, соответствующая решению уравнения (6), на каждом такте будет изменяться относительно значения kq на величину $\pm q$ и, таким образом, она будет равна входному сигналу с точностью до кванта и, с точки зрения нелинейных отображений, образует аттрактор. Аттрактор, представляющий в виде псевдофазового портрета [17] (в дальнейшем — просто фазового портрета) финальную траекторию уравновешивающей величины при бесконечном увеличении числа итераций, периодичен (рис. 2). При дальнейшем анализе для упрощения формул положим величину временного шага $\Delta t = 1$. Это допущение никак не влияет на общие результаты анализа.

Рассмотрим другой подход к анализу алгоритма (6). Поскольку уравновешивающая величина при достаточно большом числе итераций стремится с точностью до кванта к фиксированному значению параметра, то представим этот параметр в виде двоичного кода с погрешностью усечения, т. е.

$$x = E_0 \sum_{i=1}^N a_i 2^{-i} + \gamma,$$

где $0 \leq \gamma \leq q$ есть случайная величина с равномерным законом распределения вероятностей (погрешность квантования или усечения). Тогда алгоритм (6) можно представить в виде отображения

$$E(n) = E(n-1) + q \operatorname{sign} \left[E_0 \sum_{i=1}^N a_i 2^{-i} + \gamma - E(n-1) \right].$$

Усредним полученное отображение по случайной величине γ , в результате перейдем к следующему эквивалентному усредненному отображению:

$$E(n) = E(n-1) + q \left\{ 2F_\gamma \left[E_0 \sum_{i=1}^N a_i 2^{-i} - E(n-1) \right] - 1 \right\},$$

где $F_\gamma \left[E_0 \sum_{i=1}^N a_i 2^{-i} - E(n-1) \right]$ — функция распределения погрешности усечения.

Для этого отображения нетрудно определить так называемую особую или неподвижную точку [11–17] усредненного алгоритма из уравнения

$$2F_\gamma \left[E_0 \sum_{i=1}^N a_i 2^{-i} - E(n-1) \right] - 1 = 0.$$

Ввиду симметричности функции распределения вероятностей приходим к выводу, что решение данного уравнения будет достигаться при равенстве

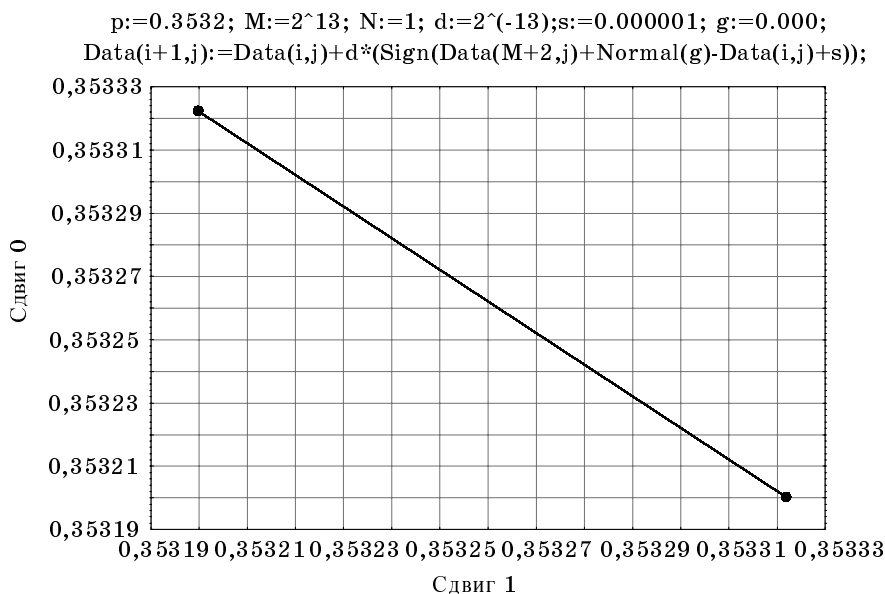
$$E(n-1) = E_0 \sum_{i=1}^N a_i 2^{-i}, \text{ т. е. неподвижная точка}$$

соответствует случаю, когда уравновешивающая величина равна двоичному эквиваленту входного сигнала.

Пусть теперь будем учитывать влияние аддитивной помехи, тогда алгоритм (6) преобразуется к виду

$$E(n) = E(n-1) + q \operatorname{sign} [x + \xi(n) - E(n-1)], \quad (7)$$

и динамика решения алгоритма (7) будет повторять динамику решения алгоритма (6), пока разность между сигналом и уравновешивающей ве-



■ Рис. 2. Фазовый портрет, характеризующий аттрактор слеящего алгоритма в отсутствие помех

личной не будет соизмерима с уровнем помехи, т. е. когда решение отображения (7) не достигнет области аттрактора. В этом случае при многократном повторении алгоритма с одним и тем же значением входного сигнала независимо от начального значения уравнивающей величины его реализации (траектории решения) от преобразования к преобразованию будут отличны друг от друга. Согласно алгоритму (7), при последующих итерациях уравнивающая величина будет иметь другой фазовый портрет, отличный от портрета, приведенного на рис. 2, и, следовательно, другую форму аттрактора. Этот фазовый портрет для погрешности преобразования представлен в трехмерном фазовом пространстве (рис. 3).

По существу, учет аддитивной помехи приводит, с математической точки зрения, к случайному изменению параметра алгоритма при фиксированном входном сигнале. Если это изменение ограничено и происходит в заданном диапазоне, то в связи с устойчивостью алгоритма в целом оно не должно существенно влиять на конечный результат преобразования. Однако картина может измениться, если помеха на верхней и нижней границах диапазона выведет значение параметра за заданные пределы. Именно устойчивость алгоритма в целом приводит к заметному эффекту помехоподавления данного типа алгоритмов, если среднеквадратическое отклонение (СКО) помехи превышает величину кванта. Чтобы убедиться в этом, для сравнения с фазовым портретом уравнивающей величины приведен фазовый портрет га-

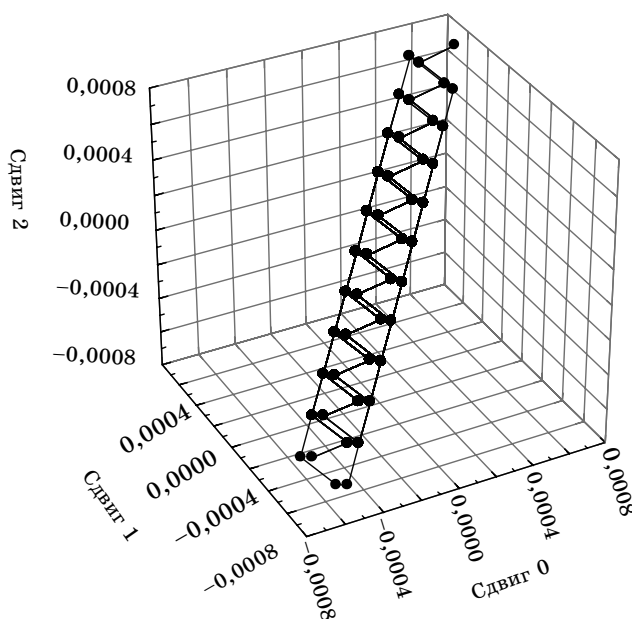
усова шума, представляющего собой аддитивную помеху $\zeta(n)$ (рис. 4).

Отметим попутно, что приведенный на рис. 4 фазовый портрет помехи выявляет его хорошие статистические свойства, в частности, свойство независимости. Сравнение фазовых портретов по геометрическим параметрам (см. рис. 2–4) демонстрирует их качественное различие. Фазовые портреты дают интегральный образ характера поведения уравнивающей величины как решения соответствующего отображения (алгоритма) с учетом влияния различных дополнительных факторов. В принципе по геометрическим параметрам фазового портрета можно получить определенные количественные оценки исследуемого объекта, включая оценки его метрологических характеристик, учитывающих влияние различных дополнительных факторов. Для иллюстрации сказанного приведем в матричной форме фазовые портреты помехи и погрешности преобразования следящим алгоритмом при воздействии помехи (рис. 5). На рисунке наглядно представлена трансформация фазового портрета помехи в фазовый портрет погрешности преобразования; отражена в этом же виде зависимость помехи и погрешности преобразования при фиксированном значении входного сигнала; отчетливо просматривается эффект взаимодействия помехи и погрешности преобразования, приводящий к определенной упорядоченности исходного фазового портрета помехи.

Однако традиционно в метрологии количественную оценку соответствующих свойств ат-

$$p:=0.35323; M:=2^{13}; N:=1; d:=2^{(-13)}; s:=0.00000; g:=0.0005;$$

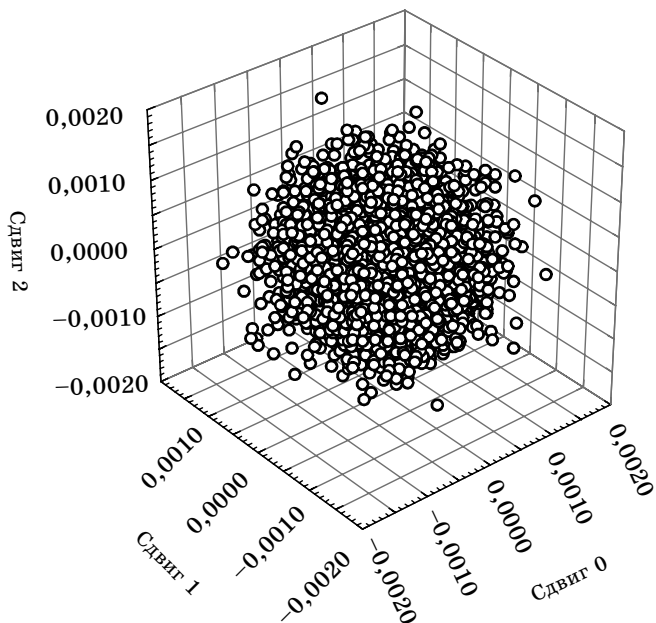
$$\text{Data}(i+1,j):=\text{Data}(i,j)+d*(\text{Sign}(\text{Data}(M+2,j))+\text{Normal}(g)-\text{Data}(i,j)+s));$$



■ Рис. 3. Трехмерный фазовый портрет, представляющий аттрактор следящего алгоритма при воздействии аддитивной помехи

$$p:0.35323; M:=2^{13}; N:=1; d:=2^{(-13)}; s:=0.00000; g:=0.0005;$$

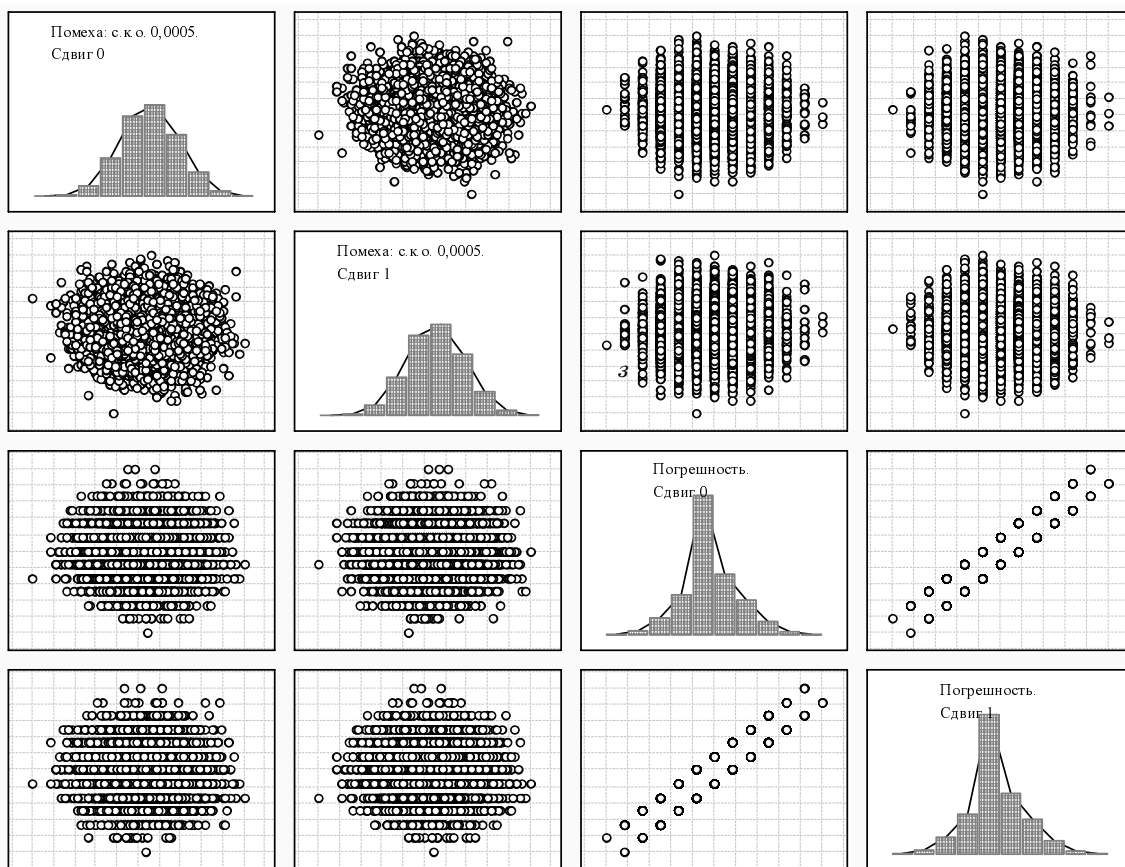
$$\text{Data}(i+1,j):=\text{Data}(i,j)+d*(\text{Sign}(\text{Data}(M+2,j)+\text{Normal}(g)-\text{Data}(i,j)+s));$$



■ Рис. 4. Трехмерный фазовый портрет, характеризующий аддитивную помеху

$$p:0.3532; M:=2^{13}; N:=1; d:=2^{(-13)}; s:=0.000001; g:=0.0005;$$

$$\text{Data}(i+1,j):=\text{Data}(i,j)+d*(\text{Sign}(\text{Data}(M+2,j)+\text{Normal}(g)-\text{Data}(i,j)+s));$$



■ Рис. 5. Матричный график, характеризующий гистограммы распределения помехи, воздействующей на сигнал, погрешность преобразования и их фазовые портреты

трактора выполняют по первым двум моментам, т. е. по математическому ожиданию и дисперсии, и значительно реже — по другим вероятностным характеристикам, например плотности распределения вероятностей. Для алгоритма следящего АЦП данные вероятностные характеристики можно получить в аналитической форме. Подробные аналитические преобразования при выводе этих характеристик, кроме плотности распределения вероятностей, сделаны в приложении к статье. Методология нахождения математического ожидания отображения (7) заключается в следующем. Усредним правую и левую части отображения (7) по помехе. В результате усреднения получим новое отображение, эквивалентное в установленном смысле исходному отображению (7):

$$\bar{E}(n) = \bar{E}(n-1) + q[2F_{\xi}[x - \bar{E}(n-1)] - 1], \quad (8)$$

где $\bar{E}(n)$ и $\bar{E}(n-1)$ — усредненные по помехе значения уравнивающей величины на n -м и $(n-1)$ -м тактах итерации.

Помеха, как правило, имеет гауссов закон распределения вероятностей с нулевым средним и, тем самым, симметрична относительно нуля. Поэтому после вычитания из входного сигнала значений правой и левой частей итерационного алгоритма (8) с последующим линейным приближением функции распределения вероятностей в малой окрестности неподвижной точки приходим к следующему линейному приближению (8):

$$V(n) = V(n-1) - q\{2[F_{\xi}(0) + w_{\xi}(0)V(n-1)] - 1\} \quad (9)$$

или

$$V(n) = V(n-1)[1 - 2qw_{\xi}(0)] - 2qF_{\xi}(0) + q.$$

Далее, учитывая, что $F_{\xi}(0) = \frac{1}{2}$, а $w_{\xi}(0) = \frac{1}{\sqrt{2\pi}\sigma_{\xi}}$,

получаем из (9)

$$V(n) = V(n-1) \left(1 - \frac{2q}{\sqrt{2\pi}\sigma_{\xi}} \right), \quad (10)$$

где $V(n) = x - E(n)$ и $V(n-1) = x - E(n-1)$.

Поскольку, как было показано выше, для определенного значения числа итераций до выхода в стационарную точку уравнивающая величина в алгоритме (7) изменяется линейно, то при достижении разности $V(n)$, например, для $n = k$ величины, при которой она становится соизмеримой с уровнем помех, начинает действовать условие сходимости, описываемое отображением (9). Применяя правило трех сигм для определения начального значения итерационного уравнения (9), получаем для $n = k$ разность $V(k) = 3\sigma_{\xi}$. Следова-

тельно, выполняя последовательно итерации в (10) для $n = k+1, k+2, \dots, m$ при начальном значении $V(k) = 3\sigma_{\xi}$, приходим к решению

$$V(m) = 3\sigma_{\xi} \left(1 - \frac{2q}{\sqrt{2\pi}\sigma_{\xi}} \right)^m.$$

Пусть $\sigma_{\xi} = \mu q$, тогда после упрощения получаем

$$V(m) = 3\mu q \left(1 - \sqrt{\frac{2}{\pi}} \frac{1}{\mu} \right)^m. \quad (11)$$

Обычно значение параметра μ таково, что $\sqrt{\frac{2}{\pi}} < \mu$, поэтому выражение (11) можно представить в приближенной, но более наглядной форме записи

$$V(m) \approx 3\mu q e^{-\sqrt{\frac{2}{\pi}} \frac{m}{\mu}}. \quad (12)$$

Из равенства (12) следует, что переходный процесс и, следовательно, его математическое ожидание в условиях воздействия помех экспоненциально стремится к нулю с ростом числа итерации. Применяя аналогичный подход, можно получить оценку дисперсии погрешности. При этом (см. приложение) с учетом равенства нулю математического ожидания финальная (или предельная при достаточно большом числе итераций m) дисперсия погрешности преобразования при воздействии определяется по формуле

$$D_{\xi}[V(m)] \approx \frac{q}{4w_{\xi}(0)} = \frac{1}{2} \sqrt{\frac{\pi}{2}} \mu q^2.$$

Если взять отношение полученной дисперсии к дисперсии помехи

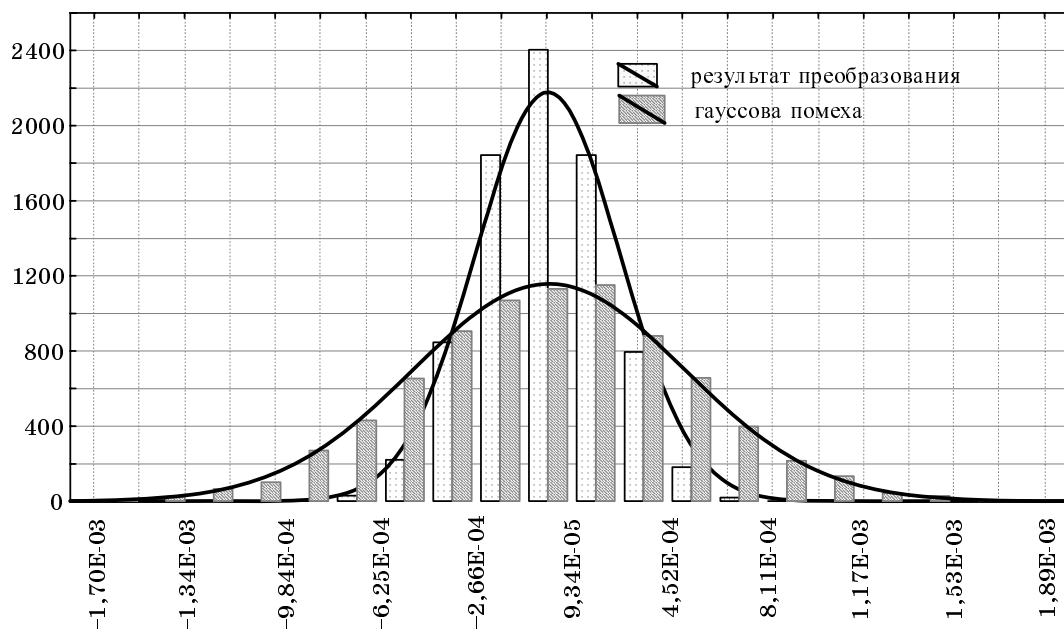
$$K \approx \frac{q}{4w_{\xi}(0)} = \frac{1}{2} \sqrt{\frac{\pi}{2}} \frac{1}{\mu},$$

то из формулы следует, что эффект подавления помехи возникает при условии, что $\mu > 0,6267$. Следовательно, для подавления помехи с фиксированным значением СКО желательно брать величину кванта q как можно меньше.

Иллюстрация эффекта подавления помехи алгоритмом следящего АЦП хорошо просматривается на рис. 6, где представлены по результатам имитационного моделирования в одном и том же масштабе гистограммы погрешности преобразования в присутствии помех и самой помехи с гауссовым законом распределения вероятностей с указанным соответствующих средних и СКО.

Аналитическая запись алгоритма работы $\Sigma\Delta$ АЦП, как отмечалось выше, позволяет сравнить

```
Data(i+1,j):=Data(i,j)+d*(Sign(Data(M+2,j)+Normal(g)-Data(i,j)+s));
Var1 = 8192*0.0002*normal(x; -6.8359E-6; 0.0003)
Var2 = 8192*0.0002*normal(x; -2.0942E-6; 0.0005)
p:=0.3532; M:=2^12; d:=0.0002;s:=0.00001; g:=0.0005;
```



■ Рис. 6. Гистограммы погрешности преобразования следающего алгоритма в присутствии помех и гауссовой помехи с параметрами, указанными на рисунке: $E(n+1) = Data(i+1, j)$; $E(n) = Data(i, j)$, $d = q = 2^{-12} \approx 0,000244$; $x = p$; $N = 12$; g соответствует СКО; s — вспомогательный параметр (математическое ожидание помехи)

его со следающим алгоритмом. Так, сравнение алгоритмов (3) и (7) показывает их аналитическую идентичность с, казалось бы, совсем незначительными отличиями. В частности, это отличие заключено в множителе перед знаковой функцией и в наличии дополнительного слагаемого $\beta\lambda$. Поэтому можно было бы предположить, что и динамические свойства данных алгоритмов будут в какой-то степени идентичны. Действительно, как это будет следовать из дальнейшего анализа, вид и, соответственно, аналитическая форма представления переходного периода у обоих алгоритмов совпадают. Заметим, что отличие в множителе, стоящем перед знаковой функцией, заключается в его размерности, так как в алгоритме (3) множитель $\beta = \alpha E_0 = E_0/f_t$ имеет размерность вольт на секунду. Поэтому следует утверждение, что чем выше частота, тем меньше величина множителя α . Если выбрать частоту f_t , равную 2^N , то получаем для $\beta = E_0/2^N = q$ почти полную идентичность следающего алгоритма и алгоритма работы $\Sigma\Delta$ АЦП первого порядка, отличающуюся только зависимостью в последнем приращении уравнивающей величины на каждом шаге итерации от уровня входного сигнала. В следающем алгоритме уравнивающая величина изменяется кратно ве-

личине кванта, а в алгоритме $\Sigma\Delta$ АЦП, согласно отображению

$$E(n) = E(n-1) + q \{ \text{sign}[x - E(n-1)] + \lambda \}, \quad (13)$$

уравнивающая величина изменяется в зависимости от величины входного сигнала пропорционально величине θq , где $q \leq \theta \leq 2q$ и $\lambda = x/E_0$.

Наличие второго параметра в отображении (13) в виде дополнительного слагаемого существенно изменяет фазовый портрет. Причем геометрическая форма аттрактора меняется в зависимости от изменения входного сигнала, т. е. параметра отображения, и тем самым форма аттрактора в отличие от просто следающего алгоритма несет информацию о величине входного сигнала. Как будет показано в части 2 статьи, изменение формы аттрактора дает новое качество алгоритму, следствием которого является значительное улучшение метрологических характеристик $\Sigma\Delta$ АЦП.

Приложение

Вычислим условное математическое ожидание следающего алгоритма без учета аддитивной помехи при условии, что входной сигнал принимает значение $m q \pm \gamma$, где γ — погрешность усечения.

Выполним для этого значения входного сигнала усреднение по случайной погрешности усечения правой и левой частей алгоритма после вычитания справа и слева фиксированного значения входного сигнала $K_x = mq$ в установившемся режиме, т. е. после переходного процесса. В результате получим (для наглядности оставим значение временного такта Δt , принимающего любую величину)

$$\begin{aligned} M_\gamma\{V(n\Delta t)\} &= \\ &= M_\gamma\left[V[(n-1)\Delta t] - q\{\text{sign}\{V[(n-1)\Delta t] \pm \gamma\}\}\right]; \\ M_\gamma\{V(n\Delta t)\} &= \\ &= M_\gamma\{V[(n-1)\Delta t]\} - qM_\gamma\{\text{sign}\{V[(n-1)\Delta t] \pm \gamma\}\}, \end{aligned}$$

где $0 \leq \gamma \leq q$, причем γ изменяется по равномерному закону с плотностью распределения вероятностей $1/q$; $V(n\Delta t) = K_x - qK(n\Delta t)$, $V[(n-1)\Delta t] = K_x - qK[(n-1)\Delta t]$ — разность кодовых эквивалентов на n -м и $(n-1)$ -м тактах итерации, при этом начальное значение для шага итерации $n=0$ определяется в соответствии с равенством $K(0) = 0$.

В переходном режиме для всех $n \leq m$ знаковая функция равна единице, а после окончания переходного режима при $qK(m\Delta t) = K_x$ с точностью до кванта $V(m\Delta t) = \pm\gamma$. Поэтому математические ожидания соответствующих значений разностей найдутся из равенств

$$\begin{aligned} M_\gamma\{V(m\Delta t)\} &= (K_x - m)q; \\ M_\gamma\{V[(m+1)\Delta t]\} &= (K_x - m)q + 2F_\gamma(0) - 1 = 0 \end{aligned}$$

с учетом того, что функция распределения погрешности усечения в нуле равна $1/2$.

Таким образом, для неподвижной точки математическое ожидание погрешности преобразования следящим алгоритмом в отсутствие помехи равно нулю.

Рассмотрим условие сходимости при воздействии аддитивной помехи, причем, помеха не зависит от номера такта итерации. В этом случае в результате аналогичных действий получаем

$$\begin{aligned} M_\eta\{V(n\Delta t)\} &= \\ &= M_\eta\left[V[(n-1)\Delta t] - q\{\text{sign}\{V[(n-1)\Delta t] + \xi + \gamma\}\}\right]; \\ M_\eta\{V(n\Delta t)\} &= \\ &= M_\eta\{V[(n-1)\Delta t]\} - qM_\eta\{\text{sign}\{V[(n-1)\Delta t] + \eta\}\}, \end{aligned}$$

так как $\eta = \xi + \gamma$. Представим математические ожидания разностей $V(n\Delta t)$ и $V[(n-1)\Delta t]$ в виде

$$\begin{aligned} M_\eta\{V(n\Delta t)\} &= V(n\Delta t) + M_\eta\{\eta\}; \\ M_\eta\{V[(n-1)\Delta t]\} &= V[(n-1)\Delta t] + M_\eta\{\eta\}, \end{aligned}$$

тогда

$$\begin{aligned} V(n\Delta t) &= V[(n-1)\Delta t] - q\{2F_\eta[V[(n-1)\Delta t]] - 1\} = \\ &= V[(n-1)\Delta t] - q\{2F_\eta(0) + 2w_\eta(0)V[(n-1)\Delta t] - 1\} = \\ &= V[(n-1)\Delta t] - 2qw_\eta(0)V[(n-1)\Delta t] = \\ &= V[(n-1)\Delta t][1 - 2qw_\eta(0)]. \end{aligned}$$

В результате

$$\begin{aligned} V[(n+m)\Delta t] &= V(n\Delta t)[1 - 2qw_\eta(0)]^m \approx \\ &\approx V(n\Delta t)e^{-2mqw_\eta(0)} \end{aligned}$$

при условии, что $2qw_\eta(0) < 1$.

Средний квадрат разности $V(n\Delta t) = K_x - q \times K(n\Delta t)$ следящего алгоритма без учета воздействия аддитивной помехи определяется следующим образом. Пусть разность $V[(n-1)\Delta t]$ равна погрешности усечения, т. е. итерационный процесс вышел в неподвижную точку и разность между уравновешивающей величиной и входным сигналом равна погрешности усечения, тогда

$$\begin{aligned} M_\gamma\{V^2(n\Delta t)\} &= M_\gamma\{\gamma - q[\text{sign}(\gamma)]\}^2 = \\ &= M_\gamma\{\gamma^2\} - 2qM_\gamma\{\gamma\text{sign}(\gamma)\} + q^2M_\gamma\{\text{sign}(\gamma)\}^2. \end{aligned}$$

Заметим, что

$$M_\gamma\{\text{sign}(\gamma)\}^2 = \int_0^q \frac{d\gamma}{2q} + \int_{-q}^0 \frac{d\gamma}{2q} = \frac{1}{2} + \frac{1}{2} = 1,$$

поэтому

$$M_\gamma\{V^2(n\Delta t)\} = M_\gamma\{\gamma^2\} - 2qM_\gamma\{\gamma\text{sign}(\gamma)\} + q^2.$$

Поскольку для погрешности усечения выполняется равенство

$$M_\gamma\{\gamma^2\} = \int_{-q}^q \frac{\gamma^2}{2q} d\gamma = \frac{2q^3}{6q} = \frac{q^2}{3},$$

то

$$M_\gamma\{V^2(n\Delta t)\} = \frac{q^2}{3} - 2qM_\gamma\{\gamma\text{sign}(\gamma)\} + q^2.$$

Учитывая, что $\gamma\text{sign}(\gamma) = |\gamma|$, $M_\gamma\{\gamma\text{sign}(\gamma)\} =$

$$= \int_{-q}^q \frac{|\gamma|}{2q} d\gamma = \int_0^q \frac{\gamma}{q} d\gamma = \frac{q}{2}, \text{ окончательно получаем для}$$

среднего квадрата методической погрешности преобразования равенство

$$M_{\gamma} \{V^2(n\Delta t)\} = \frac{q^2}{3} - 2q\frac{q}{2} + q^2 = \frac{q^2}{3}.$$

В анализируемом случае усреднение осуществляется только по погрешности квантования, соответствующей погрешности усечения, поэтому здесь должен учитываться тот факт, что при выходе алгоритма (3) в стационарную точку для разности $V(n\Delta t)$ справедливо равенство $V(n\Delta t) = K_x[(n-1)\Delta t] + \gamma - qK[(n-1)\Delta t] = \gamma$. Так как эта погрешность распределена равномерно в интервале q , то ее средний квадрат равен $q^2/3$.

Рассмотрим вариант оценки среднего квадрата с учетом воздействия помехи ξ при учете равенств $V[(n-1)\Delta t] = K_x - qK[(n-1)\Delta t]$ и $V(n\Delta t) = K_x - q \times K(n\Delta t)$, $x = K_x + \gamma$. Зафиксируем значение входного сигнала с точностью до погрешности квантования. Тогда итерационное уравнение при вычитании справа и слева погрешности усечения можно представить в виде

$$\begin{aligned} &M_{\eta} \{V^2(n\Delta t)\} = \\ &= M_{\eta} \left\{ V[(n-1)\Delta t] - q \left\{ \text{sign} \left\{ V[(n-1)\Delta t] + \xi + \gamma \right\} \right\}^2, \right. \\ &\quad \left. \eta = \xi + \gamma; \right. \\ &M_{\eta} \{V^2(1\Delta t)\} = \\ &= M_{\eta} \{V^2(0)\} - 2qM_{\eta} \{V(0)\} \left\{ \text{sign} \{V(0) + \eta\} \right\} + \\ &\quad + q^2 M_{\eta} \left\{ \text{sign}^2 \{V(0\Delta t) + \eta\} \right\}. \end{aligned}$$

Заметим, что

$$M_{\gamma} \left\{ \text{sign} \{V(0\Delta t) + \eta\} \right\}^2 = 1 \text{ и } M_{\eta} \{V^2(0)\} = V^2(0).$$

При достижении разности $V(n\Delta t)$, например для $n = k$, определенной величины, при которой она становится соизмеримой с уровнем помех, начинает действовать условие сходимости, описываемое отображением (9). Применяя правило трех сигм для определения начального значения итерационного уравнения (9), получаем для $n = k$ раз-

ность $V(k\Delta t) = 3\sigma_{\eta}$. Следовательно, выполняя последовательно итерации для $n = k + 1, k + 2, \dots, m$ при начальном значении $V(k\Delta t) = 3\sigma_{\eta}$, приходим к решению

$$\begin{aligned} &M_{\gamma} \{V^2[(k+1)\Delta t]\} = \\ &= V^2(k\Delta t) - 2qV(k\Delta t)M_{\eta} \left\{ \text{sign} \{V(k\Delta t) + \eta\} \right\} + q^2 \end{aligned}$$

и

$$\begin{aligned} &M_{\eta} \{V^2[(k+1)\Delta t]\} = \\ &= V^2(k\Delta t) - 2qV(k\Delta t) \left\{ 2F_{\eta} [V(k\Delta t)] - 1 \right\} + q^2 = \\ &= V^2(k\Delta t) - 2qV^2(k\Delta t) \left\{ 2F_{\eta}(0) + 2w_{\eta}(0) - 1 \right\} + \\ &\quad + q^2 = V^2(k\Delta t) - 4qw_{\eta}(0)V^2(k\Delta t) + q^2 = \\ &\quad = V^2(k\Delta t) \left[1 - 4qw_{\eta}(0) \right] + q^2; \\ &M_{\eta} \{V^2[(k+2)\Delta t]\} = V^2(k\Delta t) \left[1 - 4qw_{\eta}(0) \right]^2 + \\ &\quad + q^2 \left[1 - 4w_{\eta}(0)q \right] + q^2 \dots \\ &M_{\eta} [V^2(m\Delta t)] = V^2(k\Delta t) \left[1 - 4qw_{\eta}(0) \right]^m + \\ &\quad + q^2 \sum_{i=0}^{m-1} \left[1 - 4w_{\eta}(0)q \right]^i; \\ &M_{\eta} [V^2(m\Delta t)] = 9\sigma_{\eta}^2 \left[1 - 4qw_{\eta}(0) \right]^m + \\ &\quad + q^2 \frac{1 - \left[1 - 4w_{\eta}(0)q \right]^m}{4w_{\eta}(0)q}; \\ &\lim_{m \rightarrow \infty} M_{\eta} [V^2(m\Delta t)] = \frac{q}{4w_{\eta}(0)}, \end{aligned}$$

поскольку $4qw_{\eta}(0) < 1$.

Следует заметить, что в общем случае суммарная погрешность η имеет не совсем симметричную функцию распределения вероятностей, тем не менее, она достаточно близка к симметричной функции. Поэтому погрешность используемого линейного приближения функции распределения допустима для технических расчетов.

(Окончание следует)

Литература

1. Интегральные микросхемы: микросхемы для аналого-цифрового преобразования и средств мультимедиа. М.: ДОДЕКА. 1996. Вып. 1. 384 с.
2. Хорвец П., Хилл У. Искусство схемотехники: Пер. с англ. 5-е изд., перераб. М.: Мир, 1998. 704 с.
3. Информационный бюллетень компании Analog Devices. Вып. 8. 2006. www.ekis.kiev.ua
4. Голуб В. Цифровая обработка сигналов: сигма-дельта АЦП // Электроника: Наука, Технология, Бизнес. 2001. Вып. 4. С. 22–26.
5. Голуб В. С. Сигма-дельта модуляторы и АЦП // Радиоаматор. Киев, 2000. № 8. С. 36–37.
6. Голуб В. С. Сигма-дельта модуляторы и АЦП // Радиоаматор. Киев, 2001. № 2. С. 27–28.
7. Романов О. Обзор новых АЦП компании Analog Devices // Электронные компоненты. 2004. № 2. С. 34–35. http://www.eltech.spb.ru
8. Сигма-дельта АЦП фирмы Analog Devices // Электронные компоненты и системы. Киев: VD MAIS. Май 1996. С. 20–25.

9. **Тихонов Э. П.** Аналитико-имитационное исследование и оптимизация алгоритмов аналого-цифрового преобразования в условиях воздействия помех // Информационно-управляющие системы. 2007. № 2. С. 12–21.
10. **Тихонов Э. П.** Стохастический, пространственный аналого-цифровой преобразователь и его связь с нейронными структурами // Информационно-управляющие системы. 2007. № 1 (26). С. 8–18.
11. **Малинецкий Г. Г., Потапов А. Б.** Нелинейная динамика и хаос. Основные понятия: Учеб. пособие. М.: Ком Книга, 2006. 240 с.
12. **Чуличков А. И.** Математические модели нелинейной динамики. 2-е изд., испр. М.: Физматлит, 2003. 296 с.
13. **Малинецкий Г. Г.** Хаос. Структуры. Вычислительный эксперимент: Введение в нелинейную динамику. 3-е изд., стер. М.: Едиториал УРСС, 2002. 256 с.
14. **Кроновер Р.** Фракталы и хаос в динамических системах. М.: Техносфера, 2006. 488 с.
15. **Берже П., Помо И., Видаль К.** Порядок в хаосе. О детерминистском подходе к турбулентности: Пер. с франц. М.: Мир, 1991. 386 с.
16. **Мун Ф.** Хаотические колебания: Вводный курс для научных работников и инженеров: Пер. с англ. М.: Мир, 1990. 312 с.
17. **Кузнецов С. П.** Динамический хаос (курс лекций): Учеб. пособие для вузов. 2-е изд., перераб. и доп. М.: Физматлит, 2006. 356 с.
18. **Островерхов В. В.** Динамические погрешности аналого-цифровых преобразователей. Л.: Энергия, 1975. 213 с.
19. **Стахов А. П.** Введение в алгоритмическую теорию измерения. М.: Сов. радио, 1977. 288 с.
20. **Бахтиаров Г. Д., Малинин В. В., Школин В. П.** Аналого-цифровые преобразователи / Под ред. Г. Д. Бахтиарова. М.: Сов. радио, 1980. 289 с.
21. **Гитис Э. И., Пискулов Е. А.** Аналого-цифровые преобразователи. М.: Энергоатомиздат, 1981. 360 с.
22. **Шлыков Г. П.** Измерение параметров интегральных ЦАП и АЦП. М.: Радио и связь, 1985. 128 с.
23. **Алиев Т. М., Дамиров Д. И., Шекиханов А. М.** Введение в алгоритмические основы цифровых измерений // Измерения, контроль, автоматизация. 1983. № 3. С. 3–11.
24. **Тихонов Э. П.** Исследование помехоустойчивости аналого-цифрового преобразования методами адаптивного усреднения // Электронное моделирование. 1983. № 1.
25. Генетические алгоритмы. <http://algotist.manual.ru/ai/ga/>
26. **Мирский Г. Я.** Характеристики стохастической взаимосвязи и их измерения. М.: Энергоатомиздат, 1987. 320 с.

УДК 519.174

ОДНОЗНАЧНО РАСКРАШИВАЕМЫЕ ГРАФЫ С МИНИМУМОМ РЕБЕР

Ю. П. Великохатко,

научный сотрудник

НИИ электротехнических устройств

А. А. Миронов,

канд. техн. наук, доцент

Санкт-Петербургский университет информационных технологий, механики и оптики

Построен граф с количеством ребер, минимально допустимым для однозначно K -раскрашиваемого графа. Показано, как при помощи операций двух типов указанный граф можно преобразовать в множество всех графов этого класса (для произвольно заданного числа вершин). Доказано, что все такие графы $(K - 1)$ -связны. Найдено выражение для числа ребер в таких графах, а также даны уравнения, полезные для практических целей, в частности, для различных задач оптимизации.

In this paper a uniquely K -colorable graph with minimum possible number of edges is being constructed. It is shown that the graph can be transformed into the set of all possible graphs of that kind (for any specified number of vertices) using only two kinds of transformations. It has been proven that all such graphs are $(K - 1)$ -connected. An expression for the number of edges in such graphs is derived as well as some equations that have practical applications, in particular, in solving various optimization problems.

Введение

Известно, что в определении однозначно раскрашиваемых графов имеется условие неизменности разбиения на части множества вершин графа. Авторы включили это вполне естественное условие стабильности разбиения в определение многодольных графов, что позволило оба класса упомянутых графов считать одним классом. Такой подход оправдан, поскольку разбиение множества вершин графа на K частей порождает K классов эквивалентности, которые называются одноцветными классами в однозначно K -раскрашиваемом графе или долями в графе K -дольном. В тексте статьи используются оба термина.

Открытый авторами новый класс графов с минимумом ребер (однозначно K -раскрашиваемых или K -дольных) представлен специально построенным графом определенной структуры, который в качестве исходного расширен при помощи двух операций до множества всех графов этого класса.

Основная часть

На сегодняшний день в теории графов определение K -дольного графа допускает для одного и того же графа различные толкования числа долей и количественного состава отдельной доли. Таково,

например, определение 2-дольного графа и аналогичных ему K -дольных графов в работе [1].

Использование понятия разбиения в определении многодольного графа требует уточнения способа сохранения частей разбиения в неизменном количественном составе. В полном K -дольном графе это требование уже выполнено: в нем отсутствует неоднозначность в идентификации той или иной доли. Действительно, каждая доля содержит вершины одной и той же степени, равной количеству вершин в остальных долях. При этом доли четко разграничены одна от другой несмежностью вершин внутри доли и наличием ребер между вершинами различных долей.

Поэтому полный K -дольный граф можно использовать в качестве опорного при формулировке определения K -дольного.

K -дольным будем называть граф с множеством вершин, разбитым на подмножества, именуемые долями, в пределах каждой из которых все вершины взаимно-несмежны, а ребрами соединены только вершины различных долей; при этом восстановление ребер до их максимального количества (до полного K -дольного графа) не меняет количества и состава долей.

Очевидно, что в K -дольном графе существует некоторое минимально допустимое количество ре-

бер, служащее границей K -дольности. В частности, это следует из того, что восстановление ребер в пустом графе не может быть выполнено однозначно.

Тождественность K -дольности и однозначной K -раскрашиваемости графов обусловлена отношением эквивалентности, определяющим структуру этих графов.

В первом случае части разбиения именуются долями, во втором — одноцветными классами, в обоих — это классы эквивалентности.

Используя вышеприведенное определение K -дольного графа, авторы открыли класс K -дольных графов с минимально допустимым количеством ребер, который принципиально ничем не отличается от класса однозначно K -раскрашиваемых графов с минимумом ребер.

Известно выражение для минимального числа ребер, используемое для максимальных планарных графов [2, теорема 12.21]:

$$\sum_{i < j} (p_i + p_j - 1).$$

Установим для этого выражения пределы изменения i, j от 1 до K , где K — количество частей разбиения всех p вершин некоторого графа с минимумом ребер, когда каждая часть насчитывает p_i или p_j вершин. Обозначим минимальное количество ребер q_{\min} и выполним необходимые преобразования:

$$\begin{aligned} q_{\min} &= \sum_{1 \leq i < j \leq K} (p_i + p_j - 1) = (p_1 + p_2 - 1) + \dots + \\ &+ (p_2 + p_3 - 1) + \dots + (p_{K-1} + p_K - 1) = \\ &= (K-1)p_1 + (K-1)p_2 + \dots + (K-1)p_K - \binom{K}{2} = \\ &= (K-1) \sum_{i=1}^K p_i - \binom{K}{2}. \end{aligned}$$

Поскольку $\sum_{i=1}^K p_i = p$, запишем

$$q_{\min} = (K-1)p - \binom{K}{2}. \quad (1)$$

Полученное выражение (1) естественно интерпретируется как минимально допустимое количество ребер в однозначно K -раскрашиваемом графе с p вершинами, где в каждом из K одноцветных классов насчитывается p_i вершин.

Получим формулу (1) иначе, учитывая однозначную K -раскрашиваемость любого полного K -дольного графа. Для этого используем формальное тождество

$$p_1 p_2 - (p_1 - 1)(p_2 - 1) \equiv p_1 + p_2 - 1. \quad (2)$$

Подведя выражение (2) под знак суммы, получим

$$\sum p_i p_j - \sum (p_i - 1)(p_j - 1) \equiv \sum (p_i + p_j - 1),$$

где $1 \leq i < j \leq K$.

Поскольку справа от знака тождества выражение (1), то можно записать:

$$q_{\min} = \sum p_i p_j - \sum (p_i - 1)(p_j - 1) = (K-1)p - \binom{K}{2}, \quad (3)$$

где $1 \leq i < j \leq K$, $p = \sum_{i=1}^K p_i$.

В выражении (3) слева от второго знака равенства разность количеств ребер двух полных K -дольных графов, причем каждая доля второго уменьшена на одну вершину.

Заодно найдем разность численных значений характеристик связности тех же двух графов. Связность первого равна $p - p_i^{\max}$, где p_i^{\max} — наибольшая доля в графе. Связность второго графа соответственно будет $(p - K - p_i^{\max} + 1)$. Их разность равна $K - 1$.

Построим граф с количеством ребер согласно (1), следуя в точности выражению (3). Для этого в полном K -дольном графе с p вершинами и с p_i вершинами в каждой доле следующим образом устраним все излишние ребра: в каждой доле оставим по одной вершине с первоначальной степенью $(p - p_i)$ со всеми инцидентными ей ребрами. Все остальные ребра уберем. Результат соответствует вычитанию количества ребер второго графа из первого согласно (3).

Построенный граф является K -дольным и имеет минимально допустимое количество ребер, сохраняющее части разбиения в неизменном виде. Этот же граф можно назвать однозначно K -раскрашиваемым с минимумом ребер.

Проверим это, исследуя структуру графа. Каждый одноцветный класс в нем, имея p_i вершин, соединен $[(p - p_i) + (p_i - 1)(K - 1)]$ ребрами со всеми остальными вершинами графа. Каждая пара одноцветных классов имеет $(p_i + p_j - 1)$ общих ребер, т. е. представляет собой подграф, порожденный объединением двух одноцветных классов. Этот подграф — дерево. Структура такого дерева представляет собой звезду с двумя центральными вершинами.

Поскольку в правых частях выражений (1) и (3) нет p_i , нетрудно сделать вывод, что формула (1) справедлива для всех разбиений p на K частей.

Найдем графы, соответствующие этим разбиениям. При этом K вершин в графе со степенью $(p - p_i)$ назовем особыми. Все остальные $p - K$ вершин имеют степень $(K - 1)$, являющуюся минимально допустимой, обеспечивающей необходимое условие однозначной K -раскраски графа и его

$(K - 1)$ -связности. Количество вершин степени $(K - 1)$ максимально в графе с такой структурой.

Перебросим конец произвольно выбранного ребра в исходном графе с одной особой вершины на особую вершину того же цвета, что и другой конец выбранного ребра. Вершину степени $(K - 1)$, которой инцидентен другой конец этого ребра, переместим в ту часть разбиения (одноцветный класс), где находится особая вершина с отсоединенным концом ребра. В результате получится граф с той же структурой, что исходный, но с другим количественным составом частей разбиения. Повторение такой операции позволяет получить все множество графов со структурой исходного, т. е. с максимумом вершин степени $(K - 1)$, соответствующее всем разбиениям p на K частей.

Следующая операция преобразует каждый граф, принадлежащий тому или иному разбиению, в множество графов, относящихся только к этому разбиению.

Разгрузим от ребер особые вершины, перебрасывая освободившиеся концы на вершины степени $(K - 1)$, принадлежащие той же части разбиения, что и освобождаемая от ребер особая вершина. Эта операция позволяет получить из экстремального распределения вершин в каждом одноцветном классе (с максимумом вершин степени $(K - 1)$ все остальные их распределения.

В результате описанных операций двух типов получено множество всех графов, отвечающих (1) и (3). Их однозначная K -раскрашиваемость обусловлена однозначной 2-раскрашиваемостью деревьев, входящих в структуру графов. Дерево-подграф не допускает частичной перекраски одноцветного класса, служащей признаком неоднозначной раскрашиваемости.

Доказательство однозначной 2-раскрашиваемости любого дерева можно извлечь из известной теоремы Кёнига [3]. Несмотря на то что в ней нет упоминания об этой однозначности, метод четносоединимых вершин, использованный в теореме, приводит к двум классам эквивалентности, которые в дереве являются одноцветными классами однозначно 2-раскрашиваемого графа.

В любом K -дольном графе с минимумом ребер имеется $\binom{K}{2}$ подграфов-деревьев, порожденных объединениями пар подмножеств, являющихся частями разбиения множества вершин. Характеристикой связности графа является наименьшее количество вершинно-непересекающихся цепей, связывающих любые две его вершины. Если эти две вершины принадлежат дереву, то между ними имеется требуемая цепь. Нетрудно убедиться, что цепь также имеется, если такая пара вершин принадлежит двум деревьям с общей долей, причем неважно, каким двум долям из трех принадлежит эта пара.

Поскольку степень вершины, равная $(K - 1)$, является минимально допустимой степенью для рассматриваемого класса графов и ограничивает

сверху количественное значение связности, убедимся, что для любой пары вершин найдется $K - 1$ вершинно-непересекающихся цепей. Для этого докажем теорему.

Теорема. Всякий K -дольный граф с минимумом ребер $(K - 1)$ -связен.

Доказательство. Рассмотрим вершины упомянутой пары с точки зрения их принадлежности одной или различным долям графа. Пусть обе вершины принадлежат одной и той же доле. Тогда имеется $K - 1$ деревьев с этой долей и, следовательно, $K - 1$ цепей между рассматриваемыми вершинами. Пусть теперь обе вершины пары принадлежат различным долям. Тогда, во-первых, они принадлежат дереву, образованному из этих долей, что обеспечивает одну цепь между ними. Во-вторых, в графе имеется $K - 2$ пар деревьев, где каждая пара содержит деревья с интересующими нас вершинами и с общей долей. Каждая такая пара деревьев обеспечивает необходимую цепь. Таким образом, между вершинами разных долей имеется $K - 1$ цепь.

Следовательно, между любой парой вершин графа имеется $K - 1$ вершинно-непересекающихся цепей. Доказательство закончено.

Проиллюстрируем изложенное примером, приняв $K = 4$ и пронумеровав доли 1, 2, 3, 4.

Пусть пара интересующих нас вершин принадлежит доле 1. Связность обеспечена деревьями 12, 13, 14.

Пусть вершины находятся в долях 1 и 2. Связность обеспечена деревом 12 и парами деревьев 13 и 23, 14 и 24.

С целью практического построения и анализа однозначно K -раскрашиваемых графов предлагаются без вывода нижеследующие соотношения. Это уравнения возрастающих степеней вершин для

графов с числом ребер $q_{\min} = (K - 1)p - \binom{K}{2}$ и для выделенного в таком графе одноцветного класса (доли).

Уравнение для графа:

$$\sum_{i=K-1}^{i=\max} (i - K + 2)p_i = K(p - K + 1), \quad (4)$$

где $K - 1 \leq i \leq p - 1$; K — количество частей разбиения; p — количество всех вершин в графе; p_i — количество вершин степени i в графе.

Уравнение для выделенного в графе одноцветного класса (доли):

$$\sum_{i=K-1}^{i=\max} (i - K + 2)p_i = (p - K + 1), \quad (5)$$

где $K - 1 \leq i \leq p - 1$; K — количество частей разбиения (одноцветных классов, доль); p — количество всех вершин в графе; p_i — количество вершин степени i в выделенном одноцветном классе (доле).

Примечание. p_i в выражениях (4) и (5) не одно и то же.

Сумма степеней всех вершин выделенного одноцветного класса (доли)

$$\sum d = (K - 2)p_j + p - K + 1, \quad (6)$$

где p_j – количество всех вершин в выделенном одноцветном классе.

Заключение

Однозначно K -раскрашиваемые графы с минимальным количеством ребер и значением связности $(K - 1)$ могут быть использованы в различных задачах оптимизации, например, коммуникационных сетей.

Литература

1. Емеличев В. А., Мельников О. И., Сарванов В. И., Тышкевич Р. И. Лекции по теории графов. М.: Наука, 1990. С. 11.

2. Харари Ф. Теория графов. М.: Мир, 1973.
3. Зыков А. А. Основы теории графов. М.: Наука, 1987. С. 98.



Крук Е. А., Овчинников А. А.

Методы программирования и прикладные алгоритмы: учебное пособие / Е. А. Крук, А. А. Овчинников; ГУАП. — СПб., 2007. — 238 с.: ил. ISBN 978-5-8088-0237-7

Учебное пособие представляет собой курс лекций, многие годы читающийся студентам, обучающимся по направлению «Информационная безопасность», «Информационные системы», «Информатика и вычислительная техника» в Санкт-Петербургском государственном университете аэрокосмического приборостроения и в Санкт-Петербургском государственном политехническом университете.

Предназначено для студентов, обучающихся по специальности 090104, а также может быть использовано для самостоятельной работы и при выполнении заданий по НИР.

УДК 621.396.69

СИНТЕЗ ДИСКРИМИНАТОРА ПЕЛЕНГАЦИОННОЙ СИСТЕМЫ С ЭЛЛИПСООБРАЗНЫМ РАСПОЛОЖЕНИЕМ ТОЧЕК ПРИЕМА

В. С. Павлов,

доцент, канд. техн. наук

М. В. Савинов,

ассистент

Санкт-Петербургский государственный университет аэрокосмического приборостроения

Проводится двумерный статистический синтез оптимального трехканального дискриминатора пеленгационной системы для случая, когда ее пространственные характеристики могут быть представлены через совокупность точек приема, расположенных по окату эллипса в плоскости пеленгования. Исследованы условия, при которых формируемые дискриминатором оценки угловых расхождений в азимутальной и угломестной плоскостях независимы между собой, с учетом поворота совокупности точек приема на произвольный угол.

The paper gives a two-dimensional statistical synthesis of the optimum three-channel discriminator for the direction finding system in the case when its spatial characteristics can be presented as the set of reception points located on an ellipse in the plane of direction finding. We investigate the conditions under which estimations of angular mismatches formed by the discriminator in azimuthal and elevation planes are independent between themselves, taking into account the rotation of reception points through an arbitrary angle.

Развитие антенной техники и средств обработки пространственно-временных сигналов расширяет возможности применения оптимальных методов теории управления в задачах определения параметров движения локационных объектов. В связи с этими возможностями немалый интерес представляет разработка вопросов построения и совершенствования систем совместного измерения угловых координат локационного объекта в условиях, когда представление плоскости пеленгования может быть, вообще говоря, произвольным.

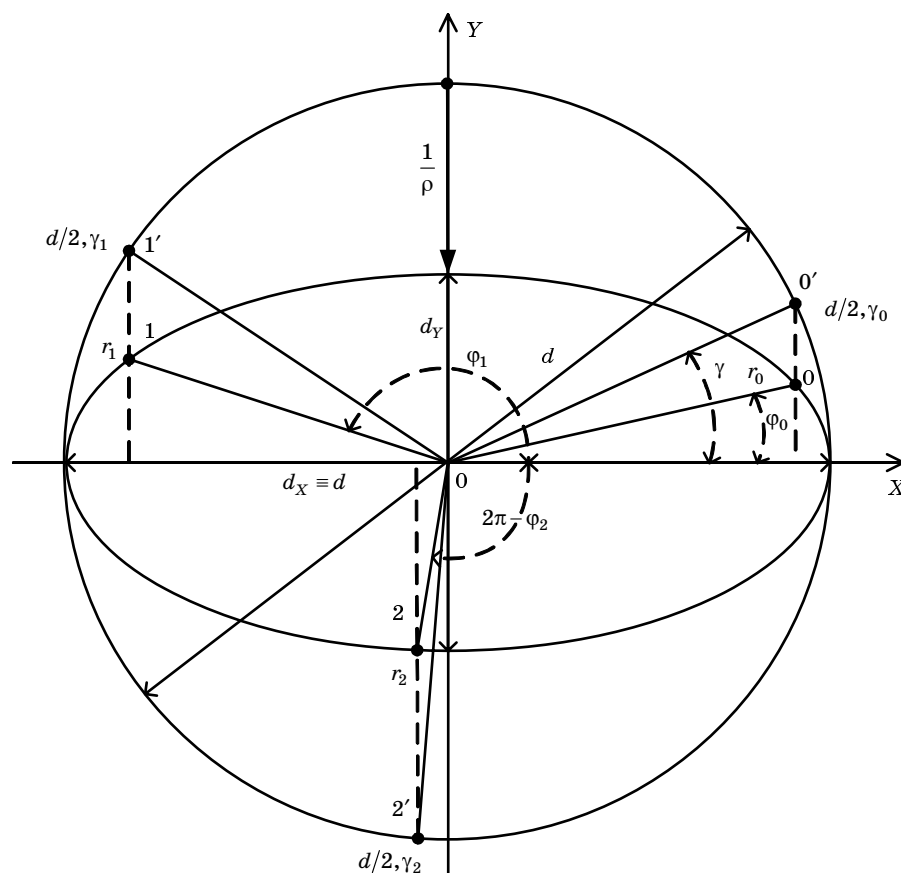
Методика статистического синтеза пеленгационных систем при различных представлениях плоскости пеленгования и информационного пространственно-временного сигнала, поступающего от локационного объекта, может существенно различаться в зависимости от выбранных моделей и выполняемых аналитических действий [1]. Примером этому является синтез, проведенный в работе [2] для случая произвольного числа пространственных отсчетов входного информационного сигнала, соответствующих отдельным точкам приема и расположенных по окружности на равном расстоянии друг от друга. Наиболее важным результатом этого синтеза

является трехканальный дискриминатор пеленгационной системы, который при структурном минимуме, необходимом для совместного измерения двух угловых координат, обеспечивает наилучшие точностные показатели их измерения [3].

Очевидный интерес представляет синтез подобного дискриминатора в более общих условиях, когда измерения двух угловых координат в ортогональных плоскостях, например азимута и угла места, характеризуются различными требованиями потенциальной точности.

В результате исследования процесса совместного измерения двух неэнергетических параметров с различной потенциальной точностью [4] установлено, что оптимальным в данном случае является положение отсчетов входного двумерного информационного сигнала, которое представлено на рис. 1. Для пеленгационной системы это означает прием и обработку пространственно-временного локационного сигнала в трех точках — 0, 1 и 2, геометрическое место всех возможных положений которых образует окат эллипса.

Следует отметить, что в случае равной потенциальной точности измерения азимута и угла мес-



■ Рис. 1. Полярные координаты трех точек приема, расположенных по окружности и по окату эллипса

та три точки приема $0', 1', 2'$ находятся на равном расстоянии друг относительно друга, а геометрическое место их возможных положений описывает окружность диаметром d в плоскости пеленгования XOY . При сжатии данной окружности в $1/\rho$ раз по оси OY и преобразовании ее в эллипс изменяется координата Y каждой точки приема. В статье [4] доказывается, что преобразование, проводимое подобным образом, не приводит к возникновению взаимной корреляции между двумя измеряемыми параметрами, при этом потенциальная точность измерения каждого из них зависит только от размеров эллипса d_x и d_y по соответствующим координатным осям OX и OY и отношения сигнал/шум.

На основе графического представления плоскости пеленгования (см. рис. 1) нетрудно определить полярные координаты каждой (n -й) точки приема — ее долготу φ_n и широту r_n , используя каноническое уравнение эллипса. Для общности дальнейших аналитических действий следует рассмотреть также и возможный поворот всей совокупности точек приема на некоторый произвольный угол γ , равный долготе 0 -й точки приема при $d_x = d_y$. С учетом этого для координат n -й точки приема ($n = 0, 1, 2$) запишем

$$\varphi_n = \arctg(\rho \operatorname{tg} \gamma_n); \quad (1)$$

$$r_n = \frac{d_y}{2\sqrt{(\rho^2 - 1)\cos^2 \varphi_n + 1}}, \quad (2)$$

где $\rho = d_x/d_y$ — отношение осей эллипса; $\gamma_n = \gamma + 2\pi n/3$ — смещение по долготе n -й точки приема.

Используя описанное представление точек приема, рассмотрим процесс пеленгования локационного объекта $ЛО$, характеризуемого азимутом α и углом места β в декартовой системе координат $\langle X, Y, Z \rangle$ (рис. 2). Ось OZ данной системы координат совпадает с равносигнальным направлением, которое соответствует опорным значениям азимута α_0 и угла места β_0 . При этом дискриминатор пеленгационной системы одновременно выделяет азимутальное $\Delta\alpha = \alpha - \alpha_0$ и угломестное $\Delta\beta = \beta - \beta_0$ рассогласования.

Входной пространственно-временной сигнал дискриминатора пеленгационной системы $y(\alpha, \beta, t)$, содержащий информацию об угловых координатах локационного объекта α и β , а также опорный пространственно-временной сигнал дискриминатора $x(\alpha_0, \beta_0, t)$ могут быть представлены в виде

произведения двух сомножителей, описывающих соответственно их временные и пространственные характеристики:

$$\begin{aligned} y(\alpha, \beta, t) &= s(\alpha, \beta)y(t); \\ x(\alpha_0, \beta_0, t) &= s(\alpha_0, \beta_0)x(t), \end{aligned} \quad (3)$$

где $y(t)$ и $x(t)$ — входной и опорный временные сигналы; $s(\alpha, \beta)$ и $s(\alpha_0, \beta_0)$ — пространственные сомножители входного и опорного сигналов, представляющие собой трехэлементные векторы-столбцы.

Элементы векторов $s(\alpha, \beta)$ и $s(\alpha_0, \beta_0)$ наиболее просто представляются в случае, когда амплитудные характеристики направленности точек приема можно не учитывать в силу их идентичности. При этом, исходя из геометрических построений, представленных на рис. 1 и 2, для n -го элемента вектора $s(\alpha, \beta)$ запишем

$$s_n(\alpha, \beta) = \exp \left\{ j \frac{2\pi r_n}{\lambda} (\sin \alpha \cos \varphi_n + \sin \beta \sin \varphi_n) \right\}, \quad (4)$$

где λ — длина волны. Аналогичное (4) выражение с точностью до замены переменных α и β на α_0 и β_0 описывает n -й элемент вектора опорного пространственно-временного сигнала $s(\alpha_0, \beta_0)$.

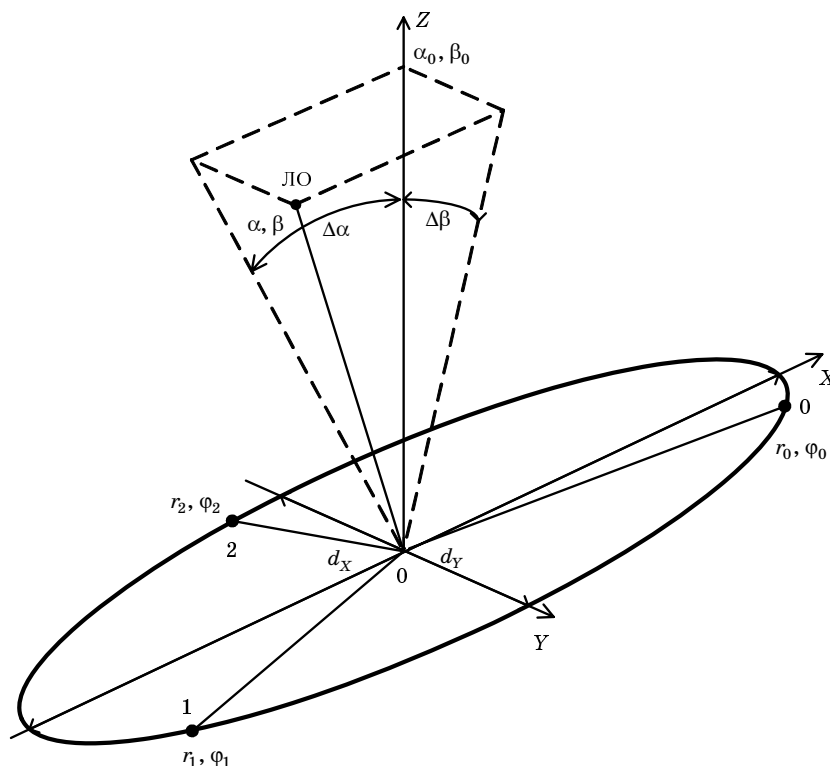
Чтобы определить процедуру, выполняемую трехканальным дискриминатором пеленгационной системы, необходимо искать решение, которое одно-

временно обеспечивает эффективность оценок рассогласований по азимуту и по углу места. Для этого воспользуемся известной методикой синтеза дискриминатора [5, 6], определяющей его структуру и параметры через отношение первой и второй производных функции правдоподобия, аргументы которой — угловые рассогласования $\Delta\alpha$ и $\Delta\beta$. В качестве аналога функции правдоподобия целесообразно использовать квадрат модуля сигнальной составляющей корреляционного интеграла, исходя из соображений, изложенных в работе [5], поскольку данная величина не зависит от случайной разности фаз между входным и опорным сигналами.

Используя пространственно-временное представление входного и опорного сигналов (3), определим их комплексный корреляционный интеграл, согласно [5, 6], в виде

$$Z(\Delta\alpha, \Delta\beta) = \frac{1}{2} \int_{-\infty}^{\infty} y^T(\alpha, \beta, t) x^*(\alpha_0, \beta_0, t) dt. \quad (5)$$

Этот интеграл описывает преобразование вектора принимаемого входного сигнала $y(\alpha, \beta, t)$ в тракте трехканального пространственно-временного приемного устройства, а его полезная составляющая $Z_S(\Delta\alpha, \Delta\beta)$ представляет собой сумму трех частных корреляционных интегралов $Z_n(\Delta\alpha, \Delta\beta)$, т. е. $Z_S(\Delta\alpha, \Delta\beta) = Z_0(\Delta\alpha, \Delta\beta) + Z_1(\Delta\alpha, \Delta\beta) + Z_2(\Delta\alpha, \Delta\beta)$. При этом n -й частный корреляционный интеграл определяется выражением



■ Рис. 2. Локационный объект в системе координат пеленгационной системы с эллипсообразным расположением точек приема

$$Z_n(\Delta\alpha, \Delta\beta) = s_n(\alpha, \beta) s_n^*(\alpha_0, \beta_0) \frac{1}{2} \int_{-\infty}^{\infty} y(t) x^*(t) dt = s_n(\alpha, \beta) s_n^*(\alpha_0, \beta_0) E \exp\{j\Delta\psi\}, \quad (6)$$

где E — усредненный квадрат модуля полезной составляющей входного временного процесса, а $\Delta\psi$ — разность начальных фаз полезного входного и опорного сигналов, являющаяся, вообще говоря, случайной величиной.

Раскрывая квадрат модуля сигнальной составляющей корреляционного интеграла через соответствующие суммы частных корреляционных интегралов, запишем

$$|Z_S(\Delta\alpha, \Delta\beta)|^2 = \sum_{n=0}^2 \sum_{m=0}^2 Z_n(\Delta\alpha, \Delta\beta) Z_m^*(\Delta\alpha, \Delta\beta) = E^2 \sum_{n=0}^2 \sum_{m=0}^2 w_{nm}(\Delta\alpha, \Delta\beta), \quad (7)$$

где

$$w_{nm}(\Delta\alpha, \Delta\beta) = s_n(\alpha, \beta) s_n^*(\alpha_0, \beta_0) s_m^*(\alpha, \beta) s_m(\alpha_0, \beta_0) \quad (8)$$

— информативная неэнергетическая величина, определяемая угловыми рассогласованиями локационного объекта относительно равносигнального направления.

Полученное выражение (7) показывает, что величина $|Z_S(\Delta\alpha, \Delta\beta)|^2$, являющаяся аналогом функции правдоподобия, не зависит на разности фаз $\Delta\psi$, но содержит энергетический множитель E^2 , который в случае модели входного сигнала со случайной амплитудой [5] следует компенсировать.

Проводя анализ информативной величины $w_{nm}(\Delta\alpha, \Delta\beta)$ (8) на основе выражения (4), описывающего ее множители, нетрудно заметить, что при $m = n$ значение $w_{nm}(\Delta\alpha, \Delta\beta) = 1$, а при любых n и m выполняется равенство $w_{nm}(\Delta\alpha, \Delta\beta) = w_{mn}^*(\Delta\alpha, \Delta\beta)$.

Эти свойства величины $w_{nm}(\Delta\alpha, \Delta\beta)$ позволяют установить связь между индексами n и m :

$$m = (n + 1) \bmod 3, \quad (9)$$

а также упростить выражение (7), сводя его к виду

$$|Z_S(\Delta\alpha, \Delta\beta)|^2 = 3E^2 + 2E^2 \sum_{n=0}^2 \operatorname{Re}\{w_n(\Delta\alpha, \Delta\beta)\}, \quad (10)$$

где $w_n(\Delta\alpha, \Delta\beta) = w_{nm}(\Delta\alpha, \Delta\beta)$ в силу зависимости (9) индекса m от n .

В области малых рассогласований, таких, что справедливы приближения $\Delta\alpha = \sin\alpha - \sin\alpha_0$, $\Delta\beta = \sin\beta - \sin\beta_0$, представим величину $w_n(\Delta\alpha, \Delta\beta)$ следующим образом:

$$w_n(\Delta\alpha, \Delta\beta) = \exp\{j(a_n\Delta\alpha + b_n\Delta\beta)\}, \quad (11)$$

где a_n и b_n — весовые коэффициенты, определяющие вклад соответственно азимутального и угломестного рассогласований в квадрат модуля n -го частного корреляционного интеграла.

Для нахождения коэффициентов a_n и b_n воспользуемся выражениями (1), (2) и (8) и, проведя преобразования, запишем:

$$a_n = \frac{2\pi}{\lambda} (r_n \cos\varphi_n - r_m \cos\varphi_m) = \sqrt{3}\pi \frac{d_X}{\lambda} \sin\gamma'_n; \quad (12)$$

$$b_n = \frac{2\pi}{\lambda} (r_n \sin\varphi_n - r_m \sin\varphi_m) = \sqrt{3}\pi \frac{d_Y}{\lambda} \cos\gamma'_n, \quad (13)$$

где $\gamma'_n = \gamma + \pi(n + m)/3$.

Выражения (12), (13) содержат множители, определяющие пеленгационную чувствительность синтезируемого дискриминатора в азимутальной и угломестной плоскостях через основные метрические показатели пеленгационной системы — соответственно d_X/λ и d_Y/λ . Множители $\sin\gamma'_n$ и $\cos\gamma'_n$ характеризуют уменьшение пеленгационной чувствительности за счет приведения n -го канала выделения углового рассогласования к плоскостям азимута и угла места.

Для определения алгоритма формирования выходного сигнала дискриминатора пеленгационной системы на основе модели полезного сигнала со случайными амплитудой и фазой [5, 6] найдем первую и вторую производные квадрата модуля сигнальной составляющей корреляционного интеграла — аналога функции правдоподобия.

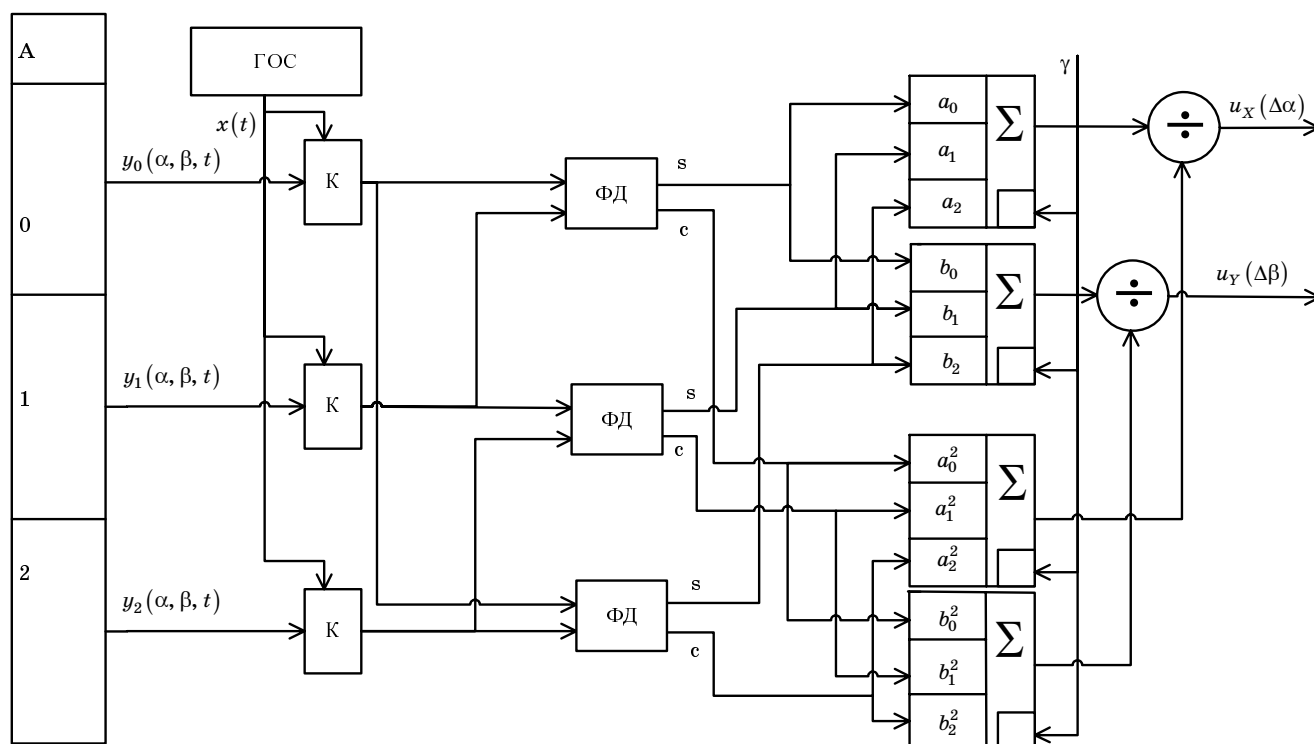
Анализируя (7) и (10), видим, что дифференцирование по одному из рассогласований $\Delta\alpha$ или $\Delta\beta$ в окрестности нулевых их значений ($\Delta\alpha \approx 0$ и $\Delta\beta \approx 0$) эквивалентно умножению произведения частных корреляционных интегралов вида $Z_n(\Delta\alpha, \Delta\beta) Z_m^*(\Delta\alpha, \Delta\beta)$ на соответствующие коэффициенты: a_n и b_n — для первой производной и a_n^2 и b_n^2 — для второй производной, а также общий коэффициент — 2. При этом в окрестности нулевых рассогласований отличными от нуля будут мнимая составляющая произведения $Z_n(\Delta\alpha, \Delta\beta) Z_m^*(\Delta\alpha, \Delta\beta)$ для первой производной и действительная составляющая этого произведения — для второй производной.

Исходя из проведенных рассуждений определим два выходных сигнала, формируемых оптимальным дискриминатором пеленгационной системы с эллипсообразным расположением точек приема соответственно для азимутальной и угломестной плоскостей:

$$u_X(\Delta\alpha) = \frac{\sum_{n=0}^2 a_n \operatorname{Im}\{Z_n(\Delta\alpha, \Delta\beta) Z_m^*(\Delta\alpha, \Delta\beta)\}}{\sum_{n=0}^2 a_n^2 \operatorname{Re}\{Z_n(\Delta\alpha, \Delta\beta) Z_m^*(\Delta\alpha, \Delta\beta)\}}; \quad (14)$$

$$u_Y(\Delta\beta) = \frac{\sum_{n=0}^2 b_n \operatorname{Im}\{Z_n(\Delta\alpha, \Delta\beta) Z_m^*(\Delta\alpha, \Delta\beta)\}}{\sum_{n=0}^2 b_n^2 \operatorname{Re}\{Z_n(\Delta\alpha, \Delta\beta) Z_m^*(\Delta\alpha, \Delta\beta)\}}, \quad (15)$$

где индексы m от n связаны между собой зависимостью (9).



■ Рис. 3. Структурная схема синтезированного дискриминатора

Структурная схема синтезированного дискриминатора, совместно реализующего алгоритмы (14) и (15), приведена на рис. 3. Входным элементом дискриминатора является антенна А с эллипсообразным расположением трех точек приема, а согласованная обработка во временной области осуществляется в трех каналах приема, представленных корреляторами К и общим генератором опорного сигнала ГОС. Мнимые и действительные составляющие произведений частных корреляционных интегралов формируются соответственно на синусных (s) и косинусных (c) выходах трех фазовых детекторов ФД, а затем суммируются в четырех весовых сумматорах Σ. Образующиеся при этом разностные и нормирующие составляющие двух сигналов рассогласования поступают на входы двух блоков деления (÷), в результате чего выполняется нормирование сигналов рассогласования $u_X(\Delta\alpha)$ и $u_Y(\Delta\beta)$.

Выводы

1. Двумерная модель условий синтеза дискриминатора пеленгационной системы позволяет получить ряд новых технических решений за счет совместной пространственной обработки в плоскостях азимута и угла места, что невозможно при одномерном подходе, основанном на факторизации функции раскрытия антенны. Методика подобного синтеза, а также его результаты могут быть распространены на случаи антенных систем сложной конфигурации, например конформных.

2. Синтезированный дискриминатор пеленгационной системы обладает инвариантностью относительно поворота совокупности трех точек приема в плоскости пеленгования на произвольный угол, а потенциальная точность формируемых им оценок угловых рассогласований определяется размерами осей эллипса, на окате которого располагаются данные точки приема.

Литература

1. Анцев Г. В., Павлов В. С., Турецкий Л. С. Возможности статистических методов синтеза пространственно-временных систем // Кибернетика и высокие технологии XXI века: Тр. VII Междунар. конф. Воронеж, 2006.
2. Зайченко К. В., Павлов В. С. Пространственный синтез многоотсчетного фазового чувствительного элемента угловых координат локационного объекта // Изв. вузов. Сер. Приборостроение. 2001. Т. 67. № 6.
3. Павлов В. С. Точностные характеристики многоотсчетных чувствительных элементов локационных

- систем измерения угловых координат // Изв. вузов. Сер. Приборостроение. 2003. Т. 46. № 1.
4. Павлов В. С., Савинов М. В. Условия оптимального измерения двух неэнергетических параметров с различной точностью // Информационно-управляющие системы. 2006. № 4 (23). С. 29–32.
5. Ширман Я. Д., Манжос В. Н. Теория и техника обработки радиолокационной информации на фоне помех. М.: Радио и связь, 1981. 415 с.
6. Теоретические основы радиолокации / А. А. Коростелев, Н. Ф. Клюев, Ю. А. Мельник и др.; Под ред. В. Е. Дулевича. М.: Сов. радио, 1978. 607 с.

УДК 681.3.06

ПРЕДМЕТНО-ОРИЕНТИРОВАННЫЙ ЯЗЫК АВТОМАТНОГО ПРОГРАММИРОВАНИЯ НА БАЗЕ ДИНАМИЧЕСКОГО ЯЗЫКА RUBY

О. Г. Степанов,

программист

Компания JetBrains s. r. o.

А. А. Шалыто,

доктор техн. наук, профессор

Д. Г. Шопырин,

канд. техн. наук, доцент

Санкт-Петербургский государственный университет информационных технологий,
механики и оптики

Решается задача преобразования диаграмм переходов, применяемых в SWITCH-технологии, в исполняемый код. Предлагается использование динамических языков программирования, позволяющих добиться изоморфности диаграммы и соответствующего программного кода, что ведет к уменьшению количества ошибок при указанном преобразовании. На базе динамического языка Ruby с использованием разработанной библиотеки STROBE создан предметно-ориентированный язык автоматного программирования.

This paper solves the problem of transforming transition diagrams used in SWITCH technology into executable code. We suggest using dynamic programming languages to solve this problem because these languages allow to achieve an isomorphism between the source diagram and the resulting code. This leads to decreasing the number of errors that appear during such transition. We also present an extension library STROBE and a domain-specific automata programming language based on the dynamic programming language Ruby.

Введение

Для проектирования и разработки реактивных систем часто используется SWITCH-технология, поддерживающая «автоматное программирование» или «программирование с явным выделением состояний» [1]. Одной из основных частей этой технологии является графический язык, позволяющий описать поведение различных подсистем в терминах состояний и переходов между ними и связей между этими системами в виде диаграмм переходов и связей соответственно.

При использовании SWITCH-технологии в разработке программного обеспечения важной частью является реализация поведения, описанного диаграммами переходов, на целевом языке программирования. Особенностью диаграмм, используемых в указанной технологии, являются сложные условия, образуемые символами входных воздействий (событий и входных переменных), и наличие символов выходных воздействий в вершинах

и/или на переходах. Автоматы, соответствующие таким диаграммам переходов, в теории автоматов называются структурными. Еще одна особенность применяемых диаграмм переходов состоит в том, что в их вершинах могут быть указаны символы вложенных в них автоматов.

Для решения этой задачи традиционно используется один из трех подходов.

1. Полностью ручное программирование. Одним из простейших общепринятых методов такого программирования является следующий: текущее состояние системы хранится в переменной интегрального или перечислимого типа и основная логика программы сосредоточена внутри одного или нескольких операторов `switch`, определяющих действия программы в зависимости от текущего состояния [2]. Другим методом является использование паттерна программирования `State` [3]. Несмотря на такие достоинства этой группы методов как высокая производительность и *полный*

контроль над получаемым кодом, она обладает существенными недостатками: низкой читаемостью кода и большой трудоемкостью.

2. Автоматическая генерация кода по диаграмме переходов. Обычно при таком подходе генерируется код, аналогичный с получаемым при использовании ручного программирования. Недостатками этого подхода являются:

- низкая читаемость кода, связанная с тем, что в качестве целевого используется императивный язык, например Java;
- потеря информации, специфичной для логики диаграмм переходов (вершины диаграммы и ее переходы заменяются их образами на целевом языке — классами и кодом выполнения переходов);
- малая степень контроля над получаемым кодом и невозможность ручного изменения этого кода;
- привязанность к конкретному формату входных данных, с использованием которого задается исходная диаграмма переходов (например, формат файлов Visio [4], XML [5, 6]).

3. Ручное написание кода с применением специальной библиотеки. В этом случае происходит перенос диаграммы переходов в вызовы указанной библиотеки, которая по этим инструкциям строит внутреннее представление рассматриваемой диаграммы. Затем по этому представлению происходит реализация автомата. Основным преимуществом этого подхода является то, что вызовы библиотеки отражают семантику диаграммы переходов (каждый вызов может, например, соответствовать объявлению состояния или перехода). Это позволяет создавать читаемый код, который легко поддерживать. Также некоторые библиотеки, указанные ниже, ориентированы на конкретные виды взаимодействия автоматного и объектно-ориентированного кода, что позволяет более эффективно объединять эти подходы к программированию. Основными недостатками такого подхода являются:

- низкая производительность некоторых библиотек (существуют, однако, реализации, основанные на метапрограммировании и статической генерации кода [7, 8], которые позволяют повысить производительность);
- невозможность описать ряд конструкций диаграмм переходов, используя ограниченный синтаксис целевого языка.

В настоящей работе предлагается развитие третьего подхода за счет повышения «качества» и читаемости получаемого кода. Разработанный подход использует динамические языки программирования — языки, которые позволяют изменять и дополнять код программы во время выполнения. В работе показано, как свойства динамических языков могут быть использованы для увеличения читаемости кода, генерирующего модель системы, особенно в части формирования условий переходов.

Поясним изложенное. В рамках настоящей работы предлагается:

- разработать *текстовый язык автоматного программирования*, который реализован на языке Ruby. Этот язык относится к классу предметно-ориентированных языков программирования (Domain Specific Language — DSL [9]);
- по заданной диаграмме переходов, описывающей поведение автомата, вручную строится его описание на разработанном текстовом языке;
- это текстовое описание транслируется. В результате трансляции с помощью разработанной в настоящей работе библиотеки STROBE (поддерживает такие понятия, как «состояние», «переход», «событие», «входное воздействие», «выходное воздействие» и т. д.) в памяти строится модель диаграммы переходов;
- построенная модель является исполняемой и используется для обработки входных воздействий.

Для формализации процесса переноса диаграмм переходов в код на предлагаемом текстовом языке была разработана операционная семантика (правила интерпретации) диаграмм переходов структурных автоматов. Предложенная семантика обеспечивает корректность реализации диаграмм переходов.

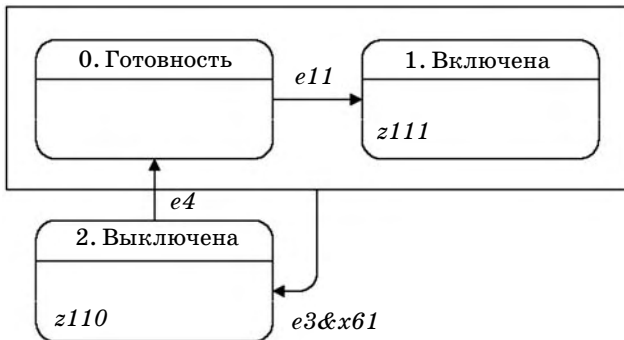
Статья имеет следующую структуру: в первом разделе предложена операционная семантика диаграмм переходов и описана проблема их переноса в исполняемый код. Во втором разделе предложен подход, позволяющий использовать свойства динамических языков для решения задачи переноса диаграмм переходов. В третьем разделе описана библиотека STROBE, реализующая разработанный подход.

Операционная семантика диаграмм переходов

При реализации поведения, описанного с помощью диаграмм переходов на языках программирования, возникают две проблемы, первая из которых состоит в корректном переносе поведения системы в исполняемый код, а вторая — в сохранении исходных обозначений и структуры описания автомата.

Рассмотрим пример использования диаграмм переходов. В работе [10] описана система управления лифтом, которая содержит автомат управления кнопкой вызова лифта на нижнем этаже (A11). Его диаграмма переходов представлена на рисунке.

Этот автомат имеет три состояния: «Готовность» (кнопка на этаже подсвечена и ее можно нажать), «Включена» (кнопка нажата) и «Выключена» (кнопку нажать нельзя). Переход в состояние «Включена» происходит по нажатию кнопки (событие *e11*), переход в состояние «Готовность» — по событию от головного автомата *e4* (разрешение на включение лампы в кнопке),



■ Диаграмма переходов автомата управления кнопкой вызова лифта

переход в состояние «Выключена» — по событию от головного автомата $e3$ (выключение лампы в кнопке) при условии, что лифт находится на нижнем этаже (определяется переменной $x61$).

Для корректного преобразования диаграмм переходов в исполняемый код предложим операционную семантику и метод для выполнения указанного преобразования. Эта семантика в некотором смысле аналогична семантике UML-диаграмм состояний, описанной в работах [11, 12].

Определим основные свойства диаграмм переходов.

1. Диаграмма изображает одно или несколько состояний системы и переходы между ними.

2. Состояния на диаграмме могут быть объединены в группы, которые могут быть вложены друг в друга. Состояния внутри группы равноправны.

3. Переходы могут начинаться в состоянии или в группе состояний, а заканчиваться только в состоянии (переходы, начинающиеся в группе состояний, называются *групповыми переходами*). Переходы могут начинаться и заканчиваться в одном и том же состоянии. При этом они называются *петлями*.

4. Каждое состояние помечено следующими атрибутами:

- имя состояния;
- номер состояния (нумерация начинается с нуля);
- действия при входе в состояние;
- вложенные автоматы (возможно с номерами воздействий, с которыми они вызываются).

5. Переход может быть помечен следующими атрибутами:

- условие перехода;
- действия на переходе;
- приоритет перехода.

Обработка события происходит следующим образом: перебираются переходы, выходящие из текущего состояния и содержащие это состояние групп в порядке приоритета (сначала рассматриваются переходы с меньшими номерами). Для каждого перехода вычисляется условие, которое является булевой формулой. Эта формула может использовать следующие символы:

- ei — имеет значение «истина», если i -е событие наступило, и «ложь» — в противном случае (при написании программ событие $e0$ используется для инициализации каждого автомата);

- xi — переменная xi ;

- yi — номер состояния автомата Ai .

Выполняется первый переход, для которого значение условия истинно. Выполнение перехода состоит из следующих шагов:

- выполняются действия на переходе (вызываются выходные воздействия zi в порядке их следования на переходе);

- осуществляется переход в состояние, в котором заканчивается переход;

- если произошла смена состояния, то выполняются действия при входе в состояние;

- вызываются вложенные автоматы: если для автомата указан номер события, он вызывается с этим событием, иначе — с событием $e0$.

Применим описанную семантику к автомату $A11$.

1. Если текущее состояние — «Готовность» и наступило событие $e11$, то выполняется переход в состояние «Включена». При этом производится выходное воздействие $z111$ (включить лампу в кнопке).

2. Если текущее состояние — «Готовность» или «Включена», наступило событие $e3$ и при этом значение переменной $x61$ — «истина», то осуществляется переход в состояние «Выключена». При этом производится выходное воздействие $z110$ (выключить лампу в кнопке).

3. Если текущее состояние — «Выключена» и наступило событие $e4$, то производится переход в состояние «Готовность».

Реализация автоматных систем на языке Ruby

Для решения задачи переноса диаграмм переходов в исполняемый код предлагается использовать динамические языки программирования [13]. Отличительными свойствами этих языков, позволяющими упростить перенос диаграмм переходов, являются:

- динамическая генерация кода;
- использование замыканий (*замыкание* — совокупность процедуры и связанного с ней лексического контекста, http://en.wikipedia.org/wiki/Closure_%28computer_science%29).

Динамическая генерация кода позволяет для каждой диаграммы переходов создавать индивидуальный код выполнения автомата. Это обеспечивает повышение производительности исполняемого кода.

Для переноса условий переходов в код на целевом языке программирования практически без изменений используются замыкания. Это позволяет сохранить естественную запись условий.

Для практической реализации предложенного подхода был выбран язык Ruby [13], разработанный

ный Юкихино Мацумото (Yukihiro Matsumoto). Этому динамическому языку программирования присущ ряд полезных свойств:

- динамичность (возможность изменения и дополнения программы на лету);
- простой синтаксис;
- объектная ориентированность;
- поддержка примесей (<http://ru.wikipedia.org/wiki/Mixin>), позволяющих уточнить поведение создаваемых классов за счет включения модулей;
- поддержка замыканий;
- легкая переносимость (среда исполнения Ruby может быть запущена в большинстве операционных систем);
- возможность интеграции с кодом на других языках программирования.

В настоящей работе разработана библиотека STROBE и построенный на ее основе текстовый язык автоматного программирования. Они позволяют декларативно задавать диаграммы переходов на языке Ruby. Для программиста эта библиотека предоставляет дополнительный набор инструкций (реализованных в виде методов), позволяющих поэлементно определять диаграммы переходов (в число инструкций входят, например, `state` для определения состояния и `transition` для определения перехода). Методы используют технологию именованных параметров (при вызове метода значение каждого параметра связывается с параметром явно по имени). Это позволяет увеличить читаемость кода. Именами параметров, использованных в примере ниже, являются `:to` или `:output_actions`.

Для пояснения структуры разрабатываемого языка приведем код на этом языке, описывающий поведение автомата *A11*:

```
# Подключение библиотеки STROBE
require 'strobe/automaton'
module Elevator
  # Декларация класса автомата
  class A11 < Strobe::Automaton
    # Декларация внутренней переменной x61
    attr_accessor :x61
    # Декларация событий
    inputs :e3, :e4, :e11
    # Декларация выходных воздействий
    outputs :z110, :z111
    # Начало группы состояний
    begin_group
      # Состояние "Готовность"
      state :ready
      # Переход в состояние "Включена"
      transition :to => :on,
                :if => lambda { e11 }
      # Состояние "Включена"
      state :on,
            :output_actions => :z111
      # Групповой переход в состояние "Выключена"
      group_transition :to => :off,
                      :if => lambda { e3 && x61 }
    end_group
    # Состояние "Выключена"
    state :off,
          :output_actions => :z110
```

```
# Переход в состояние "Включена"
transition :to => :ready,
          :if => lambda { e4 }

end
end
```

В этом примере декларирован класс *A11*. Он объявлен автоматным через наследование от библиотечного класса `Strobe::Automaton`. Внутри декларации класса последовательно определена диаграмма состояний автомата *A11*.

Инструкции `inputs`, `outputs`, `state`, `transition` и `group_transition` декларируют события, выходные воздействия, обычный и групповой переход соответственно. Инструкции `begin_group` и `end_group` объединяют все декларированные между ними состояния в группу.

Метод преобразования диаграмм переходов в исполняемый код на языке Ruby

Представленный выше пример был построен с использованием формального метода, позволяющего перенести любую диаграмму переходов в исполняемый код на языке Ruby.

Опишем схему метода.

1. Для каждой диаграммы создается новый класс с именем, совпадающим с названием автомата.

2. Внутри класса, как это было выполнено в примере, с помощью конструкций предлагаемого языка описывается диаграмма переходов. При этом:

- состояниям присваиваются идентификаторы, соответствующие их именам на диаграмме;
- в описаниях переходов ссылки на состояния производятся по именам.

3. Класс автомата связывается с неавтоматной частью программы через подписку на выходные воздействия автомата.

Ключевыми моментами использования языка Ruby являются полностью декларативное описание поведения системы и автоматическое порождение специальных методов, позволяющих переносить диаграмму переходов с минимумом изменений. Покажем это на примере. Допустим, что в диаграмме существует переход, условие которого задается формулой

$$e1 \vee e0 \wedge (x1 \vee (y2 = 3)).$$

В коде программы эта формула будет представлена в виде выражения

$$e1 || e0 \&\& (x1 || (y2 == 3)).$$

При вычислении условия перехода это выражение будет вычислено в контексте автоматного класса, что позволит использовать порожденные по описанию автомата методы `e0`, `e1`, `x1` и `y2`. При этом подвыражение `y2 == 3` принимает значение «истина», если второй автомат находится в

третьем состоянии, и значение «ложь» — в противном случае.

В общем случае по описанию автомата порождаются следующие методы:

- для каждого события — метод с именем события, который возвращает значение «истина», если наступило это событие, и «ложь» — в противном случае;

- для каждой входной переменной — метод с именем переменной, который возвращает текущее ее значение;

- для связей с другим автоматом системы — метод, имя которого состоит из символа y и номера автомата. Этот метод возвращает номер текущего состояния этого автомата.

При попытке практической реализации предложенного подхода возникает ряд проблем:

- возможность использования конструкций описания автомата непосредственно в теле класса наряду со стандартными декларациями языка;

- обеспечение выполнения условий переходов в соответствующем контексте;

- поддержка наличия нескольких экземпляров одного и того же автомата и обеспечение связей между автоматами через переменные y_i ;

- поддержка интеграции с программами на других языках, включая обеспечение возможности управления физическими устройствами;

- обеспечение приемлемой производительности получаемого кода.

Поставленные проблемы решены в библиотеке STROBE, особенности реализации которой описаны ниже.

Реализация библиотеки STROBE

Библиотека STROBE позволяет перенести в код на языке Ruby любую синтаксически верную диаграмму переходов, а также перенести несколько автоматов и связать их. Возможна также интеграция с модулями на других языках программирования, в том числе модулями управления физическими объектами. При этом описание автомата изоморфно диаграмме состояний и понятно без дополнительных инструкций и описаний.

Описание автомата с помощью библиотеки STROBE имеет следующие особенности.

1. Внешняя конфигурация автомата (входные и выходные воздействия, связи с другими автоматами) описывается в начале класса.

2. Состояния описываются последовательно, в порядке их нумерации на диаграмме переходов.

3. Переходы из состояния описываются непосредственно после описания самого состояния.

4. Связь с другими компонентами осуществляется путем подписки на конкретные выходные воздействия.

5. Связь с другими автоматами по переменным состояния (y_i) выполняется путем явного связывания экземпляров автомата и имен переменных в блоке описания конфигурации автомата.

6. Для независимого выполнения нескольких экземпляров одного автомата используется система доменов.

Домены — логические области, к одной из которых может быть приписан экземпляр автомата при его создании. Каждый экземпляр автомата имеет уникальный в пределах домена идентификатор, по умолчанию равный имени класса. Таким образом, ссылаться на другие автоматы можно по имени класса, используя стандартные алгоритмы разрешения ссылок языка.

Производительность решения находится на одном уровне с динамически выполняемыми решениями третьей группы (интерпретирующими внутреннее представление диаграммы переходов). Однако по всем остальным показателям предложенный подход решает проблемы, описанные в конце предыдущего раздела.

Заключение

В настоящей работе поставлена и решена проблема переноса диаграмм переходов автоматов, разработанных по SWITCH-технологии, в исполняемый код. Были рассмотрены основные направления решения этой проблемы и предложен подход, развивающий одно из них. Этот подход заключается в создании текстового предметно-ориентированного языка описания автоматов (более подробно он описан в работе [14]) на базе динамического языка Ruby. Описаны особенности этого подхода: декларативная структура кода и его изоморфность (особенно в области задания условий переходов) исходной диаграмме. Описаны основные проблемы, возникающие при практической реализации рассматриваемого подхода, и предложена конкретная реализация на динамическом языке программирования Ruby, решающая большинство этих проблем.

Литература

1. Шалыто А. А. SWITCH-технология. Алгоритмизация и программирование задач логического управления. СПб.: Наука, 1998. 628 с. <http://is.ifmo.ru/books/switch/1>
2. Шалыто А. А., Туккель Н. И. Реализация автоматов при программировании событийных систем // Про-

граммист. 2002. № 4. С. 74–80. <http://is.ifmo.ru/works/evsys/>

3. Гамма Э., Хелм Р., Джонсон Р., Влиссидес Дж. Приемы объектно-ориентированного проектирования. Паттерны проектирования. СПб.: Питер, 2001. 368 с.

4. **Инструментальное средство** автоматного программирования Visio2SWITCH. <http://is.ifmo.ru/progeny/visio2switch/>
5. **State Chart XML (SCXML): State Machine Notation for Control Abstraction 1.0.** <http://www.w3.org/TR/2005/WD-scxml-20050705/>
6. **Инструментальное средство** автоматного программирования UniMod. <http://unimod.sourceforge.net>
7. **The Boost C++ Metaprogramming Library.** www.mywikinet.com/mpl/paper/mpl_paper.pdf
8. **Шопырин Д. Г., Шалыто А. А.** Объектно-ориентированный подход к автоматному программированию // Информационно-управляющие системы. 2003. № 5. С. 29–39. <http://is.ifmo.ru/works/ooaut/>
9. **Чарнецки К., Айзенкер У.** Порождающее программирование. Методы, инструменты, применение. СПб.: Питер, 2005.
10. **Наумов А. С., Шалыто А. А.** Система управления лифтом. Проектная программная документация / СПбГУ ИТМО. СПб., 2003. <http://is.ifmo.ru/projects/elevator/>
11. **Гуров В. С., Мазин М. А., Шалыто А. А.** Операционная семантика UML-диаграмм состояний в программном пакете Unimod // Материалы научно-метод. конф. «Телематика-2005» / СПбГУ ИТМО. СПб., 2005. <http://tm.ifmo.ru/tm2005/src/224as.pdf>
12. **Маврин П. Ю.** Реализация диаграмм состояний / СПбГУ ИТМО. СПб., 2006. <http://is.ifmo.ru/papers/statec/>
13. **Thomas D., Fowler C., Hunt A.** Programming Ruby. Second Edition. Pragmatic Bookshelf, 2004.
14. **Степанов О. Г.** Автоматное программирование с использованием динамических языков программирования: Магистерская дис. / СПбГУ ИТМО. СПб., 2006. <http://is.ifmo.ru>, раздел «Работы».



Потапов А. С.

Распознавание образов и машинное восприятие: Общий подход на основе принципа минимальной длины описания. – СПб.: Политехника, 2007. – 548 с.: ил.

ISBN 5-7325-0881-3

В книге подробно рассмотрен принцип минимальной длины описания, являющийся следствием теоретико-информационного подхода к построению моделей и выбору гипотез. Этот принцип становится все более популярным при решении сложных задач автоматического анализа данных, традиционно относившихся к области искусственного интеллекта. Рассмотрены задачи распознавания образов, машинного восприятия и грамматического и логического выводов, для которых использование принципа минимальной длины описания уже позволило получить более эффективные решения. На конкретных примерах показана возможность разработки унифицированного подхода к решению указанных задач.

Книга предназначена для широкого круга читателей: студентов, молодых ученых и специалистов, интересующихся компьютерными науками и, в частности, искусственным интеллектом.

По вопросам приобретения книги обращаться по адресу:

191023, г. Санкт-Петербург, ул. Инженерная, д. 6, 3-й этаж, ОАО «Издательство «Политехника»»
 телефон/факс: (812) 312-44-95 (отдел реализации)

УДК 004.434

ЯЗЫК ОПИСАНИЯ ДИАГРАММ

Ф. А. Новиков,

канд. физ.-мат. наук, доцент

Санкт-Петербургский государственный политехнический университет

К. Б. Степанян,

архитектор проектов

ЗАО «Astrosoft Development»

Предлагается язык описания диаграмм, позволяющий формально определить графический синтаксис (нотацию) диаграмм заданного типа и связать нотацию с семантикой, заданной в форме набора классов. Язык может быть применен для автоматической генерации визуализаторов и графических редакторов диаграмм различных типов.

A Diagram Definition Language (DiaDeL) is proposed. The paper contains textual specifications for graphical notation and its binding to semantics. The semantics is defined as a set of classes. DiaDeL could be used for automatic generation of various diagram visualizers and editors.

Введение

В современной компьютерной индустрии для представления информации широко используются графические конструкции (схемы, диаграммы и т. п.). Особенно большое значение в последнее время приобрели различные диаграммы, используемые при моделировании, проектировании и реализации программного обеспечения. Разнообразие применяемых при этом визуальных образов довольно велико. В данной работе рассматриваются вопросы описания плоских (двумерных) графоподобных диаграмм, которые чаще всего применяются при разработке программ. Неформально графоподобная диаграмма — это диаграмма, состоящая из графических конструкций, которые можно разбить на два множества — множество фигур (вершин) и множество линий, соединяющих эти фигуры (ребер). Формальное определение подобных конструкций можно найти в работе [1].

Изначально подобные диаграммы предназначались только для рисования и чтения человеком. С развитием современных технологий они стали использоваться в целях автоматизации производства. Но инструментальное программное обеспечение, применяемое при работе с диаграммами, должно «понимать», что представлено на диаграмме. Для этого требуется строго формализовать нотацию и семантику всех графических конструкций, используемых на диаграммах данного типа. Если это условие соблюдено, диаграмму можно тем или иным способом интерпретировать, вследствие чего

класс диаграмм данного типа становится визуальным (графическим) языком (например, UML, SDL, MSC). Вопрос о способе формализации описания семантики и нотации графического языка (употребляются также термины абстрактный и конкретный синтаксис соответственно) является одним из центральных в этой статье.

Семантика специфицирует смысловую нагрузку, а нотация — графическое представление конструкций визуального языка. Нотация зависит от семантики: правила построения графических конструкций на диаграмме определяются для семантических элементов языка. Для отображения диаграммы, описанной на визуальном языке, необходима как семантика (что отображаем), так и нотация (как отображаем). Но для иного использования диаграммы, например для генерации кода, достаточно иметь только семантическую составляющую.

В наиболее типичном случае использование конкретного типа диаграмм поддерживается специализированным графическим редактором. Последний выполняет две основные функции: поддерживает соответствие между семантикой и нотацией, визуально отображая изменения в семантической части диаграммы, и поддерживает соответствие между нотацией и семантикой, транслируя визуальные манипуляции с диаграммой в изменения ее смысла. Важно подчеркнуть, что в общем случае эти соответствия не являются взаимнооднозначными: заданный смысл диаграммы может быть изображен различными графиче-

скими конструкциями и, наоборот, разные визуальные манипуляции с изображением диаграммы могут иметь один и тот же смысл.

Важной особенностью современной практики применения диаграмм при разработке программного обеспечения является их «непостоянство». Различные стандарты, инструменты крупных производителей и даже отдельные разработчики изображают и понимают немного по-разному диаграммы с одним и тем же названием. Дело в том, что использование диаграмм в программировании еще не устоялось до такой же степени, как применение чертежей в технике. Со временем унификация, наверное, произойдет, но сейчас для эффективного использования диаграмм при разработке необходимо иметь возможность менять как нотацию, так и семантику буквально «на ходу».

Таким образом, в настоящее время актуальными являются задачи создания адекватных средств описания семантики и нотации диаграмм различных типов и разработки гибких инструментальных средств поддержки работы с диаграммами в разнообразных областях применения.

Обзор существующих решений

Вопросу описания диаграмм и генерации графических редакторов посвящено множество статей и разработок. Все они в качестве результата генерируют редактор диаграмм по заданной спецификации. Ниже рассмотрены некоторые разработки, наиболее примечательные с точки зрения авторов.

Среди всех работ можно выделить множество, в котором нотация диаграммы задается в виде начального алфавита символов (или графических примитивов) и порождающей графической грамматики. К указанному классу работ относятся системы *CIDER* [2], *TIGER* [3] и *DiaGen* [4]. Большим плюсом такого подхода является возможность описывать диаграммы, которые не подпадают под определение графоподобных (например, диаграммы Нейси–Шнейдермана). В то же время, используя подобный способ задания нотации диаграммы, затруднительно описывать сложные конструкции (например, составное состояние в диаграмме состояний или класс в диаграмме классов UML). Две первые системы не позволяют описывать подобные конструкции и редактировать их в сгенерированном редакторе. Причиной данного ограничения авторы считают невозможность указать с помощью используемых в системах грамматик отношение включения одной графической конструкции в другую при описании нотации.

Другим недостатком названных систем является использование своих собственных уникальных семантических моделей. Следовательно, при необходимости использовать внешнюю систему с заданной семантической моделью встает вопрос об интеграции сгенерированного редактора и внешней системы. Подобная интеграция подразумевает

ет трансформацию семантической модели, что является трудной и не всегда выполнимой задачей.

Одной из основ системы *CIDER* является метафора «интеллектуальная диаграмма». По мнению авторов системы, редактор должен допускать синтаксически некорректные конструкции для удобства редактирования. Поэтому сгенерированные редакторы оперируют конструкциями не в семантических терминах, а в терминах графических примитивов, что имеет свои недостатки. Например, для удаления сложной графической конструкции необходимо удалить по порядку все ее примитивные составляющие.

Система *TIGER* является новой версией проекта GenGED [5] и, как следствие, основана на тех же идеях. Главным нововведением является то, что грамматика задается в графическом виде. Такой подход позволяет на диаграмме за одну операцию создавать конструкции, состоящие более чем из одного семантического элемента. Другой особенностью является наличие атрибутов у правил грамматики. Один из них является действием, которое описывает правило. Допустимо четыре типа действий: создание, редактирование, движение и удаление.

Данная система реализована как подключаемый модуль системы Eclipse. Генерируемые редакторы диаграмм также являются подключаемыми модулями для Eclipse. Это придает им гибкость с точки зрения интеграции с другими модулями Eclipse.

В данной статье мы остановимся только на одной отличительной особенности системы *DiaGen*, которая связана со спецификацией нотации для диаграммы. Подход для описания диаграммы, используемый в системе *DiaGen* и в предлагаемом нами языке *DiaDeL*, имеет под собой одинаковую основу, а именно: разделение диаграммы на абстрактный (семантическая модель) и конкретный (нотация) синтаксис. Отличие от систем *CIDER*, *TIGER* и *GenGED* заключается в том, что кроме спецификации графических примитивов, в *DiaGen* пользователь также задает графические отношения (например, «внутри» или «касается»), что придает системе большую гибкость и свободу в описании нотации.

Особого внимания заслуживает проект *GMF* [6]. Система *GMF* является подключаемым модулем среды Eclipse и представляет собой мощный инструмент, позволяющий автоматически генерировать редакторы для диаграмм. Подобно подходу, реализованному как в системе *DiaGen*, так и в предлагаемом языке *DiaDeL*, *GMF* разделяет описание семантики и нотации. Спецификация для генерации редактора в среде *GMF* состоит из следующих частей: описание семантической модели; описание нотации; описание инструментов для работы с сущностями диаграммы; связывание трех описаний воедино. Описание семантической модели и сама семантическая модель должны быть ре-

ализованы на основе библиотеки EMF, что позволяет визуализировать уже существующие модели. Это очевидное преимущество одновременно является существенным недостатком. По сути, GMF может быть использована только в рамках системы Eclipse и не может быть рассмотрена как независимая и сколько-нибудь универсальная система отображения диаграмм.

Основные концепции

Данная статья посвящена языку описания диаграмм DiaDeL (Diagram Definition Language), который позволяет специфицировать нотацию графоподобной диаграммы и связать ее с уже существующей семантикой.

Семантическая модель. Язык DiaDeL основан на предположении, что семантика каждой отдельной диаграммы задана в виде набора конкретных программных объектов определенных классов. Набор объектов, соответствующих конкретной диаграмме, называется ее семантической моделью, а набор всех возможных классов объектов, которые могут появляться на диаграммах данного типа, называется метамоделью. Фактически метамодель описывает абстрактный синтаксис визуального языка.

Подобный подход является гибким в смысле разделения методов компьютерной обработки информации и ее графического представления для пользователя. Зачастую семантическая метамодель задана стандартом, а методы обработки семантических моделей уже существуют (реализованы в каком-либо инструменте), в то время как графическое представление либо определено недостаточно формально, либо у пользователей возникает требование динамически менять графическое представление в соответствии со своими предпочтениями. В таких случаях использование существующих решений (таких, например, как Microsoft Visio) для построения и отображения диаграмм подразумевает интеграцию с заданной метамоделью и методами ее обработки, что не всегда возможно и требует значительных трудозатрат почти всегда, когда допустимо. Также затруднительно использовать генераторы редакторов диаграмм, поскольку они самостоятельно генерируют абстрактный синтаксис визуального языка, который может быть не связан с заданной метамоделью, что потребует программирования трансформаций. Предлагаемый язык имеет целью преодоление этих трудностей.

Основным назначением языка DiaDeL является:

- описание нотации (графического синтаксиса) диаграмм и
- описание связи нотации с существующей семантикой.

Нотация диаграммы задается в виде описания графических конструкций и графических отношений между ними. Связывание нотации с семантикой является ключевым моментом в описании ди-

аграммы. Для этого элементам из семантической модели, которые должны быть представлены визуально, сопоставляются графические конструкции.

Семантическая метамодель должна соответствовать ряду требований для возможности ее использования в DiaDeL-описании.

1. Метамодель должна представлять собой набор классов или интерфейсов.

2. Среди классов метамодели должен быть выделен корневой класс, который соответствует диаграмме в целом. В любой конкретной семантической модели диаграммы объект этого класса должен быть единственным.

3. Классы метамодели должны быть связаны отношениями ассоциации таким образом, чтобы от корневого элемента можно было осуществить навигацию к любому другому элементу.

4. Один элемент семантической модели может быть представлен с помощью одной и только одной графической конструкции. Одна и та же графическая конструкция может являться представлением нескольких разных элементов модели.

5. Элементы семантической модели должны предоставлять возможность извлечения информации, необходимой для построения диаграммы.

Указанные ограничения являются довольно сильными, но, к счастью, большая часть семантических метамodelей, построенных для различных типов диаграмм, им удовлетворяет. Например, для известных метамodelей диалектов UML ограничения 1 и 3–5 выполнены. Если ограничение 2 не выполнено, то нетрудно доработать метамодель, добавив соответствующий корневой класс.

Способы использования языка. Возможны несколько вариантов использования языка DiaDeL.

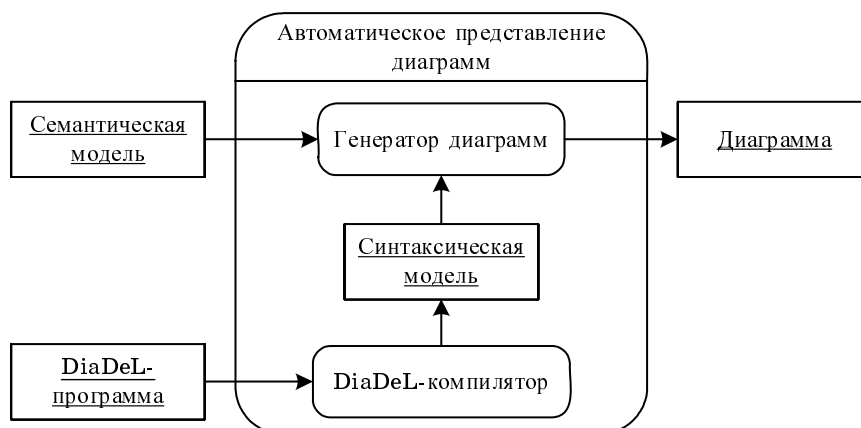
1. Построение системы автоматического представления диаграмм – системы, которая отображает диаграмму, используя описание на языке DiaDeL и экземпляр семантической модели (рис. 1).

2. Построение синтезатора визуализаторов диаграмм – системы, которая генерирует код визуализатора диаграмм с описанной на DiaDeL нотацией и для указанной семантической модели (рис. 2).

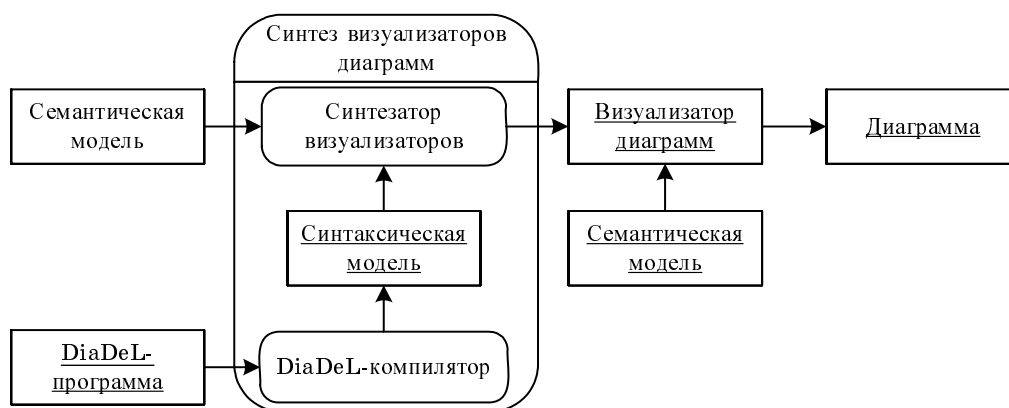
Система автоматического представления диаграмм состоит из двух компонент: компилятора языка DiaDeL и генератора диаграмм. Компилятор производит компиляцию описания диаграммы из конкретного синтаксиса в синтаксическую модель. Генератор диаграмм, используя скомпилированную синтаксическую модель и экземпляр семантической модели, генерирует экземпляр диаграммы.

Синтезатор визуализаторов диаграмм работает по аналогичной схеме, только вместо генератора диаграмм в ней находится синтезатор визуализаторов, который на вход получает саму семантическую модель, а на выходе строит синтезированный код визуализатора.

Приведенные способы использования предлагаемого языка DiaDeL не исчерпывают всех воз-



■ Рис. 1. Схема работы системы автоматического представления диаграмм



■ Рис. 2. Схема работы системы, синтезирующей визуализаторы диаграмм

возможностей, здесь приведены два наиболее важных, с точки зрения авторов. Описание остальных способов использования DiaDeL мы оставляем за рамками статьи.

Особенности языка. Предлагаемый язык разработан с учетом имеющихся решений данной проблемы и нацелен на преодоление определенных недостатков этих решений. Описанная выше архитектура языка базируется на принятии некоторых основных положений, каждое из которых имеет как свои плюсы, так и минусы. Опишем их подробнее.

1. Ограниченный набор графических конструкций. Язык DiaDeL позволяет описывать сравнительно небольшое количество графических конструкций, но уровень этих конструкций достаточно высок. Набор выбранных графических примитивов был определен путем анализа диаграмм тех типов, которые чаще всего используются при разработке программного обеспечения в настоящее время. Например, в число основных конструкций включена «рамка» – конструкция, которую трудно отнести к традиционным графическим примитивам, но которая часто используется в современных визуальных языках. Заметим, что в более ран-

них языках, например в граф-схемах алгоритмов, подобные конструкции не применялись. Достоинством данного решения является удобство описания самых распространенных типов диаграмм, недостатком – отсутствие универсальности.

2. Простой текстовый синтаксис. Для языка DiaDeL определен текстовый синтаксис в традиционном стиле (см. раздел «Описание языка» статьи). Альтернативными способами описания типов диаграмм могли бы быть, например, графические грамматики, визуальные примеры, манипулирование классами модели с помощью различных конструкторов. Выбранное средство (формальный текстовый язык) является наиболее привычным для основной категории пользователей DiaDeL – разработчиков программного обеспечения и в то же время одним из самых простых в реализации. Недостатки текстовых языков общеизвестны, прежде всего, они сравнительно многословны и требуют предварительного изучения перед использованием. Заметим, что в языке DiaDeL используется единый синтаксис как для описания нотации, так и для описания связи нотации с семантикой, что отличает DiaDeL от, например, системы DiaGen.

3. Динамическая связь нотации и семантики. Важным допущением, положенным в основу языка DiaDeL, является предположение о том, что средство описания связи семантики и нотации (семантические мосты, см. раздел «Семантические мосты») имеет «двустороннее действие». Мы предполагаем, что заданная извне семантическая модель обеспечивает не только получение информации, достаточной для отображения диаграммы, но и сама модель позволяет себя менять извне теми же средствами. Это, очевидно, открывает путь к применению «живых» диаграмм, графические манипуляции с которыми отражаются в семантической модели, что является несомненным достоинством. С другой стороны, если используемая семантическая модель не поддерживает «двустороннее» действие, описание диаграммы на языке DiaDeL невозможно «оживить», поскольку DiaDeL не является языком программирования семантических моделей.

Независимость от платформы и языка программирования. Синтаксис языка DiaDeL спроектирован независимо от какого-либо языка программирования и платформы, на которой строится визуализатор диаграмм, описанных на DiaDeL. Это делает возможным построение визуализатора с использованием различных средств и платформ, таких как Java, Eclipse, .NET и т. д. Это выгодно отличает его от большинства существующих решений, которые «привязаны» к конкретной платформе или языку программирования. Например, упомянутые выше решения TIGER и GMF жестко привязаны к Eclipse. Вследствие чего описание диаграммы возможно задать только в среде Eclipse или программно, но используя библиотеки Eclipse. Часть описания диаграммы в системе DiaGen производится на Java, что привязывает систему к целевому языку при генерации редактора.

Описание языка

Язык DiaDeL имеет вполне традиционный синтаксис. Ниже приведен фрагмент формальной порождающей грамматики языка DiaDeL в стандартной форме Бекуса—Наура. Терминальные конструкции набраны полужирным шрифтом, пустая цепочка обозначена ε.

```

Program ::= Diagram DiagramDefs
Diagram ::= diagram id { DiagramDefBody }
DiagramDefs ::= EntityDef DiagramDefs | ε
EntityDef ::= Figure | Line | Text | Decoration
                | Frame | SemBridge
Figure ::= figure id { FigureDefBody }
Line ::= line id { LineDefBody }
Text ::= text id { TextDefBody }
Decoration ::= decoration id { DecorationDefBody }
Frame ::= frame id { FrameDefBody }
SemBridge ::= bridge id { BridgeDefBody }
    
```

Формальное описание дальнейших синтаксических деталей опущено для экономии места. Далее основные конструкции языка рассмотрены на примерах.

Конструкция diagram. В описании нотации конкретного типа диаграмм должна присутствовать одна и только одна конструкция с типом **diagram** (диаграмма). Данная конструкция описывает тип диаграммы в целом и задает для нее допустимые графические конструкции.

Пример описания диаграммы классов UML:

```

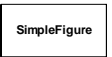
diagram classdiag {
    contain = {class, interface, generalization,
               implementation, association,
               dependency};
}
    
```

Конструкция figure. Фигура — основной элемент для конструирования диаграмм. В графоподобной диаграмме фигура представляет вершину (сущность отображаемой модели), размером и положением которой управляет пользователь. Фигура бывает простой, составной и контейнером. Все они описываются с помощью одного ключевого слова **figure**.

Пример описания простой фигуры class, представленной прямоугольником с текстом внутри фигуры:

```

figure class {
    // геометрический примитив
    shape = rectangle;
    // свойства пера для рисования примитива
    pen.style = solid;
    pen.color = black;
    pen.width = 1;
    // свойства кисти для рисования
    // примитива
    brush.style = solid;
    brush.color = white;
    // расположение текста относительно примитива
    text_position = inside;
}
    
```

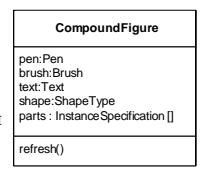


Составная фигура и контейнер являются сложными фигурами. Отличие от простой фигуры заключается в определении отношения включения в сложную фигуру других конструкций. Для составной фигуры отношение включения определяет части фигуры, из которых она состоит (свойство **parts**). Экземпляры этих конструкций создаются вместе с созданием составной фигуры.

Пример составной фигуры **compound_class**, которая включает в себя три части (ns, fs, os), расположенные вертикально друг под другом так, чтобы заполнить всю область родительской фигуры (этот пример соответствует нотации классификаторов на диаграммах классов UML):

```

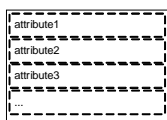
figure compound_class {
    ...
    // части составной фигуры
    parts = {ns : name_sec, // имя
            fs : attributes_sec, // атрибуты
            os : operations_sec}; // операции
    // расположение частей
    layout = vertical;
}
    
```



Конструкция **diagram** должна содержать декларацию для каждой части: `name_sec`, `attributes_sec`, `operations_sec`, т. е. типы частей должны быть объявлены.

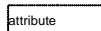
Секция атрибутов `attributes_sec`, в свою очередь, является контейнером. Для контейнера отношение включения определяет конструкции, которые могут быть отображены внутри контейнера (свойство **contain**). Список экземпляров этих конструкций формируется динамически во время построения и обновления диаграммы. Таким образом, два различных экземпляра одной фигуры-контейнера могут включать различные наборы экземпляров графических конструкций:

```
figure attributes_sec {
    shape = rectangle;
    pen.style = solid;
    pen.color = black;
    pen.width = 1;
    contain = {attribute};
    layout = vertical;
}
```



Конструкция text. В рассматриваемом примере секция атрибутов является контейнером для атрибутов, которые представляются в виде текста. Конструкция для представления текста объявляется с помощью ключевого слова **text**. Она позволяет определить шрифт (свойство **font**), перенос слов (свойство **word_wrap**) и выравнивание для отображаемого текста (свойства **hor_align** и **vert_align**):

```
text attribute {
    // шрифт и его свойства отображения
    font.name = "Arial";
    font.color = black;
    font.size = 10;
    // слова не переносятся
    word_wrap = false;
    // выравнивание слева по горизонтали
    hor_align = left;
    // по центру по вертикали
    vert_align = center;
}
```



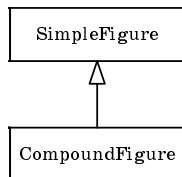
Конструкция line. Линии на диаграмме представляют семантические отношения между сущностями модели. Поэтому у непротиворечивой диаграммы все концы линий инцидентны двумерным графическим конструкциям.

Линия описывается с помощью ключевого слова **line**. Описание линии содержит декларацию пар конструкций, которые линия соединяет на диаграмме (свойство **links**). Для каждой пары указывается допустимая кратность этой линии.

Пример линии **generalization** (отношение обобщения):

```
line generalization {
    links = {(class, class, 1),
            (interface, interface, 1)};

    attachments[100%] =
        (:gen_dec)[100%,50%];
}
```



Представленная линия может соединять два экземпляра конструкции **class** или **interface**. Меж-

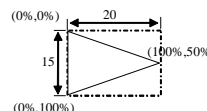
ду ними может быть только один экземпляр линии **generalization**. Треугольник на конце является декорацией, прикрепленной к линии (свойство **attachments**, подробнее см. ниже).

Конструкция decoration. Для отображения графических элементов с фиксированными размерами используется конструкция декорация. Конструкция декорация очень похожа на простую фигуру, но имеет ряд отличительных черт: размер декорации декларируется в описании и не может быть изменен при отображении диаграммы; декорация не может иметь прикреплений, но ее экземпляры могут быть прикреплены к другим конструкциям (см. ниже описание отношения прикрепления); декорация может быть представлена картинкой.

Основное назначение данной конструкции – украшение других конструкций. Но она также может представлять сущность семантической модели (например, интерфейс на диаграмме классов UML, отображаемый в виде кружка).

Пример описания декорации для линии, представляющей отношение обобщения:

```
decoration gen_dec {
    minsize = (20,15);
    shape = polygon((0%,0%),
                  (0%,100%), (100%,50%));
}
```



Поскольку декорация имеет фиксированный размер в пикселях, он должен быть задан в описании (свойство **minsize**). Данный пример также иллюстрирует применение полигона как графического примитива. В скобках через запятую указываются узловые точки, положение которых задается относительно размеров конструкции. Положение (0%, 0%) – это левый верхний угол конструкции, положение (100%, 100%) – правый нижний.

Отношение прикрепления. Для создания комплексных конструкций используется отношение прикрепления, декларирующее, что к родительской конструкции крепится экземпляр другой конструкции. Термин «крепится» несет следующий смысл: положение экземпляра на диаграмме однозначно определяется с помощью положения родительской конструкции и свойств прикрепления, причем прикрепленный экземпляр рисуется поверх родительской конструкции. В роли родительской конструкции могут выступать фигуры, рамки и линии. К ним можно прикреплять экземпляры декораций и текстовых конструкций.

Декларация прикрепления находится в родительской конструкции и имеет следующий синтаксис:

```
attachments[<МТочка>]=(<имя экз>:
                        <имя констр>)[<Сточка>];
```

где **<имя экз>**:**<имя констр>** – объявление экземпляра конструкции, который прикрепляется (имя экземпляра может быть опущено); **<МТочка>** – точка родительской фигуры, к которой соверша-

ется прикрепление; <СТочка> – точка экземпляра, которой он крепится.

При отображении экземпляр размещается таким образом, чтобы указанные точки совпадали. Способ задания точек такой же, как и в случае описания полигона. Следует также отметить, что указание точек для линий и двумерных конструкций отличается. Поскольку линия – это одномерная конструкция, то для указания ее <МТочки> используется только одна координата, которая указывает положение вдоль линии.

Пример конструкции описания прикрепления для линии *generalization*:

```
line generalization {
  ...
  attachments[100%] =
    (:gen_dec)[100%,50%];
}
```

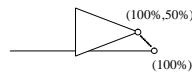


Иллюстрация справа показывает, как должна располагаться декорация относительно линии в приведенном примере. При повороте линии прикрепленные к ней декорации должны поворачиваться соответствующим образом.

Конструкция *frame*. Для обрамления и группирования элементов диаграммы используется графическая конструкция рамка (ключевое слово *frame*). По существу, рамка является фигурой, и к ней применимы все те же правила, но со следующими отличиями: расположение рамок на диаграмме не подчиняется правилам расположения других конструкций; рамка рисуется поверх всех других двумерных конструкций, но не линий (для того чтобы она не перекрывала обрамляемые конструкции при отображении, ее делают прозрачной); отношение включения в рамку других конструкций допустимо только для секций.

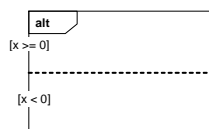
Секция – это специальный тип простой фигуры, который может быть включен в рамку. Рамка может не включать секции (аналогично простой фигуре), включать фиксированное число секций (аналогично составной фигуре, свойство *parts*) и включать изменяемое число секций (аналогично контейнеру, свойство *contain*). Секция для рамки объявляется с использованием ключевого слова *section* и полностью идентична простой фигуре.

Пример описания рамки для представления составных шагов взаимодействия на диаграмме последовательности UML 2.0:

```
frame fragment {
  shape = rectangle;
  pen.style = solid;
  pen.color = black;
  pen.width = 1;
  brush.style = none;

  attachment[0%,0%] =
    (name:alt_dec)[0%,0%];

  layout = vertical;
  contain = {operand};
}
```



```
section operand {
  shape = polyline((0%,0%), (100%,0%));
  pen.style = dash;
  brush.style = none;

  attachment[0%,30%]=
    (cnd:cond_txt)[50%,50%];
}
```

В верхнем левом углу она имеет прикрепленную декорацию для обозначения типа составного шага (на рисунке это рамка с меткой *alt*). Рамка состоит из неопределенного числа секций, разбивающих ее горизонтально. Секции представляют операнды. Каждая секция имеет прикрепленный к ней текст.

Семантические мосты. Для связывания графических конструкций с элементами семантической модели используются семантические мосты. Для одной графической конструкции может быть объявлено несколько семантических мостов, т. е. одна конструкция может представлять на диаграмме разные семантические элементы. Но семантический элемент может быть представлен на диаграмме только одной графической конструкцией.

Декларация семантического моста имеет следующий синтаксис:

```
bridge <имя конструкции> {
  <декларация связей с семантическим элементом>
  <управление отображением>
}
```

Тело декларации моста состоит из двух частей: декларации связей с семантическим элементом и секции управления отображением (см. ниже). Декларация связей является перечислением свойств или методов семантического элемента, которые предоставляют информацию, необходимую для построения экземпляра графической конструкции. Очевидно, что для построения экземпляров разных типов необходима различная по содержанию информация. Например, для отображения экземпляра фигуры контейнера необходимо иметь информацию об элементах, которые включены в нее. А для отображения линии необходимо иметь информацию о ее полюсах.

Дадим описание всех допустимых вариантов декларации связи с указанием, к каким типам конструкций они применимы.

model – объявление элемента (класса или интерфейса) семантической модели, с которой связывается конструкция данным мостом. Эта связь должна быть описана в любом мосту вне зависимости от типа конструкции, для которого он декларируется.

properties – объявление множества свойств элемента, изменение которых влечет обновление диаграммы на экране. Эта связь может быть описана в любом мосту, но не является обязательной.

edges – объявление свойства семантического элемента, используемого для извлечения отношений этого элемента. Данное свойство применимо ко всем двумерным графическим конструкциям,

т. е. к фигуре (любого типа), декорации, рамке и секции.

start, end – объявление свойств семантического отношения, используемых для получения его полюсов. Данные объявления допустимы только в мостах для линий.

children – объявление свойства семантического элемента, используемого для извлечения информации о других элементах, содержащихся в данном. Данная декларация допустима только для диаграммы, контейнера, рамки.

inside – объявление свойства семантического элемента, используемого для извлечения информации о элементах, которые должны быть обрамлены рамкой (или ее секцией). Данное объявление допустимо только для рамок или секций.

Корректное описание типа диаграммы на языке DiaDeL содержит семантические мосты не только для графических конструкций, но и мост для конструкции **diagram**. Он используется для определения элементов, которые должны быть отображены на диаграмме.

Пример семантического моста для диаграммы классов UML:

```
bridge <classdiag> {
    model := org.eclipse.uml2.uml.Package;
    children := model.getPackagedElements();
}
```

Управление отображением. Семантический мост также может содержать блок управления отображением. Этот блок позволяет изменять атрибуты представления графической конструкции в зависимости от состояния семантического элемента. Данная возможность особенно удобна, когда графическая конструкция используется для отображения нескольких семантических элементов.

Блок управления отображением вводится ключевым словом **refresh** и состоит из набора операторов. Допустимы два типа операторов: присваивание и условный оператор. Операторы выполняются каждый раз, когда строится или обновляется представление диаграммы.

Пример семантического моста для составной фигуры **compound_class**:

```
bridge <compound_class> {
    model := org.eclipse.uml2.uml.Class;
    edges := model.getRelations();
    refresh() {
        ns.text.value = model.getName();
        if (model.isAbstract())
            ns.text.font.italic = true;
    }
}
```

Блок управления отображением содержит два оператора. Первый оператор извлекает из модели имя отображаемого элемента (имя класса) и присваивает его значение тексту части, отображающей имя. Второй оператор условный: в случае, если фигура представляет абстрактный класс, то текст в секции имени отображается курсивом.

Следует обратить внимание, что, используя ключевое слово **model**, допустимо обращаться к свойствам и методам семантического элемента (в примере **model.getName()** и **model.isAbstract()**). Обращение к свойствам или частям графической конструкции выполняется без каких-либо префиксных ключевых слов (**ns.text.value** — обращение к значению текста секции имени фигуры **class**).

Особым случаем является объявление семантического моста для конструкции, которая является частью составной фигуры. В этом случае связь с семантическим элементом выводится из связи родительской фигуры, для чего используется ключевое слово **parent**.

Пример семантического моста для секции **attributes_sec** составной фигуры **class**:

```
bridge <attributes_sec> {
    model := parent.model;
    children := model.getOwnedAttributes();
}
```

Пример описания диаграммы классов UML

Приведем пример описания упрощенной нотации для диаграммы классов UML (рассматриваются только классы и обобщения) и ее связь с существующей метамоделью UML, реализованной в виде библиотеки Eclipse.

```
// объявление диаграммы
diagram classdiag {
    contain = {class, generalization}; // содержащиеся конструкции
}
// объявление фигуры класса
figure class {
    shape = rectangle; // графический примитив - прямоугольник
}
// объявление линии обобщения
line generalization {
    links = {(class, class, 1)}; // допустимые полюса и кратность
    attachments[100%] = (:gen_dec)[100%,50%]; // прикрепление декорации
}
// объявление декорации в виде замкнутого треугольника для линии обобщения
decoration gen_dec {
    minsize = (20,15); // размер конструкции
    shape = polygon((0%,0%), // графический примитив - ломаная
        (0%,100%), (100%,50%));
}
// объявление моста для диаграммы
bridge <classdiag> {
    model := org.eclipse.uml2.uml.Package;
    children := model.getPackagedElements(); // элементы диаграммы
}
// объявление моста для фигуры класса
bridge <class> {
    model := org.eclipse.uml2.uml.Class;
    edges := model.getGeneralizations(); // отображаемые отношения
    refresh() {
        text.value = model.getName(); // извлечение имени класса
        if (model.isAbstract()) // если класс абстрактный
    }
}
```

```

        text.font.italic = true; // отображать курсивом
    }
}
// объявление моста для линии обобщения
bridge <generalization> {
    model := org.eclipse.uml2.uml.Generalization;
    start := model.getSpecific(); // начало линии
    end := model.getGeneral(); // конец линии
}

```

Заключение

В статье представлен язык описания диаграмм DiaDeL [7]. Предложенный язык позволяет описать нотацию для диаграммы при условии, что семантика диаграммы (абстрактный синтаксис) задана изначально. Это дает возможность визуализировать уже существующую информацию в виде диаграмм и является выгодным отличием от большинства подобных решений (DiaGen, TIGER, CIDER). На семантику накладывается ряд ограничений, но они более слабые, чем те, которые требует GMF. Более

того, DiaDeL не накладывает ограничений на язык и платформу семантической модели.

Язык DiaDeL включает в себя описание представления в виде графических конструкций и связывание графических элементов с элементами семантической модели. Представлением элемента допустимо управлять динамически. Это придает языку гибкость в смысле описания визуального представления, которое должно динамически меняться в зависимости от состояния семантического элемента. Язык DiaDeL предоставляет возможность описывать сложные конструкции (такие как составные фигуры, контейнеры и рамки), что не предусматривается в системах CIDER, TIGER и GenGED.

Авторы уверены, что использование языка DiaDeL позволит расширить область практической применимости диаграмм различных типов, используемых при разработке программного обеспечения.

Литература

1. Жоголев Е. А. Графические редакторы и графические грамматики // Программирование. 2001. № 3. С. 30–42.
2. Chok S. S., Marriott K. Automatic Generation of Intelligent Diagram Editors // ACM Transactions on Computer-Human Interaction. September 2003. Vol. 10. N 3. P. 244–276.
3. Ehrig K., Ermel C., Hänggen S., Taentzer G. Generation of visual editors as eclipse plug-ins // ACM Intern. Conf. on Automated Software Engineering: Proc. 20th IEEE // IEEE Computer Society. Long Beach, California, USA. 2005.
4. Minas M., Viehstaedt G. DiaGen: A Generator for Diagram Editors Providing Direct Manipulation and Execution of Diagrams: Proc. IEEE Symp. on Visual Languages. September 5–9. Darmstadt, Germany. 1995. P. 203–210.
5. Bardohl R. GenGED – Visual Definition of Visual Languages based on Algebraic Graph Transformation: Дис. ... д-ра. Технический университет Берлина. 1999.
6. Eclipse Consortium. Eclipse Graphical Modeling Framework (GMF) – Version 1.0.2. <http://www.eclipse.org/gmf>. 2006.
7. Степанян К. Б. Язык описания диаграмм // Научно-технические ведомости СПбГПУ. 2006. № 6-1. С. 36–41.

УДК 004.7

МЕТОДИКИ ОЦЕНКИ ПАРАМЕТРОВ РАДИОКАНАЛА

С. Д. Андреев,

аспирант

С. А. Семёнов,

аспирант

А. М. Турликов,

канд. техн. наук, доцент

Санкт-Петербургский государственный университет аэрокосмического приборостроения

Ставится задача измерения пропускной способности и вероятности потери пакета в радиоканале. Уточняется, что в ряде практических приложений необходимо измерять значения пропускной способности с высокой дискретизацией. Предлагаются две методики получения искомых значений, удовлетворяющих выдвинутым требованиям, и обсуждаются границы их применимости.

The paper addresses the problem of the wireless channel throughput and packet error rate estimation. It is shown that some practical applications require the throughput estimation values obtained with high sampling rate. Two methods are introduced to obtain the sought values with high sampling rate and the limits of their applicability are discussed.

Постановка задачи

При передаче видеоданных с высоким разрешением системе, осуществляющей такую передачу, требуется значительный объем ресурсов канала связи. Например, при передаче HDTV-видеоданных [1] по беспроводным широкополосным каналам связи стандарты беспроводных сетей передачи данных накладывают различные ограничения на объем доступных системе передачи ресурсов.

Для выбора оптимальных параметров работы схемы передачи в режиме реального времени необходимо построение той или иной модели канала связи. Для построения такой модели требуется предварительно измерить основные параметры радиоканала (вероятность ошибки и пропускную способность). При этом, в силу существования ограничения на доступные системе передачи ресурсы, пропускную способность необходимо измерять с максимально высоким разрешением по времени, что представляет собой, как будет показано ниже, определенную сложность. Рассмотрим перечисленные параметры канала связи с более формальной точки зрения.

Важнейшей характеристикой любой системы передачи данных, бесспорно, является физическая скорость передачи R (Bit rate, Data rate), равняющаяся числу бит, передаваемых источником в единицу времени. В теореме для канала с шумом К. Шеннон ввел теоретический предел, или пороговое значение, для величины R . В зарубежной ли-

тературе этот порог носит название Capacity (C), а в отечественных работах за ним закрепилось название пропускной способности канала связи. Термин Capacity часто употребляют синонимично с термином Maximum throughput, однако сам термин Throughput означает объем данных, принимаемый получателем в единицу времени. При решении реальных задач исследователя интересует именно реальное значение Maximum throughput или его наиболее точная оценка. К сожалению, в отечественной литературе нет установившегося эквивалента термину Throughput, поэтому далее мы будем не совсем правильно называть пропускной способностью реальное значение Maximum throughput.

С вероятностью ошибки дела обстоят несколько проще. Очевидно, наличие шумов в реальном канале связи ведет к тому, что на стороне получателя значения некоторых бит передаваемого пакета данных изменяются на противоположные. Естественным параметром канала тогда становится вероятность ошибки на бит (Bit Error Ratio — BER). В реальных системах для обнаружения битовых ошибок применяются всевозможные схемы помехоустойчивого кодирования, которые уничтожают принятый пакет, если не удастся исправить в нем все обнаруженные ошибки. Таким образом, даже оценка вероятности ошибки на бит выглядит весьма затруднительной. Однако упрощенно можно пред-
ставлять, что пакеты данных в канале связи «те-

ряются» с некоторой вероятностью. Тогда достаточно легко вычислить оценку вероятности потери пакета (Packet Error Ratio — PER). Подводя итог, отметим, что для решения поставленной задачи необходимо получить оценку пропускной способности канала с высокой дискретизацией, а также оценку вероятности потери пакета в среднем.

При передаче видеоданных по сети, особенно в режиме реального времени, основным требованием является отсутствие задержек при воспроизведении. При этом допускается частичная потеря качества изображения. Поэтому для передачи видеоданных следует по возможности свести к минимуму повторные передачи пакетов на канальном уровне, а лучше совсем от них отказаться.

В этих условиях использование протокола TCP является неэффективным, так как данный протокол изначально проектировался для проводных сетей, в которых ошибки редки, и если и возникают, то чаще всего являются результатом перегрузки. Поэтому когда в канале происходят потери пакетов, протокол TCP уменьшает окно передачи, снижая, таким образом, скорость передачи данных. Протокол UDP допускает потери пакетов на канальном и физическом уровне, поэтому в статье будут рассматриваться методы измерения пропускной способности с использованием именно этого протокола.

Средства измерения пропускной способности

Существующие средства измерения (с открытым исходным кодом) пропускной способности сети, такие как *iperf*, *netperf*, *ttcp* и *treno* [2–5] не подходят для решения поставленной задачи, поскольку ориентированы скорее на определение «проблемных» участков сетей со сложной топологией, чем на измерения в одном канале с двумя абонентами (источником и получателем). Под «проблемными» участками здесь подразумеваются места, в которых производительность сети падает по причине перегрузок отдельных компонентов, например вследствие неправильной настройки коммутаторов или брандмауэров.

Поддержку измерения пропускной способности с использованием протокола UDP имеют практически все рассматриваемые средства измерения. Однако предоставляемое временное разрешение составляет не выше 100 мс и является неприемлемым для задачи построения имитационной модели канала передачи видеоданных. Ниже, после обсуждения особенностей функционирования операционной системы (ОС), предлагаются две методики оценки рассмотренных параметров радиоканала, которые дают более высокое разрешение по времени.

Особенности реализации сокетов в ОС Windows

В целях реализации рассматриваемых методик опишем процесс создания приложения на основе архитектуры «клиент—сервер». При этом клиент (ис-

точник) будет, действуя по какому-либо алгоритму, посылать пакеты, а сервер (получатель) будет получать и обрабатывать пакеты так, чтобы результатом явилась искомая оценка. Программный интерфейс, с помощью которого осуществляется передача пакетов, называется сокетом и определяет пару значений <IP-адрес, порт>. С точки зрения ОС, сокет связан с парой буферов определенной длины, один из которых содержит пакеты, предназначенные для отправки (выходной), а другой — принятые, но еще не обработанные пакеты (входной).

По функционированию сокетов подразделяют на блокирующие и неблокирующие. Блокирующие сокет приостанавливают (блокируют) исполнение программы в двух случаях: если выходной буфер заполнен, а в него делается попытка записать новый пакет, и если входной буфер пуст, а из него делается попытка считать очередной пакет. Неблокирующие сокет, как ясно из названия, не приостанавливают работу программы ни в одном из этих случаев. Тип сокета получателя чаще всего выбирают блокирующим, поскольку пока пакеты из канала не пришли, считывание из входного буфера получателя бессмысленно. Вопрос выбора типа сокета источника представляется более интересным. При использовании блокирующего сокета можно гарантировать, что все пакеты, сгенерированные источником, рано или поздно попадут в канал. Таким образом, поскольку потерь пакетов из-за переполнения буфера источника не возникает, данный тип сокета подходит для измерения оценки вероятности потери пакета, например способом непрерывной передачи пакетов в канал.

Однако существует один тонкий момент, заключающийся в том, что при достаточно высокой заполненности выходного буфера, особенно при его переполнении, возрастает нагрузка на ОС, связанная с обработкой вновь поступивших пакетов. Экспериментально было выяснено, что в ОС Windows при постоянном переполнении блокирующего сокета число посылаемых в единицу времени пакетов уменьшается по сравнению с аналогичной величиной для неблокирующего сокета. Скорее всего, это вызвано тем, что ОС Windows начинает замедлять свою работу при частых попытках передачи управления приложению-клиенту после разблокировки сокета. Таким образом, появляется дополнительная задержка, которая тем больше, чем дольше буфер проводит в заполненном состоянии. Увеличение задержки, в свою очередь, ведет к уменьшению оценочных значений пропускной способности по сравнению с реальными значениями. Следовательно, использование методики непрерывной передачи пакетов в совокупности с блокирующим сокетом не решает проблемы точной оценки пропускной способности канала связи.

Методика непрерывной передачи пакетов

Первой из предлагаемых методик оценки параметров радиоканала является методика непрерыв-

ной передачи пакетов. Отправитель генерирует пакеты с произвольным содержимым и с максимальной возможной интенсивностью передает по каналу. Получатель протоколирует время и размеры принятых пакетов, после чего вычисляется пропускная способность в каждый момент времени способом «скользящего окна».

Способ «скользящего окна» представляет собой сбор статистики в пределах некоторого временного интервала, называемого окном (рис. 1). В этом окне вычисляется пропускная способность, исходя из общего числа принятых бит. Полученное значение присваивается позиции окна. Затем окно сдвигается на следующий отсчет времени, и процедура вычисления повторяется.

Таким образом, точность оценки пропускной способности будет зависеть от размера окна, а разрешение по времени будет определяться величиной сдвига на каждой итерации. При обработке результатов эксперимента исследователь сталкивается с выбором: с одной стороны, при уменьшении размера окна увеличивается разрешающая способность по времени, с другой стороны, число пакетов в окне уменьшается и увеличивается ошибка оценки пропускной способности, которая может быть вычислена по следующей формуле:

$$\Delta B = \frac{\text{PacketLength} \times 8}{\text{WindowSize}}, \text{ где PacketLength — длина}$$

пакета в байтах, а WindowSize — размер окна. Данная формула и была использована для выбора минимально возможного размера окна, исходя из заданных параметров точности измерения.

Очевидно, что для точной оценки пропускной способности при такой методике измерений необходимо максимально «загружать» канал. Поэтому источник должен посылать наибольшее возможное число пакетов в единицу времени. Как было отмечено выше, использование блокирующего сокета в этом случае не позволяет получить точную оценку пропускной способности. Для неблокирующего сокета использование методики непрерывной передачи пакетов будет приводить к переполнению выходного буфера и, как следствие, к неизбежным потерям пакетов на стороне источни-

ка. При таком подходе загруженность канала будет максимальна, однако оценить вероятность потери пакета не представляется возможным.

Интерпретация процесса передачи пакетов в терминах теории массового обслуживания

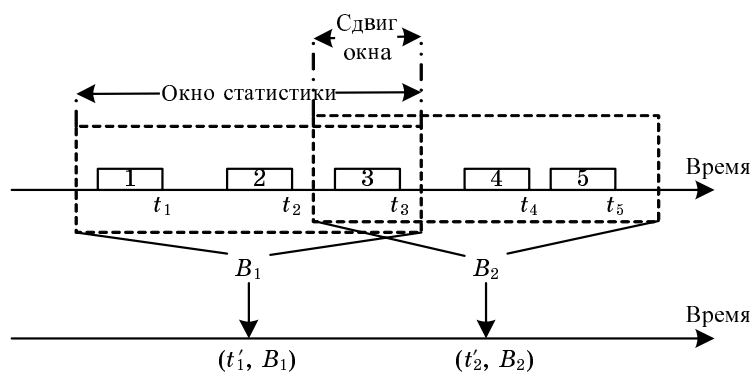
Для того чтобы показать, как можно обеспечить максимальную загрузку канала, опишем рассматриваемую систему передачи в терминах теории массового обслуживания. Тогда ее можно рассматривать как обслуживающий прибор с некоторым временем обслуживания и очередью под пакеты (рис. 2).

Для оценки параметров радиоканала требуется, с одной стороны, поддерживать такую систему массового обслуживания в состоянии насыщения (т. е. в очереди должен быть как минимум один готовый для передачи пакет), а с другой стороны, не допускать потерь пакетов, связанных с переполнением очереди. Предположим, что в простейшем случае время обслуживания рассматриваемого обслуживающего прибора постоянно: $T_0 = x$.

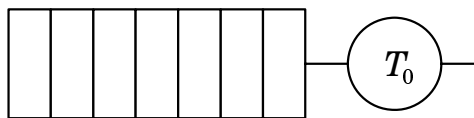
Из теории массового обслуживания известно, что интенсивность входного потока пакетов в систему не должна превышать величины, обратной x , иначе количество пакетов в очереди будет неограниченно возрастать. Известно также, что величина $\frac{1}{x}$ совпадает с максимальной интенсивностью выходного потока, а значит, определяет искомый параметр пропускной способности канала связи.

Очевидно, что в реальных системах рассматриваемое время обслуживания является некоторой случайной величиной $T_0 = X$. Тогда максимальная интенсивность выходного потока является обратной среднему значению X , т. е. математическому ожиданию X ($M[X]$). Таким образом, для получения оценки пропускной способности канала необходимо оценить величину $\frac{1}{M[X]}$.

Чтобы еще более приблизиться к описанию реальной системы, но при этом сохранить относительную простоту модели, следует рассматривать



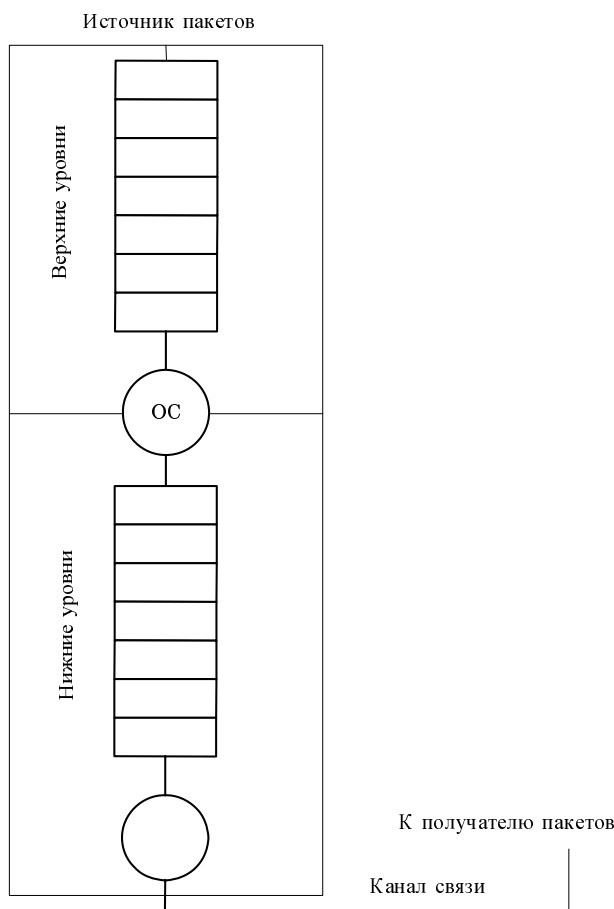
■ Рис. 1. Графическое представление первой методики измерения



■ Рис. 2. Простейшая модель системы передачи

систему передачи данных как две последовательно соединенные системы массового обслуживания, первая из которых соответствует вышележащим (верхним) уровням источника, а вторая включает его нижележащие (нижние) уровни (рис. 3). Очередь первой системы массового обслуживания как раз и представляет собой выходной буфер сокет, а обслуживающим прибором является ОС. Очередь второй системы обусловлена наличием буферизации пакетов на уровнях ниже сетевого [6].

Следует отметить, что функционированием первой системы можно управлять программно, задавая размер очереди (выходного буфера источника) и выбирая блокирующий или неблокирующий тип сокета, в то время как на поведение второй системы напрямую воздействовать не удастся. Таким образом, основная сложность состоит в поддержании второй очереди в состоянии насыщения. Вви-



■ Рис. 3. Усложненная модель системы передачи

ду того, что при непрерывной передаче пакетов вторая очередь не всегда находится в состоянии насыщения, т. е. могут возникать перерывы в передаче пакетов, которые не могут быть учтены при обработке на стороне получателя, при использовании первой методики возникает неточность в оценке пропускной способности канала. Предлагаемый ниже подход призван решить данную проблему.

Описание методики оценки с использованием циклов

Для решения проблемы, связанной с переполнением выходного буфера неблокирующего сокета (первой из двух очередей на рис. 3), предлагается модифицировать алгоритм непрерывной отправки пакетов таким образом, чтобы в передаче пакетов возникали перерывы, которые бы учитывались на стороне получателя. Таким образом, все время эксперимента T по отсылке N пакетов длиной L разбивается на так называемые циклы. В течение каждого цикла источник сначала посылает пакеты с максимально возможной интенсивностью так, чтобы они не переполняли выходной буфер. Затем источник ожидает оставшееся до конца цикла время, а заполненный буфер имеет за это время возможность освободиться (рис. 4). Центральным вопросом при использовании такого подхода является выбор подходящих параметров цикла, а именно, его продолжительности t и числа пакетов n в нем.

Для определения количества пакетов в цикле естественно учитывать размер буфера BS (в байтах) таким образом, чтобы суммарная длина этих пакетов nL не превышала BS . Как уже обсуждалось, имеет смысл выбирать число n так, чтобы пакеты заполняли не весь выходной буфер, а лишь некоторую его часть, определяемую коэффициентом заполнения FC (в процентах). Число пакетов

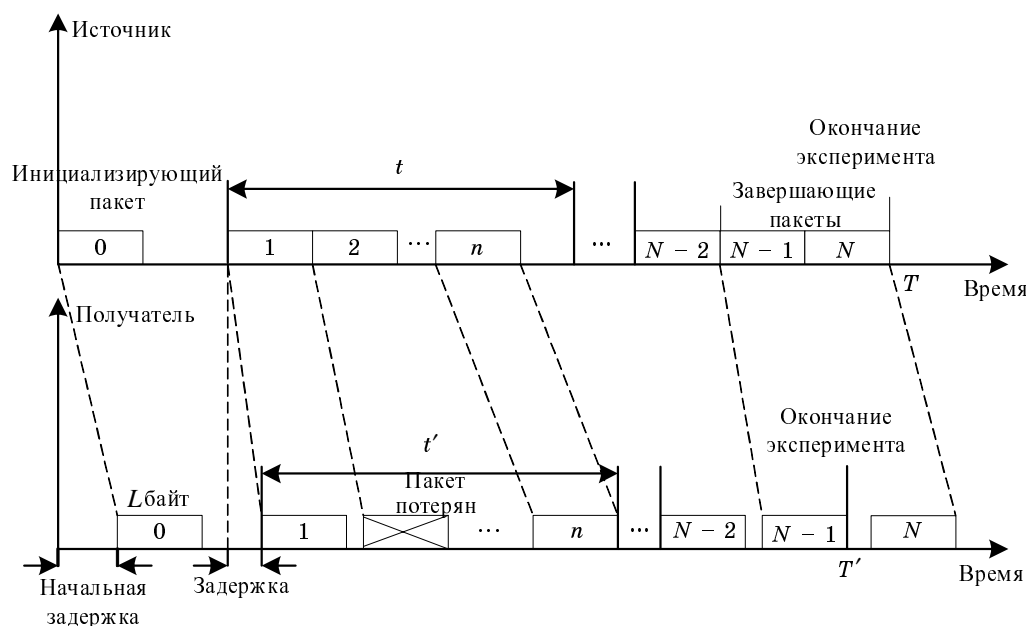
в цикле, не переполняющих буфер: $n = \left\lfloor \frac{BS \times FC}{L \times 100} \right\rfloor$.

Тогда продолжительность цикла t рассчитывается, исходя из длительности эксперимента T и общего числа посылаемых пакетов N : $t = \frac{n \times T}{N}$.

При этом можно также рассчитать количество циклов q в течение одного эксперимента: $q = \left\lceil \frac{T}{t} \right\rceil$.

На стороне получателя вычисляется оценка вероятности потери пакета в среднем как $\hat{P} = 1 - \frac{N'}{N}$,

где N' – общее число успешно принятых пакетов, $N' \leq N$. Заметим также, что, зная число пакетов n в цикле, можно по номеру предыдущего пакета однозначно определить номер того цикла, в котором он был послан. Таким образом, для каждого цикла получатель замеряет время от момента приема первого пакета в цикле до момента приема послед-



■ Рис. 4. Графическое представление второй методики измерений

него пакета в нем t' , $t' \geq t$. Кроме того, для каждого цикла вычисляется число успешно принятых в нем пакетов n' , $n' \leq n$. Тогда оценкой пропускной способности будет являться величина $\hat{\delta} = \frac{n' \times L \times 8}{t' \times 1000^2}$

[Мбит/с].

Проверка методик с использованием симулятора ns-2

Для проверки методик был использован симулятор ns-2 [7]. В качестве модели была выбрана сеть, состоящая из двух узлов и канала без задержки, пропускная способность которого меняется со временем по следующему синусоидальному закону:

$$B(t) = C + A \times \sin\left(\frac{\pi \times t}{\text{period}}\right).$$

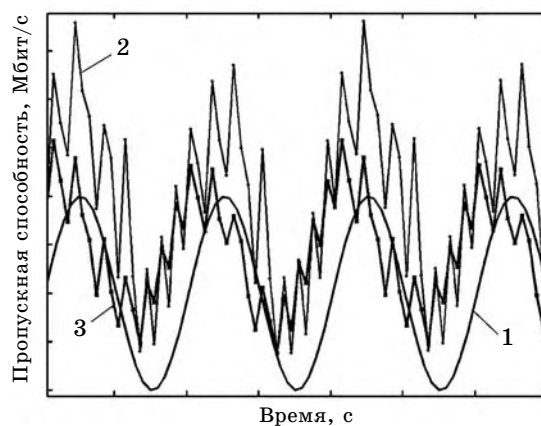
Буфер сокета моделируется очередью пакетов в канал, размер которой задается в пакетах и равен $\frac{\text{SocketBufSize}}{\text{PacketSize}}$, где SocketBufSize — размер входного буфера сокета (в байтах), а PacketSize — размер пакета (в байтах).

Симулятор ns-2 позволяет моделировать работу очереди в режимах, полностью соответствующих работе блокирующего и неблокирующего сокетов. Как показало моделирование, результаты в обоих режимах оказались идентичными, что не отражает реальную ситуацию в ОС Windows. Обе методики, реализованные в симуляторе ns-2 (рис. 5, 6) дают адекватную оценку пропускной способно-

сти канала. При использовании первой методики уменьшение окна статистики ведет к увеличению не только разрешающей способности, но и ошибки измерения.

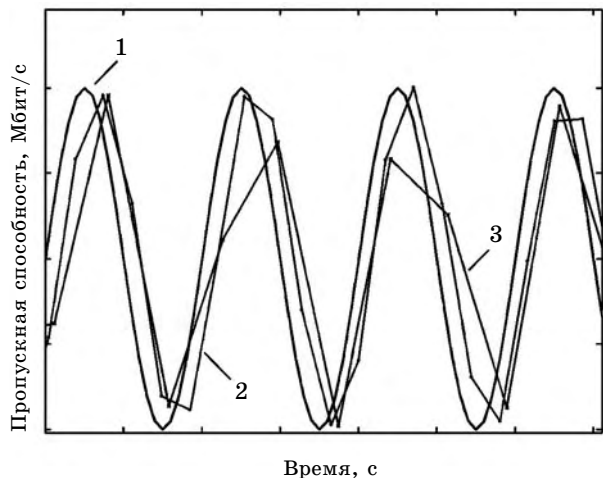
Результаты измерений

Приведем результаты оценки параметров канала связи под управлением двух стандартов беспроводных сетей передачи данных: сравнительно нового стандарта для персональных сверхширокополосных (Ultra-WideBand — UWB) беспроводных сетей [8] (далее — стандарт UWB) и стандарта для



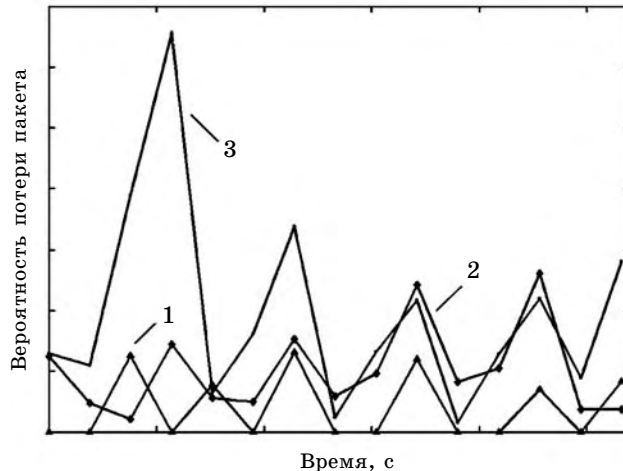
■ Рис. 5. Проверка первой методики измерения:

1 — $50 + 10 \times \sin\left(\frac{\pi \times t}{0,001}\right)$ [Мбит/с]; 2 — окно статистики — 0,1 мс, сдвиг 0,1 мс; 3 — окно статистики — 0,5 мс, сдвиг 0,1 мс



■ Рис. 6. Проверка второй методики измерения:

1 — $50 + 10 \times \sin\left(\frac{\pi \times t}{0,01}\right)$ [Мбит/с]; 2 — буфер 20 000 байт; 3 — буфер 10 000 байт



■ Рис. 7. Зависимость вероятности потери пакета от времени для трафика различных приоритетов в сверхширокополосном радиоканале: 1 — Best Effort; 2 — Video; 3 — Voice

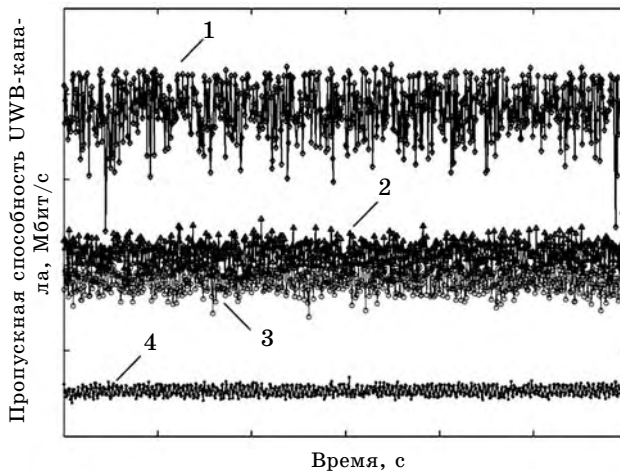
локальных беспроводных сетей передачи данных IEEE 802.11 [9] (далее — стандарт WiFi). В целях потенциальной максимизации пропускной способности (уменьшения количества дополнительных пакетов) для базовой скорости стандарта UWB авторами была проведена соответствующая настройка оборудования. Выбранные для оценки канала параметры даны в табл. 1.

На графиках (рис. 7) показан характер временной зависимости вероятности потери пакета для различных значений приоритета трафика, заявленных в стандарте UWB. Каждая точка графиков получена по итогам отправки примерно 1 млн пакетов. Отметим, что наименьший приоритет трафика (Background) не рассматривался, поскольку для него значение вероятности потери близко к нулю, т. е. потери пакетов практически не происходят.

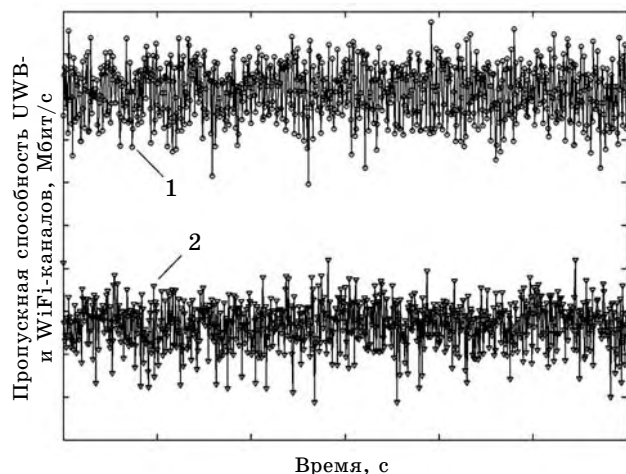
С повышением приоритета вероятность потери пакета в среднем увеличивается. Этот факт можно пояснить следующей гипотезой. Характеристики радиоканала постоянно изменяются, что можно условно представить как переход канала из «хорошего» состояния в «плохое» и обратно. В силу специфики используемого на канальном уровне алгоритма случайного множественного доступа с повышением приоритета абонент ждет меньше время перед тем, как послать пакет в канал. Таким образом, повышение приоритета трафика ведет к тому, что «загружается» отслеживание состояний канала и передаваемый пакет искажается его «плохим» состоянием.

■ Таблица 1

Параметр	Значение
Количество абонентов	2
Политика подтверждения	Без подтверждения (No-ACK)
Механизм RTS/CTS	Отключен
Расстояние от источника до получателя	1 м
Особенности беспроводной среды	Линия видимости (Line Of Sight—LOS)
Скорость передачи данных R	53,3 Мбит/с
Механизм доступа к каналу	Состязательный доступ с приоритетом (PCA)



■ Рис. 8. Зависимость пропускной способности канала от времени для трафика различных приоритетов в сверхширокополосном радиоканале: 1 — Voice; 2 — Best Effort; 3 — Background; 4 — Video



■ **Рис. 9.** Зависимость пропускной способности канала от времени: 1 — для сверхширокополосного радиоканала; 2 — для радиоканала WiFi

Далее продемонстрируем зависимость пропускной способности от времени. На рис. 8 дано сравнение четырех приоритетов стандарта UWB. Видно, например, что значения для трафика с приоритетом Video расположены выше остальных, что поясняется заданным в стандарте UWB максимальным временем «захвата» канала для трафика этого приоритета. Однако для приоритета Video наблюдается также наибольший размах колебаний полученных значений пропускной способности. На рис. 9 сравнивается временная зависимость оценки пропускной способности WiFi-канала с аналогичной оценкой для UWB-канала с наименьшим приоритетом трафика. Видно, что уже на базовой скорости 53,3 Мбит/с пропускная способность канала UWB выше.

Заключение

В данной статье решалась задача получения оценочных значений пропускной способности с

■ **Таблица 2**

Методика оценки	Оценка пропускной способности	Оценка вероятности потери пакета
Непрерывной передачи пакетов	Дает наиболее высокое разрешение по времени	Неприменима, поскольку пакеты могут теряться на стороне источника
С использованием циклов	С момента посылки последнего пакета в цикле и до начала следующего цикла канал не отслеживается	Может использоваться для оценки как вероятности потери, так и вида распределения ошибок

высокой дискретизацией и вероятности потери пакета в среднем применительно к радиоканалу. Показана необходимость в выработке некоторой методики измерений, поскольку доступные на сегодняшний день программные средства оказались неспособны решить поставленную задачу. Был предложен естественный алгоритм непрерывной передачи пакетов. Хотя использующая его методика измерений и позволяет получить наивысшее разрешение по времени, она оказывается неспособной вычислить значение вероятности потери пакета. Исходя из рассмотрения особенностей работы ОС по передаче пакетов был сформулирован альтернативный алгоритм посылки, предполагающий разбиение общего времени передачи на так называемые циклы. Данная методика позволяет решить поставленную задачу, получая оценки обоих параметров радиоканала за один эксперимент. Тем не менее, серьезным недостатком второй методики является наличие интервалов времени, когда параметры канала не отслеживаются. Результат сравнения предложенных методик измерения представлен в табл. 2.

Литература

1. High-definition television. <http://www.dtv.gov/>
2. Средство измерения Iperf. <http://dast.nlanr.net/Projects/Iperf/>
3. Средство измерения Netperf. <http://www.netperf.org/netperf/>
4. Средство измерения Test TCP (TTCP). <http://www.pcausa.com/Utilities/pcattcp.htm>
5. Средство измерения TReno. <http://www.psc.edu/networking/papers/inet96.treno.html>
6. Стандарт ISO/IEC 7498-1 (рекомендации ITU-T (CCITT) серии X.200). Information technology – Open Systems Interconnection – Basic Reference Model: The Basic Model. Январь 1994. [http://standards.iso.org/ittf/PubliclyAvailableStandards/s020269_ISO_IEC_7498-1_1994\(E\).zip](http://standards.iso.org/ittf/PubliclyAvailableStandards/s020269_ISO_IEC_7498-1_1994(E).zip)

7. Программа Network Simulator (ns-2). <http://www.isi.edu/nsnam/ns>
8. Стандарт ECMA-368. High Rate Ultra Wideband PHY and MAC Standard. Декабрь 2005. <http://www.ecma-international.org/publications/files/ECMA-ST/ECMA-368.pdf>
9. Стандарт IEEE 802.11 (ISO/IEC 8802-11). IEEE Standards for Information Technology – Telecommunications and Information Exchange between Systems – Local and Metropolitan Area Network – Specific Requirements – Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. Июнь 1997. <http://standards.ieee.org/getieee802/802.11.html>

УДК 621.39

ПОМЕХОУСТОЙЧИВОСТЬ МОДЕМОВ С ДВУМЕРНЫМИ СИГНАЛЬНЫМИ КОНСТРУКЦИЯМИ ПО ТОЧНЫМ ФОРМУЛАМ ВЕРОЯТНОСТИ ОШИБКИ В КАНАЛЕ БЕЗ ЗАМИРАНИЙ И С ОБЩИМИ ЧЕТЫРЕХПАРАМЕТРИЧЕСКИМИ ЗАМИРАНИЯМИ

Н. В. Савищенко,
доктор техн. наук, профессор
Военная академия связи

Рассматривается помехоустойчивость модемов с двумерными сигнальными конструкциями в канале без замираний и с общими замираниями и белым шумом при когерентном приеме. Введена новая специальная интегральная \mathcal{S} -функция, которая включает в себя \mathcal{H} -функцию, функции Лапласа, Оуэна, Никольсона, (обобщенного) Маркума. Введенная \mathcal{S} -функция позволяет систематизировать вычисление вероятности ошибки в канале с общими четырехпараметрическими замираниями и белым шумом при когерентном приеме.

We discuss the noise immunity of modems with two-dimensional signal constructions in fadingless channels, channels with common fades and with white noise during coherent reception. A new special integral \mathcal{S} -function with includes the \mathcal{H} -function, Laplace's, Owen's, Nicolson's, (generalized) Marcum's functions is introduced. The \mathcal{S} -function allows for a systematic calculation of error probability in the channel with common four-parameter fades and white noise during coherent reception.

В настоящее время в различных системах передачи используются в основном двумерные сигнальные конструкции. Но их многообразие ставит перед исследователями естественный вопрос о том, чем отличаются эти сигнальные конструкции. При равной удельной скорости сравнение осуществляется по энергетическим затратам на передачу одного бита. При этом для многих практически важных каналов связи характерно многолучевое распространение, приводящее к замираниям сигнала, значительно снижающим помехоустойчивость приема. Известные методики расчета вероятности в канале с замираниями основываются на точных формулах в каналах без замираний, которые в научной литературе рассматриваются лишь для относительно простых сигнальных конструкций. В данной статье приведены основные положения по расчету вероятности ошибочного приема в канале с общими замираниями.

Математическая модель канала связи

Предположим, что для передачи цифровой информации используется M сигналов: $s_r(t)$, $r = 0, M - 1$. Сигнал передается на символическом ин-

тервале времени T . Математическая модель канала связи представляется выражением

$$y(t) = \mu(t)s_r(t) + n(t), \quad 0 \leq t \leq T, \quad (1)$$

где $y(t)$ — принятый сигнал; $\mu(t)$ — коэффициент передачи канала; $s_r(t)$ — переданный сигнал и $n(t)$ — белый гауссов шум с односторонней спектральной плотностью шума N_0 . В данной модели можно выделить два наиболее важных случая, рассматриваемых в статье:

- 1) если $\mu(t) = 1$, $0 \leq t \leq T$, то получаем классический канал с аддитивным белым гауссовым шумом;
- 2) если $\mu(t) = \mu$, $0 \leq t \leq T$, где μ — случайная величина, то приходим к каналу с неселективными по частоте общими замираниями. В этом случае предполагается, что замирания настолько медленны, что фазу сигнала можно оценить по принимаемому сигналу без ошибок.

В соответствии с теоремой ортогонализации Грама—Шмидта произвольный сигнал $s_r(t)$, $r = 0, M - 1$ может быть представлен в виде $s_r(t) = \sum_{v=1}^N s_{r,v} \varphi_v(t)$, где $\varphi_v(t)$ — базисные функции.

Это позволяет перейти к геометрической интерпретации сигналов и рассматривать их в конечном евклидовом пространстве, при этом может быть

определена энергия любого сигнала: $E_r = \sum_{v=1}^N s_{rv}^2$.

Так как все энергии конечны, т. е. $E_r < \infty$,

$r = \overline{0, M-1}$, то среди них есть сигналы, имеющие максимальную энергию, которую будем обозначать E_m : $E_m = \max_{r=0, M-1} E_r$. Очевидно, что $E_r \leq E_m$,

$r = \overline{0, M-1}$ и энергия E_m конечна, т. е. $E_m < \infty$.

Средняя энергия сигнала определяется как

$E_c = \sum_{r=0}^{M-1} p(r)E_r$, где $p(r)$ — априорная вероятность передачи r -го сигнала. Обычно величина средней энергии определяется для случая, когда все сигналы равновероятны, т. е. $p(r) = 1/M$:

$E_c = \frac{1}{M} \sum_{r=0}^{M-1} E_r$. Отношение максимальной энергии к средней энергии есть квадрат пикфактора:

$\Pi^2 = E_m/E_c$.

Расстояние между двумя сигналами может быть представлено в виде $d_{rk} = \|s_r - s_k\|$ или

$d_{rk} = \sqrt{\sum_{v=1}^N (s_{rv} - s_{kv})^2}$. Среди всех возможных значений d_{rk} есть минимальная величина $d = \min_{r \neq k} d_{rk}$,

$r, k = \overline{0, M-1}$, отличная от нуля, если $r \neq k$. Традиционно величина d называется минимальным евклидовым расстоянием. В дальнейшем будут использоваться также коэффициенты помехоустойчивости $g_m = \left(\frac{d}{2\sqrt{E_m}}\right)^2$ и $g_c = \left(\frac{d}{2\sqrt{E_c}}\right)^2$.

Отношение максимальной энергии и средней энергии к односторонней спектральной плотности шума определяется соответственно как

$h_m^2 = \frac{E_m}{N_0}$ и $h_c^2 = \frac{E_c}{N_0}$, при этом $h_m^2 = \Pi_c^2 h_c^2$. Величины $E_{bm} = \frac{E_m}{\log_2 M}$ и $E_{bc} = \frac{E_c}{\log_2 M}$ — соответственно максимальное и среднее значения энергии, затраченной для передачи одного бита: $h_{bm}^2 = \frac{E_{bm}}{N_0}$, $h_{bc}^2 = \frac{E_{bc}}{N_0}$, $h_{bm}^2 = \Pi_c^2 h_{bc}^2$ и $h_c^2 = h_{bc}^2 \log_2 M$.

Точные формулы вероятности ошибки в M -ичном символе и на бит сигналов ФМ-М.

Вероятность ошибки в M -ичном символе ($M \geq 2$) двумерного сигнала ФМ-М при когерент-

ном приеме в канале с детерминированными параметрами и белым шумом [1, 2]

$$P_e = Q\left(\sqrt{2h_m^2} \sin \frac{\pi}{M}\right) + 2T\left(\sqrt{2h_m^2} \sin \frac{\pi}{M}, \operatorname{ctg} \frac{\pi}{M}\right), \quad (2)$$

где $h_m^2 = E_m/N_0$, E_m — максимальная энергия сигналов ФМ-М; $Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty \exp\left(-\frac{t^2}{2}\right) dt$ — функция

Лапласа и $T(v, a) = \frac{1}{2\pi} \int_0^a \exp\left[-\frac{v^2}{2}(1+x^2)\right] \frac{dx}{1+x^2}$,

$v \geq 0, a \geq 0$ — функция Оуэна.

Для расчета вероятности ошибки на бит при манипуляционном кодировании кодом Грея для ФМ-М ($M = 2^k$) справедливы следующие формулы [2]:

$$P_{b1} = P_{b2} = \frac{4}{M} \sum_{i=1}^{M/4} Q\left(\sqrt{2h_m^2} \sin \left[\frac{(2i-1)\pi}{M}\right]\right)$$

для первых двух бит и для бит $i \geq 3$

$$P_{bi} = \frac{2^{i+1}}{M} \sum_{j=1}^{M/4} (-1)^{\operatorname{ent}\left[\frac{j-1}{2^{k+1-i}}\right]} \times T\left(\sqrt{2h_m^2} \sin \frac{(2j-1)\pi}{M}, \operatorname{ctg} \frac{(2j-1)\pi}{M}\right),$$

где $M = 2^k$, в частности:

$$P_{b3} = 2P_{b1} - \frac{16}{M} \sum_{j=1}^{M/8} Q\left(\sqrt{2h_m^2} \sin \frac{(2j-1)\pi}{M}\right) \times Q\left(\sqrt{2h_m^2} \cos \frac{(2j-1)\pi}{M}\right).$$

Используя свойства функции Оуэна, нетрудно показать, что справедливы следующие верхние и

нижние границы ($i \geq 3$): $\frac{2^i}{M} Q_1 \leq P_{bi} \leq 2^{i-2} P_{b1}$, где

$Q_m = Q\left(\sqrt{2h_m^2} \sin \left[\frac{m\pi}{M}\right]\right)$. Для средней вероятности ошибки на бит имеем

$$\frac{1}{k} P_e \leq \frac{2}{k} Q_1 \leq P_b = \frac{1}{k} \sum_{i=1}^k P_{bi} \leq \frac{M}{2k} P_{b1}.$$

Приведем примеры расчета вероятности ошибки для сигналов квадратурной амплитудной модуляции, используемых в современных модемах.

1. Сигналы КАМ-16 (факс-протокол V. 29). Согласно рекомендации протокола V. 29 (факс-про-

токол модуляции V. 29 по рекомендации МККТТ), в модемах может использоваться сигнальное созвездие КАМ-16 с четырьмя градациями амплитуды и восемью градациями фазы, для которой

$$E_m = \frac{25}{4}d^2, \quad E_c = \frac{27}{8}d^2 \text{ и } d = \frac{2\sqrt{E_m}}{5} = \frac{2}{3}\sqrt{\frac{2E_c}{3}},$$

$$\Pi_c^2 = \frac{50}{27} = 1, (851), \quad h_{bm}^2 = \Pi^2 h_{bc}^2.$$

Данная сигнальная конструкция КАМ-16 обладает пониженной чувствительностью к дрожанию (флуктуациям) фазы несущей частоты благодаря большому, по сравнению с классическими сигналами КАМ-16, углу между сигнальными точками. Вероятность ошибки на символ в этом случае вычисляется по формуле

$$P_e = \sum_{k=1}^4 a_k T(b_k \sqrt{h_{bc}^2}, c_k) + \sum_{k=1}^4 d_k Q(f_k \sqrt{h_{bc}^2}) + \sum_{k=1}^3 g_k Q(p_k \sqrt{h_{bc}^2}) Q(s_k \sqrt{h_{bc}^2}),$$

где

$$\bar{a} = \begin{pmatrix} \frac{1}{2} & 1 & 1 & 1 \end{pmatrix}, \quad \bar{b} = \begin{pmatrix} \frac{4}{3\sqrt{3}} & \frac{2\sqrt{5}}{3\sqrt{3}} & \frac{4\sqrt{2}}{3\sqrt{3}} & \frac{2}{\sqrt{3}} \end{pmatrix},$$

$$\bar{c} = \begin{pmatrix} \frac{3}{4} & \frac{1}{2} & \frac{1}{2} & \frac{1}{3} \end{pmatrix},$$

$$\bar{d} = \begin{pmatrix} \frac{3}{4} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{pmatrix}, \quad \bar{f} = \begin{pmatrix} \frac{4}{3\sqrt{3}} & \frac{2\sqrt{5}}{3\sqrt{3}} & \frac{\sqrt{5}}{\sqrt{3}} & \frac{2\sqrt{13}}{3\sqrt{3}} \end{pmatrix},$$

$$\bar{g} = \begin{pmatrix} -\frac{1}{4} & -\frac{1}{2} & -1 \end{pmatrix},$$

$$\bar{p} = \begin{pmatrix} \frac{4}{3\sqrt{3}} & \frac{2\sqrt{5}}{3\sqrt{3}} & \frac{2}{\sqrt{3}} \end{pmatrix}, \quad \bar{s} = \begin{pmatrix} \frac{4}{3\sqrt{3}} & \frac{2\sqrt{5}}{3\sqrt{3}} & \frac{4}{3\sqrt{3}} \end{pmatrix}.$$

2. Сигналы КАМ-16 (стандарт MIL-STD-188-110B). На основе результатов, приведенных в работе [2], показано, что вероятность ошибки на символ для КАМ-16, образуемой объединением двух сигналов фазовой модуляции (внутренней ФМ-4 и внешней ФМ-12) и используемой в стандарте MIL-STD-188-110B, в канале с детерминированными параметрами и аддитивным белым шумом при оптимальном когерентном приеме по правилу максимального правдоподобия, равна:

$$P_e = \sum_{k=1}^8 a_k T(b_k \Pi_c \sqrt{h_{bc}^2 \log_2 M}, c_k) + Q(\lambda \Pi_c \sqrt{h_{bc}^2 \log_2 M}) + g Q^2(\lambda \Pi_c \sqrt{h_{bc}^2 \log_2 M}),$$

где

$$\bar{a} = (1 \ 1 \ 1 \ 1 \ 1/2 \ -1/2 \ -1/2 \ 1/2),$$

$$\bar{b} = (\rho \ \rho \ k \ k \ \lambda \ \lambda \ \lambda \ \lambda),$$

$$\bar{c} = (\text{ctg}\xi \ \text{ctg}\varphi \ \text{ctg}\psi \ \text{tg}\left(\xi - \frac{\pi}{3}\right) \rightarrow \rightarrow \text{ctg}\chi \ \text{ctg}(2\varphi) \ \text{ctg}(\xi + \psi) \ \text{ctg}\left(2\xi - \frac{2\pi}{3}\right)),$$

$$g = -\frac{1}{4}.$$

При этом

$$\rho = \sqrt{1 - \frac{\sqrt{6} + \sqrt{3} - \sqrt{2}}{4}} = 0,555129;$$

$$k = \sqrt{1 - \frac{\sqrt{2} + \sqrt{3}}{4}} = 0,461989; \quad \lambda = \frac{\sqrt{3} - 1}{2} = 0,366025;$$

$$\text{tg}\xi = \frac{\sqrt{6} + \sqrt{3} + \sqrt{2} - 3}{\sqrt{6} - \sqrt{3} - \sqrt{2} + 1}; \quad \text{tg}\varphi = \sqrt{6} + \sqrt{2} - 1;$$

$$\text{tg}\psi = \frac{\sqrt{6} + \sqrt{3} + \sqrt{2} - 3}{\sqrt{6} + \sqrt{3} - \sqrt{2} - 1}; \quad \sin\chi = \frac{2 + 3\sqrt{2} - 2\sqrt{6}}{8\rho k}.$$

Максимальная и средняя энергии определяются как

$$E_m = \frac{d^2}{2 - \sqrt{3}}, \quad E_c = \frac{1 - \sqrt{3}/8}{2 - \sqrt{3}} d^2,$$

где d — минимальное евклидово расстояние сигнальной конструкции. Квадрат пикфактора

$$\Pi_c^2 = \frac{8}{8 - \sqrt{3}} = 1,276335. \quad \text{Значения } h_m^2 = \frac{E_m}{N_0},$$

$h_c^2 = \frac{E_c}{N_0}$ есть, соответственно, отношение максимальной и средней энергий к односторонней спектральной плотности шума, $h_m^2 = \Pi_c^2 h_c^2$, $h_c^2 = h_{bc}^2 \log_2 M$, $M = 16$. Коэффициенты помехоустойчивости

$$g_m = \frac{2 - \sqrt{3}}{4} = 0,066987; \quad g_c = \frac{4 - 2\sqrt{3}}{8 - \sqrt{3}} = 0,085498.$$

3. Сигналы КАМ-32 (стандарт цифрового телевидения DVB-T). Точная формула средней вероятности ошибки в M -ичном символе при оптимальном когерентном приеме двумерных сигналов КАМ-32 по правилу максимального правдоподобия в канале с детерминированными параметрами и аддитивным белым шумом при равновероятных битах источника получена в виде [3]

$$P_e = \frac{1}{8}Q\left(\sqrt{h_{bc}^2}\right) + \frac{26}{8}Q\left(\frac{1}{\sqrt{2}}\sqrt{h_{bc}^2}\right) - \frac{23}{8}Q^2\left(\frac{1}{\sqrt{2}}\sqrt{h_{bc}^2}\right).$$

Максимальная и средняя энергии определяют-ся как $E_m = \frac{17}{2}d^2$, $E_c = 5d^2$, коэффициенты помехо-устойчивости $g_m = \frac{1}{34} = 0,029412$, $g_c = \frac{1}{20} = 0,05$.

4. Сигналы КАМ-32 (стандарт MIL-STD-118-110). Точная формула средней вероятности ошибки в M -ичном символе при оптимальном когерентном приеме двумерных сигналов КАМ-32-MIL (стандарт MIL-STD-118-110) по правилу максимального правдоподобия в канале с детерминированными параметрами и аддитивным белым шумом при равновероятных битах источника получена в виде

$$P_e = \sum_{k=1}^{10} \tau_k T\left(\xi_k \Pi_c \sqrt{h_{bc}^2 \log_2 M}, \omega_k\right) + \sum_{k=1}^6 \rho_k Q\left(\eta_k \Pi_c \sqrt{h_{bc}^2 \log_2 M}\right) + \sum_{k=1}^3 q_k Q\left(v_k \Pi_c \sqrt{h_{bc}^2 \log_2 M}\right) Q\left(y_k \Pi_c \sqrt{h_{bc}^2 \log_2 M}\right),$$

где

$$\bar{\tau} = \left(\frac{1}{2} \quad -\frac{1}{4} \quad \frac{1}{8} \quad 1 \quad \frac{1}{8} \quad \frac{3}{4} \quad \frac{3}{8} \quad \frac{3}{8} \quad \frac{1}{4} \quad \frac{1}{4} \right),$$

$$r = \frac{\cos \alpha - 3 \sin \alpha}{2},$$

$$\bar{\xi} = \left(\frac{2r \cos(\psi - 2\alpha)}{\sqrt{2} \sin \psi} \quad \frac{2r \operatorname{ctg} \psi}{\sqrt{2}} \quad \frac{2r}{\sqrt{2}} \quad \sqrt{2} \sin \alpha \quad \sqrt{2} \sin \alpha \rightarrow \right.$$

$$\rightarrow \sqrt{2} \sin \alpha \quad \frac{2r \cos(2\alpha)}{\sqrt{2} \sin \psi} \quad \frac{2r \cos(2\alpha)}{\sqrt{2} \sin \psi} \rightarrow$$

$$\left. \rightarrow \frac{\sin \chi}{\cos \theta} \left(\frac{2r \cos(2\alpha)}{\sqrt{2} \sin \psi} \right) \sqrt{2} \sin \beta \right),$$

$$\bar{\omega} = \left(\operatorname{tg}(\psi - 2\alpha) \quad \operatorname{tg} \psi \quad \operatorname{ctg} \psi \quad \operatorname{tg} \left(\frac{3\pi}{8} - \frac{\gamma}{2} \right) \rightarrow \right.$$

$$\left. \rightarrow \operatorname{tg} \varphi \operatorname{tg} \gamma \quad \operatorname{tg}(2\alpha) \operatorname{tg} \theta \quad \operatorname{ctg} \chi \operatorname{ctg}(2\gamma) \right),$$

$$\bar{\rho} = \left(\frac{5}{16} \quad \frac{19}{16} \quad \frac{1}{8} \quad \frac{1}{4} \quad \frac{1}{8} \quad \frac{1}{8} \right),$$

$$\bar{\eta} = \left(\frac{2r}{\sqrt{2}} \sqrt{2} \sin \alpha \quad \frac{2r \operatorname{ctg} \psi}{\sqrt{2}} \quad \frac{2r \cos(\psi - 2\alpha)}{\sqrt{2} \sin \psi} \rightarrow \right.$$

$$\left. \rightarrow \frac{\sin \chi}{\cos \theta} \left(\frac{2r \cos(2\alpha)}{\sqrt{2} \sin \psi} \right) \sqrt{2} \sin \beta \right),$$

$$\bar{q} = \left(-\frac{3}{8} \quad -\frac{1}{4} \quad -1 \right), \quad \bar{v} = \left(\sqrt{2} \sin \alpha \quad \frac{2r}{\sqrt{2}} \quad \sqrt{2} \sin \alpha \right),$$

$$\bar{y} = \left(\frac{2r}{\sqrt{2}} \quad \frac{2r \operatorname{ctg} \psi}{\sqrt{2}} \quad \sqrt{2} \sin \alpha \right).$$

Соответствующие углы определяются по формулам:

$$3\alpha + \beta = \frac{\pi}{4}, \quad \sin \gamma = \frac{1 \sin \beta}{2 \sin \alpha}, \quad \operatorname{tg} k = \frac{1}{3},$$

$$\operatorname{tg} \psi = \frac{\cos(2\alpha)}{1 - \cos(2\alpha)} \operatorname{tg}(k - \alpha),$$

$$\operatorname{tg} \theta = \frac{3 \sin \alpha - \sin(3\alpha)}{\cos(3\alpha) - 3 \sin \alpha}, \quad \varphi = \psi - 2\alpha - \theta,$$

$$\chi = \frac{\pi}{4} + \psi - 2\alpha + \theta - \gamma.$$

Для определения угла α необходимо найти решение уравнения $f(\alpha) = 0$, где

$$f(\alpha) = 1 - 6 \sin \alpha (\sin(3\alpha) + \cos(3\alpha)) + 14 \sin^2 \alpha,$$

которое после элементарных преобразований можно свести, например, к уравнению

$$21 \operatorname{tg}^4 \alpha + 18 \operatorname{tg}^3 \alpha - 2 \operatorname{tg}^2 \alpha - 6 \operatorname{tg} \alpha + 1 = 0.$$

Численное решение этого уравнения $\sin \alpha = 0,173405$, $|f(\alpha)| \cong 5,5 \cdot 10^{-17}$. В стандарте MIL-STD-118-110 этому углу соответствует значение $\sin \alpha = 0,173415$, $|f(\alpha)| \cong 4,9 \cdot 10^{-5}$. В дальнейшем вычисления вероятности ошибки осуществлялись для значения угла, определенного в стандарте.

Максимальная и средняя энергии равны:

$$E_m = \frac{d^2}{4 \sin^2 \alpha}, \quad E_c = \frac{1}{8} \left(10 + \frac{1}{\sin^2 \alpha} \right) d^2.$$

Коэффициенты помехоустойчивости $g_m = \sin^2 \alpha$, $g_c = 2 \left(10 + \frac{1}{\sin^2 \alpha} \right)^{-1}$ или, соответственно, $g_m = 0,030073$, $g_c = 0,04624$. Таким образом, по средней энергии сигналы КАМ-32 эффективнее КАМ-32-MIL, но по максимальной энергии они про-

игрывают. Квадрат пикфактора $\Pi_c^2 = \frac{2}{1 + 10 \sin^2 \alpha}$ или $\Pi_c^2 = 1,537601$.

5. Сигналы КАМ-64 ($M = 64$). Построены на основе квадратной решетки с внешними точками, сдвинутыми по оси координат. Использовались в модеме Paradyne при скорости передачи 14,4 кбит/с [4, с. 591, рис. 9.18, а]. Вероятность ошибки на символ

$$P_e = \sum_{k=1}^6 a_k T(b_k \sqrt{h_{bc}^2}, c_k) + \sum_{k=1}^3 d_k Q(f_k \sqrt{h_{bc}^2}) + gQ^2(p\sqrt{h_{bc}^2}),$$

где

$$\bar{a} = \begin{pmatrix} \frac{1}{2} & \frac{3}{8} & \frac{1}{4} & \frac{1}{4} & -\frac{1}{4} & -\frac{1}{4} \end{pmatrix},$$

$$\bar{b} = (\vartheta \quad \vartheta \quad \sqrt{3}\vartheta \quad \vartheta \quad \sqrt{3}\vartheta \quad \sqrt{7}\vartheta),$$

$$\bar{c} = \begin{pmatrix} \sqrt{3} & \frac{1}{\sqrt{3}} & \frac{5}{\sqrt{3}} & 3\sqrt{3} & \frac{1}{\sqrt{3}} & \sqrt{3} \end{pmatrix},$$

$$\bar{d} = \begin{pmatrix} \frac{47}{16} & \frac{1}{8} & \frac{1}{16} \end{pmatrix}, \bar{f} = (\vartheta \quad \sqrt{7}\vartheta \quad \sqrt{2}\vartheta), g = -\frac{47}{16},$$

$$p = \vartheta, \vartheta = \frac{4\sqrt{\log_2 M}}{\sqrt{313+7\sqrt{3}}},$$

$$\Pi_c^2 = 16 \frac{26+7\sqrt{3}}{313+7\sqrt{3}} = 1,876173.$$

Рассмотрим теперь гексагональные сигнальные конструкции.

1. Вероятность ошибки для треугольной НЕХ-8 (2, 2, 4) [4, с. 592, рис. 9.17] составляет

$$P_e = \frac{9}{2} T\left(\frac{2}{\sqrt{3}}\sqrt{h_{bc}^2}, \frac{1}{\sqrt{3}}\right) + T\left(\frac{2}{\sqrt{3}}\sqrt{h_{bc}^2}, \sqrt{3}\right) - \frac{1}{2} T\left(2\sqrt{h_{bc}^2}, \frac{1}{\sqrt{3}}\right) + \frac{1}{2} Q\left(\frac{2}{\sqrt{3}}\sqrt{h_{bc}^2}\right) + \frac{1}{4} Q\left(2\sqrt{h_{bc}^2}\right).$$

Квадрат пикфактора $\Pi_c^2 = \frac{14}{9} = 1,5$.

2. Сигналы ГЕКС-16. Для сравнения приведем формулу для вероятности ошибки на символ для ГЕКС-16, $M = 16$ [4, с. 594, рис. 9.20, а]. Полученный сигнальный ансамбль ГЕКС часто изображается в виде трех состыкованных шестиугольников. Для этой сигнальной конструкции максимальная энергия $E_m = 4d^2$, средняя энергия

$$E_c = \frac{9}{4}d^2, \text{ квадрат пикфактора } \Pi_c^2 = \frac{16}{9} = 1,77, \text{ ко-}$$

эффициенты помехоустойчивости $g_m = \frac{1}{16}, g_c = \frac{1}{9}$. Вероятность ошибки на символ может быть определена по формуле [2]

$$P_e = \sum_{k=1}^3 a_k T(b_k \sqrt{h_{bc}^2}, c_k) + \sum_{k=1}^2 d_k Q(f_k \sqrt{h_{bc}^2}),$$

где

$$\bar{a} = \begin{pmatrix} \frac{27}{4} & \frac{3}{4} & -\frac{3}{8} \end{pmatrix}, \bar{b} = \begin{pmatrix} \frac{2\sqrt{2}}{3} & \frac{2\sqrt{2}}{3} & \frac{2\sqrt{2}}{\sqrt{3}} \end{pmatrix},$$

$$\bar{c} = \begin{pmatrix} \frac{1}{\sqrt{3}} & \sqrt{3} & \frac{1}{\sqrt{3}} \end{pmatrix},$$

$$\bar{d} = \begin{pmatrix} \frac{3}{8} & \frac{3}{16} \end{pmatrix}, \bar{f} = \begin{pmatrix} \frac{2\sqrt{2}}{3} & \frac{2\sqrt{2}}{\sqrt{3}} \end{pmatrix}.$$

3. Сигналы ГЕКС-64. Приведем точные формулы вероятности ошибки двумерного гексагонального набора при $M = 64$, $\Pi_c^2 = \frac{268}{141} = 1,900709$ [5, с. 16, рис. 1, d]:

$$P_e = \sum_{k=1}^3 a_k T(b_k \sqrt{h_{bc}^2}, c_k) + \sum_{k=1}^2 d_k Q(f_k \sqrt{h_{bc}^2}),$$

где

$$\bar{a} = \begin{pmatrix} \frac{75}{8} & \frac{3}{8} & -\frac{3}{16} \end{pmatrix}, \bar{b} = \begin{pmatrix} \frac{4}{\sqrt{47}} & \frac{4}{\sqrt{47}} & \frac{4\sqrt{3}}{\sqrt{47}} \end{pmatrix},$$

$$\bar{c} = \begin{pmatrix} \frac{1}{\sqrt{3}} & \sqrt{3} & \frac{1}{\sqrt{3}} \end{pmatrix}, \bar{d} = \begin{pmatrix} \frac{7}{32} & \frac{3}{32} \end{pmatrix},$$

$$\bar{f} = \begin{pmatrix} \frac{4}{\sqrt{47}} & \frac{4\sqrt{3}}{\sqrt{47}} \end{pmatrix}.$$

Для сигнальной конструкции, применяемой в модеме Codex/ESE SP 14.4 [4, с. 594, рис. 9.20, б], вероятность ошибки на символ

$$P_e = \sum_{k=1}^6 a_k T(b_k \sqrt{h_{bc}^2}, c_k) + \sum_{k=1}^3 d_k Q(f_k \sqrt{h_{bc}^2}),$$

где

$$\bar{a} = \begin{pmatrix} \frac{297}{32} & \frac{3}{8} & -\frac{3}{16} & \frac{3}{32} & -\frac{3}{32} & \frac{3}{32} \end{pmatrix},$$

$$\bar{b} = \begin{pmatrix} \frac{8}{9}\sqrt{\frac{3}{7}} & \frac{8}{9}\sqrt{\frac{3}{7}} & \frac{8}{3}\sqrt{\frac{1}{7}} & \frac{8}{3}\sqrt{\frac{1}{7}} & \frac{8\sqrt{3}}{9} & \frac{8}{9}\sqrt{\frac{3}{7}} \end{pmatrix},$$

$$\bar{c} = \begin{pmatrix} \frac{1}{\sqrt{3}} & \sqrt{3} & \frac{1}{\sqrt{3}} & \frac{5}{\sqrt{3}} & \sqrt{3} & 3\sqrt{3} \end{pmatrix},$$

$$\bar{d} = \begin{pmatrix} \frac{3}{16} & \frac{3}{64} & \frac{3}{64} \end{pmatrix}, \bar{f} = \begin{pmatrix} \frac{8}{9}\sqrt{\frac{3}{7}} & \frac{8}{3}\sqrt{\frac{1}{7}} & \frac{8\sqrt{3}}{9} \end{pmatrix},$$

$$\Pi_c^2 = \frac{1216}{567} = 2,144621.$$

Рассмотрим теперь методику расчета вероятности ошибки приема сигнальных конструкций при общих замираниях. Вероятность ошибки при общих замираниях определяется как

$$P = \int_0^{\infty} P_{e/b}(\sqrt{h_{c,u}^2}) \omega(\mu) d\mu, \quad (3a)$$

где $P_{e/b}$ — вероятность ошибки (в символе/бите) в канале с детерминированными параметрами и белым шумом; $\omega(\mu)$ — плотность распределения вероятностей; μ — коэффициент передачи. В общем случае вероятность ошибок двумерных сигналов при когерентном приеме в канале с детерминированными параметрами и белым шумом может быть представлена в виде [6]

$$P_{e/b}(h_{bc}^2) = \sum_k a_k T(\alpha_{k1} \sqrt{h_{bc}^2}, \eta_k) + \sum_k b_k Q(\alpha_{k2} \sqrt{h_{bc}^2}) + \sum_k c_k Q(\alpha_{k3} \sqrt{h_{bc}^2}) Q(\alpha_{k4} \sqrt{h_{bc}^2}), \quad (3б)$$

где $T(v, a)$, $v \geq 0, a \geq 0$ и $Q(x)$ соответственно функции Оуэна и Лапласа. Примеры, подтверждающие это положение, приведены в [2–4, 6, 7].

Основная сложность заключается в вычислении интеграла вида (3) от функций Лапласа, их произведения, а также от функции Оуэна и получении аналитических формул, удобных для проведения расчетов. Результаты получены в основном для относительно «простых» законов распределения — Релея и Накагами, меньше — для закона распределения Райса.

Дадим краткую характеристику методики расчета вероятности ошибки при медленных обших замираниях.

Пусть

$$h_{c,\mu}^2 = \frac{P_{c,\mu} T}{N_0} = \mu^2 \frac{P_{cп} T}{N_0}, \quad P_{c,\mu} = \mu^2 P_{cп},$$

где $P_{c,\mu}$ — мощность принятого сигнала, а $P_{cп}$ — мощность переданного сигнала. Тогда математическое ожидание мощности сигнала определяется как

$$\overline{P_{c,\mu}} = \int_0^\infty P_{c,\mu} \omega(\mu) d\mu = P_{cп} \int_0^\infty \mu^2 \omega(\mu) d\mu = m_2 P_{cп},$$

где $m_2 = \mu^2$ — начальный момент второго порядка.

Следовательно, $P_{cп} = \overline{P_{c,\mu}}/m_2$ и $h_{c,\mu}^2 = \mu^2 \frac{1}{m_2} \frac{\overline{P_{c,\mu}} T}{N_0} =$

$= \mu^2 \frac{1}{m_2} h_c^2$. При расчете по (3а) необходимо вместо величины h^2 , входящей в формулу для вероятности ошибки в гауссовом канале, подставить величину $h_{c,\mu}^2 = \mu^2 h_c^2 / m_2$ и вычислить полученный интеграл.

Наиболее общим законом распределения замираний является четырехпараметрический закон распределения вероятностей случайного коэффициента передачи канала μ [3]:

$$\omega(\mu_c, \mu_s) = \frac{1}{2\pi\sigma_c\sigma_s} \exp\left[-\frac{1}{2\sigma_c^2}(\mu_c - m_c)^2 - \frac{1}{2\sigma_s^2}(\mu_s - m_s)^2\right],$$

$$\mu = \sqrt{\mu_c^2 + \mu_s^2},$$

где m_c и m_s — математические ожидания квадратурных составляющих μ_c и μ_s ; $\mu_0 = \sqrt{m_c^2 + m_s^2}$ — регулярная составляющая коэффициента передачи; σ_c^2 и σ_s^2 — дисперсии квадратурных составляющих μ_c и μ_s . Следуя [7], наряду с параметрами $m_c, m_s, \sigma_c^2, \sigma_s^2$ будем использовать параметры, имеющие наглядный физический смысл.

1. Отношение дисперсий квадратурных составляющих σ_c^2 и σ_s^2 — величина $q^2 = \frac{\sigma_c^2}{\sigma_s^2}$. Коэффициент q^2 характеризует асимметрию канала по дисперсиям. Без ограничения общности рассматриваются значения q^2 из интервала $[0, 1]$, т. е. $0 \leq q^2 \leq 1$.

2. Фазовый угол $\varphi_0 = \arctg \frac{m_s}{m_c}$ или $\text{tg} \varphi_0 = \frac{m_s}{m_c}$.

3. Отношение средних мощностей регулярной и флуктуирующей частей сигнала — $\gamma^2 = \frac{m_c^2 + m_s^2}{\sigma_c^2 + \sigma_s^2} = \frac{\mu_0^2}{\sigma_c^2 + \sigma_s^2}$. Это выражение удобнее представить в

виде $\gamma^2 = \frac{2q^2}{1+q^2} \frac{\mu_0^2}{2\sigma_c^2} = \frac{2q^2}{1+q^2} \gamma_0^2$, где $\gamma_0^2 = \frac{\mu_0^2}{2\sigma_c^2}$ — вели-

чина, характеризующая глубину замираний в канале с райсовскими замираниями ($q^2 = 1$). Коэффици-

циент $\frac{2q^2}{1+q^2} \in [0, 1]$ характеризует уменьшение γ^2

по сравнению с величиной γ_0^2 . При релейских за-

мираниях $\gamma_0^2 = 0$, в канале без замираний $\gamma_0^2 \rightarrow \infty$

(присутствует только регулярная составляющая).

Следует заметить, что величина γ^2 может принимать

одинаковые значения при разных значениях γ_0^2

и q^2 . Очевидно, что для этого необходимо, что-

бы выполнялось тождество $\frac{\gamma_{0,1}^2}{\gamma_{0,2}^2} = \frac{1+q_1^{-2}}{1+q_2^{-2}}$. Кроме

этого справедливы следующие соотношения:

$$\gamma^2 = \frac{m_c^2}{\sigma_c^2} \frac{q^2}{1+q^2} \frac{1}{\cos^2 \varphi_0} \quad \text{или} \quad \gamma^2 = \frac{m_s^2}{\sigma_s^2} \frac{1}{1+q^2} \frac{1}{\sin^2 \varphi_0}.$$

4. Средний квадрат коэффициента передачи (начальный момент второго порядка) $m_2 = \mu_0^2 +$

$+ \sigma_c^2 + \sigma_s^2$ или $m_2 = 2\sigma_c^2 \left(1 + \frac{\mu_0^2}{2\sigma_c^2} + \frac{1-q^2}{2q^2}\right) =$

$= 2\sigma_c^2 \left(1 + \gamma_0^2 + \frac{1-q^2}{2q^2}\right)$, $q^2 \neq 0$. Если $q^2 = 0$, то $\sigma_c^2 = 0$,

а величина σ_s^2 является неопределенной, либо $\sigma_s^2 \rightarrow \infty$, а величина σ_c^2 является неопределенной.

Многочисленные теоретические работы и экспериментальные данные показывают, что общая гауссова модель и ее частные случаи охватывают широкий класс каналов связи в различных диапазонах волн [3, 4, 6, 7]. Сложность вычисления вероятностей ошибок для четырехпараметрического закона замираний привела к тому, что на практике традиционно используются только плотности распределения Релея и Райса.

Одномерное распределение коэффициента передачи канала μ может быть определено по формуле [7]

$$\omega(\mu) = \frac{\mu}{2\pi\sigma_c\sigma_s} \times \int_0^{2\pi} \exp\left[-\frac{1}{2\sigma_c^2}(\mu\cos\varphi - m_c)^2 - \frac{1}{2\sigma_s^2}(\mu\sin\varphi - m_s)^2\right] d\varphi, \mu \geq 0.$$

В результате несложных преобразований четырехпараметрическое распределение может быть представлено в виде

$$\omega(\mu) = q \exp\left[-\frac{m_s^2}{2\sigma_s^2}\right] \times \sum_{m=0}^{\infty} \left(\frac{(-1)^m H_{2m}(-i\chi)}{2^m (2m)!!} (1-q^2)^m \omega_{m+1}^{RN}(\mu) \right),$$

где $q^2 = \frac{\sigma_c^2}{\sigma_s^2}$, $\chi^2 = \frac{m_s^2}{2\sigma_s^2} \frac{q^2}{1-q^2}$ и $\omega_p^{RN}(\mu)$ — распределение Райса–Накагами [1]:

$$\omega_p^{RN}(\mu) = \frac{(\beta\mu)^p}{\theta^{p-1}} \exp\left[-\frac{\theta^2}{2\beta} - \frac{\beta}{2}\mu^2\right] I_{p-1}(\theta\mu), \mu \geq 0, \quad (4)$$

где $p > 0, \theta \geq 0, \beta > 0$ — параметры распределения, а $I_{p-1}(\theta\mu)$ — функция Бесселя от мнимого аргумента порядка $(p-1)$.

Распределение Райса–Накагами (4) при соответствующем выборе параметров p, θ, β совпадает с распределениями Релея, Райса и Накагами

(m -распределение): при $p=1, \theta=0, \beta=\frac{1}{\sigma^2}$ — распределение Релея; при $p=1, \theta=\frac{\mu_0}{\sigma^2}, \beta=\frac{1}{\sigma^2}$ — распределение Райса; при $p=m, \theta=0, \beta=2m/\mu^2$ — распределение Накагами (при выборе $\theta=0$ следу-

ет учесть, что $\lim_{\theta \rightarrow 0} \frac{I_{p-1}(\theta\mu)}{\theta^{p-1}} = \frac{1}{2^{p-1}} \frac{1}{\Gamma(p)}$, где $\Gamma(x)$ — гамма-функция).

Если положить $\theta = \frac{m_c}{\sigma_c^2}$ и $\beta = \frac{1}{\sigma_c^2}$, то

$$\omega_{k+1}^{RN}(\mu) = \frac{\mu^{k+1}}{\sigma_c^2} \frac{1}{(m_c)^k} \exp\left[-\frac{m_c^2 + \mu^2}{2\sigma_c^2}\right] I_k\left(\frac{m_c}{\sigma_c^2}\mu\right).$$

Начальный второй момент распределения Райса–Накагами определяется по формуле

$$m_2 = \frac{2p}{\beta} \exp\left(-\frac{\theta^2}{2\beta}\right) {}_1F_1\left(p+1; p; \frac{\theta^2}{2\beta}\right).$$

Если $p \in \mathbb{N}$, то нетрудно убедиться, что

$$m_2 = \frac{2}{\beta} \left(p + \frac{\theta^2}{2\beta} \right).$$

Рассмотрим более подробно задачу вычисления интеграла от функции Оуэна и функции Лапласа при четырехпараметрических замираниях:

$$M_\omega T(\alpha, \eta) = \int_0^\infty T(\alpha\mu, \eta) \omega(\mu) d\mu, \quad (5)$$

где

$$\omega(\mu) = q \exp\left[-\frac{m_s^2}{2\sigma_s^2}\right] \times \sum_{m=0}^{\infty} \left(\frac{(-1)^m H_{2m}(-i\chi)}{2^m (2m)!!} (1-q^2)^m \omega_{m+1}^{RN}(\mu) \right),$$

параметр $\alpha = \sqrt{\frac{2gh_{bc}^2}{m_2}}$, величина g определяется

в зависимости от сигнальной конструкции, значение $m_2 = \mu^2$ — начальный момент второго порядка — определяется для четырехпараметрического закона распределений замираний как

$$m_2 = 2\sigma_c^2 \left(1 + \gamma_0^2 + \frac{1-q^2}{2q^2} \right).$$

Для распределения Райса–Накагами (1) справедливо соотношение [6, 8, 9]

$$\int_0^\infty T(\alpha\mu, \eta) \omega_p^{RN}(\mu) d\mu = \mathcal{H}_p(z, b, \eta), \quad (6a)$$

где

$$\mathcal{H}_p(z, b, \eta) = \frac{(1-b^2)^p}{2\pi} \int_0^\eta \frac{1}{1+x^2} \frac{1}{(1+b^2x^2)^p} \times \exp\left(-\frac{z^2}{2} \frac{1+x^2}{1+b^2x^2}\right) dx,$$

$$\eta \geq 0, p \in \mathbb{R}, p \geq 0 \quad (6б)$$

и $b^2 = \frac{\alpha^2}{\alpha^2 + \beta}$, $z^2 = \frac{\theta^2}{\beta} b^2$, $0 \leq b^2 \leq 1$. Учитывая, что \mathcal{H} -функция для случая $p=1$ может применяться для расчета при замираниях Релея и Райса, то без ограничения общности можно положить, что $\mathcal{H}_1(z, b, \eta) \triangleq \mathcal{H}(z, b, \eta)$.

Возможно его альтернативное представление в виде

$$\begin{aligned} \mathcal{H}_p(z, b, \eta) &= \frac{(1-b^2)^p}{2\pi} \times \\ &\times \int_0^{\text{arctg}(\eta)} \frac{\cos^{2p} t}{(1-(1-b^2)\sin^2 t)^p} \exp\left(-\frac{z^2}{2} \frac{1}{1-(1-b^2)\sin^2 t}\right) dt \end{aligned} \quad (6в)$$

или в виде

$$\begin{aligned} \mathcal{H}_p(z, b, \eta) &= b \frac{(1-b^2)^p}{2\pi} \times \\ &\times \int_0^{\text{arctg}(b\eta)} \frac{\cos^{2p} t}{1-(1-b^2)\cos^2 t} \exp\left(-\frac{z^2}{2b^2} (1-(1-b^2)\cos^2 t)\right) dt, \end{aligned} \quad (6г)$$

$b \neq 0$.

Учитывая введенную \mathcal{H} -функцию, получаем, что

$$\int_0^\infty T(\alpha\mu, \eta) \omega_{m+1}^{RN}(\mu) d\mu = \mathcal{H}_{m+1}(z_c, b_c, \eta),$$

где $b_c^2 = \frac{\alpha^2 \sigma_c^2}{1 + \alpha^2 \sigma_c^2}$, $z_c^2 = \frac{\alpha^2 m_c^2}{1 + \alpha^2 \sigma_c^2}$, так как $\theta = \frac{m_c}{\sigma_c^2}$,

$\beta = \frac{1}{\sigma_c^2}$ и $\frac{\theta^2}{2\beta} = \frac{m_c^2}{2\sigma_c^2}$. Следовательно, для соотношения (5) получаем, что

$$\begin{aligned} M_\omega T(\alpha, \eta) &= q \exp\left[-\frac{m_s^2}{2\sigma_s^2}\right] \times \\ &\times \sum_{m=0}^\infty \left(\frac{(-1)^m H_{2m}(-i\chi)}{2^m (2m)!!} (1-q^2)^m \mathcal{H}_{m+1}(z_c, b_c, \eta) \right) \end{aligned}$$

или

$$\begin{aligned} M_\omega T(\alpha, \eta) &= q \frac{1-b_c^2}{2\pi} \exp\left[-\frac{m_s^2}{2\sigma_s^2}\right] \times \\ &\times \int_0^\eta \frac{1}{1+x^2} \frac{1}{1+b_c^2 x^2} \exp\left(-\frac{z_c^2}{2} \frac{1+x^2}{1+b_c^2 x^2}\right) \times \\ &\times \sum_{m=0}^\infty \left(\frac{(-1)^m H_{2m}(-i\chi)}{2^m (2m)!!} \left[\frac{(1-q^2)(1-b_c^2)}{1+b_c^2 x^2} \right]^m \right). \end{aligned}$$

Положим $b_s^2 = \frac{\alpha^2 \sigma_s^2}{1 + \alpha^2 \sigma_s^2}$, $z_s^2 = \frac{\alpha^2 m_s^2}{1 + \alpha^2 \sigma_s^2}$, тогда $z_s^2 = \frac{m_s^2}{\sigma_s^2} b_s^2$, $q^2 = \frac{b_c^2}{b_s^2} \frac{1-b_s^2}{1-b_c^2}$ и, следовательно:

$$\begin{aligned} M_\omega T(\alpha, \eta) &= \frac{\sqrt{1-b_c^2} \sqrt{1-b_s^2}}{2\pi} \times \\ &\times \int_0^\eta \frac{1}{1+x^2} \frac{1}{\sqrt{1+b_c^2 x^2}} \frac{1}{\sqrt{1+b_s^2 x^2}} \times \\ &\times \exp\left(-\frac{z_c^2}{2} \frac{1+x^2}{1+b_c^2 x^2} - \frac{z_s^2}{2} \frac{1+x^2}{1+b_s^2 x^2}\right) dx. \end{aligned}$$

Определим новую специальную интегральную функцию

$$\begin{aligned} \mathcal{S}(z_c, z_s, b_c, b_s, \eta) &= \frac{\sqrt{1-b_c^2} \sqrt{1-b_s^2}}{2\pi} \times \\ &\times \int_0^\eta \frac{1}{1+x^2} \frac{1}{\sqrt{1+b_c^2 x^2}} \frac{1}{\sqrt{1+b_s^2 x^2}} \times \\ &\times \exp\left(-\frac{z_c^2}{2} \frac{1+x^2}{1+b_c^2 x^2} - \frac{z_s^2}{2} \frac{1+x^2}{1+b_s^2 x^2}\right) dx, \end{aligned} \quad (7)$$

где $z_c^2 = \frac{\alpha^2 m_c^2}{1 + \alpha^2 \sigma_c^2}$, $z_s^2 = \frac{\alpha^2 m_s^2}{1 + \alpha^2 \sigma_s^2}$, $b_c^2 = \frac{\alpha^2 \sigma_c^2}{1 + \alpha^2 \sigma_c^2}$,

$b_s^2 = \frac{\alpha^2 \sigma_s^2}{1 + \alpha^2 \sigma_s^2}$. В частности, для распределения Райса при симметрии канала по дисперсиям квадратурных составляющих: $q^2 = 1$, т. е. $b_c^2 = b_s^2$, введенная функция совпадает с \mathcal{H} -функцией [2]:

$$\begin{aligned} \mathcal{S}(z_c, z_s, b_c, b_s, \eta) &= \mathcal{H}\left(\sqrt{z_c^2 + z_s^2}, b_c, \eta\right) = \\ &= \frac{1-b_c^2}{2\pi} \int_0^\eta \frac{1}{1+x^2} \frac{1}{1+b_c^2 x^2} \times \\ &\times \exp\left(-\frac{z_c^2 + z_s^2}{2} \frac{1+x^2}{1+b_c^2 x^2}\right) dx, \end{aligned}$$

где $z_c^2 + z_s^2 = \frac{\alpha^2 (m_c^2 + m_s^2)}{1 + \alpha^2 \sigma_c^2} = \frac{\alpha^2 \mu_0^2}{1 + \alpha^2 \sigma_c^2}$. Соответственно, для распределения Релея

$$\mathcal{S}(z_c, z_s, b_c, b_s, \eta) = \mathcal{H}(0, b_c, \eta) = \mathcal{H}\left(0, \frac{\alpha\sigma}{\sqrt{1+\alpha^2\sigma^2}}, \eta\right).$$

Преобразуем аргументы функции, учитывая, что

$\alpha = \sqrt{\frac{2gh_{bc}^2}{m_2}}$ и $m_2 = 2\sigma_c^2 \left(1 + \gamma_0^2 + \frac{1-q^2}{2q^2} \right)$. Определим

$$\tilde{h}_{bc}^2 = \frac{h_{bc}^2}{1 + \gamma_0^2 + \frac{1-q^2}{2q^2}}, \text{ т. е.}$$

$$h_{bc}^2 [\text{дБ}] = \tilde{h}_{bc}^2 [\text{дБ}] + 10 \lg \left(1 + \frac{1+q^2}{2q^2} \gamma^2 + \frac{1-q^2}{2q^2} \right),$$

тогда

$$z_c^2 = \frac{2gh_{bc}^2 m_c^2}{m_2 + 2gh_{bc}^2 \sigma_c^2} = \frac{m_c^2}{\sigma_c^2} \frac{g\tilde{h}_{bc}^2}{1 + g\tilde{h}_{bc}^2},$$

$$z_s^2 = \frac{2gh_{bc}^2 m_s^2}{m_2 + 2gh_{bc}^2 \sigma_s^2} = \frac{m_s^2}{\sigma_s^2} \frac{g\tilde{h}_{bc}^2}{q^2 + g\tilde{h}_{bc}^2}, \quad b_c^2 = \frac{g\tilde{h}_{bc}^2}{1 + g\tilde{h}_{bc}^2},$$

$$b_s^2 = \frac{g\tilde{h}_{bc}^2}{q^2 + g\tilde{h}_{bc}^2} \quad (8)$$

и

$$\frac{m_c^2}{\sigma_c^2} = \cos^2 \varphi_0 \frac{1+q^2}{q^2} \gamma^2, \quad \frac{m_s^2}{\sigma_s^2} = \sin^2 \varphi_0 (1+q^2) \gamma^2.$$

Для численных расчетов на ЭВМ удобнее использовать альтернативное определение \mathcal{S} -функции:

$$\begin{aligned} \mathcal{S}(z_c, z_s, b_c, b_s, \eta) &= \frac{\sqrt{1-b_c^2} \sqrt{1-b_s^2}}{2\pi} \times \\ &\times \int_0^{\text{arctg}(\eta)} \frac{\cos^2 t}{\sqrt{1-(1-b_c^2)\sin^2 t} \sqrt{1-(1-b_s^2)\sin^2 t}} \times \\ &\times \exp \left[-\frac{1}{2} \left(\frac{z_c^2}{1-(1-b_c^2)\sin^2 t} + \frac{z_s^2}{1-(1-b_s^2)\sin^2 t} \right) \right] dt. \end{aligned} \quad (9)$$

Таким образом, для четырехпараметрического распределения математическое ожидание функции Оуэна

$$\int_0^\infty T(\alpha\mu, \eta) \omega(\mu) d\mu = \mathcal{S}(z_c, z_s, b_c, b_s, \eta),$$

где $\alpha^2 = \frac{2gh_{bc}^2}{m_2}$ и $m_2 = 2\sigma_c^2 \left(1 + \gamma_0^2 + \frac{1-q^2}{2q^2} \right)$, а аргументы функции определены в (8).

Получим теперь соотношения для усреднения функции Лапласа и их произведения через \mathcal{S} -функцию. Если положить $\eta = +\infty$, то из свойств функции Оуэна следует, что

$$\int_0^\infty T(\alpha\mu, +\infty) \omega(\mu) d\mu = \frac{1}{2} \int_0^\infty Q(\alpha\mu) \omega(\mu) d\mu$$

или для четырехпараметрического распределения

$$\int_0^\infty Q(\alpha\mu) \omega(\mu) d\mu = 2\mathcal{S}(z_c, z_s, b_c, b_s, +\infty).$$

Для определения соотношения $\int_0^\infty Q(\alpha\mu) Q(\beta\mu) \times$

$\times \omega(\mu) d\mu$, где $\alpha^2 = \frac{2g_1 h_{bc}^2}{m_2}$ и $\beta^2 = \frac{2g_2 h_{bc}^2}{m_2}$, воспользу-

емся одним из свойств функции Оуэна, которое преобразуем к виду

$$\begin{aligned} Q(\alpha\mu) Q(\beta\mu) &= T(\alpha\mu, +\infty) + T(\beta\mu, +\infty) - \\ &- \left[T\left(\alpha\mu, \frac{\beta}{\alpha}\right) + T\left(\beta\mu, \frac{\alpha}{\beta}\right) \right]. \end{aligned}$$

Тогда

$$\begin{aligned} \int_0^\infty Q(\alpha\mu) Q(\beta\mu) \omega(\mu) d\mu &= \\ &= \left(\mathcal{S}(z_c, z_s, b_c, b_s, +\infty) - \mathcal{S}\left(z_c, z_s, b_c, b_s, \frac{\beta}{\alpha}\right) \right) + \\ &+ \left(\mathcal{S}(\tilde{z}_c, \tilde{z}_s, \tilde{b}_c, \tilde{b}_s, +\infty) - \mathcal{S}\left(\tilde{z}_c, \tilde{z}_s, \tilde{b}_c, \tilde{b}_s, \frac{\alpha}{\beta}\right) \right). \end{aligned}$$

При численных расчетах на ЭВМ удобнее, возвращаясь к определению \mathcal{S} -функции, ввести функцию

$$\begin{aligned} \mathcal{S}\left(z_c, z_s, b_c, b_s, \begin{bmatrix} \eta_2 \\ \eta_1 \end{bmatrix}\right) &= \frac{\sqrt{1-b_c^2} \sqrt{1-b_s^2}}{2\pi} \times \\ &\times \int_{\eta_1}^{\eta_2} \frac{1}{1+x^2} \frac{1}{\sqrt{1+b_c^2 x^2}} \frac{1}{\sqrt{1+b_s^2 x^2}} \times \\ &\times \exp \left(-\frac{z_c^2}{2} \frac{1+x^2}{1+b_c^2 x^2} - \frac{z_s^2}{2} \frac{1+x^2}{1+b_s^2 x^2} \right) dx. \end{aligned}$$

Тогда

$$\begin{aligned} \int_0^\infty Q(\alpha\mu) Q(\beta\mu) \omega(\mu) d\mu &= \mathcal{S}\left(z_c, z_s, b_c, b_s, \begin{bmatrix} +\infty \\ \beta/\alpha \end{bmatrix}\right) + \\ &+ \mathcal{S}\left(\tilde{z}_c, \tilde{z}_s, \tilde{b}_c, \tilde{b}_s, \begin{bmatrix} +\infty \\ \alpha/\beta \end{bmatrix}\right). \end{aligned}$$

В частности, при $\alpha = \beta$: $\int_0^\infty Q^2(\alpha\mu) \omega(\mu) d\mu =$

$$= 2\mathcal{S}\left(z_c, z_s, b_c, b_s, \begin{bmatrix} +\infty \\ 1 \end{bmatrix}\right).$$

Поскольку $\mathcal{S}(z_c, z_s, b_c, b_s, \eta) = \mathcal{H}(\sqrt{z_c^2 + z_s^2}, b_c, \eta)$, то некоторые свойства \mathcal{S} -функции совпадают со свойствами \mathcal{H} -функции, приведенными, например, в работе [2]. Кроме этого, можно показать, что новая специальная функция в частном случае выражается через эллиптические функции. Действительно, сделаем замену $xb_c = -\text{ctg}\varphi$, тогда функция может быть записана в виде

$$\mathcal{S}(z_c, z_s, b_c, b_s, \eta) = \frac{\sqrt{1-b_c^2}\sqrt{1-b_s^2}}{2\pi} \frac{1}{b_s} \times \int_{\pi/2}^{\text{arctg}(-\eta b_c)} \frac{b_c^2 \sin^2 \varphi}{(1-u^2 \sin^2 \varphi)\sqrt{1-v^2 \sin^2 \varphi}} \times \exp\left[-(1-u^2 \sin^2 \varphi) \left\{ \frac{z_c^2}{2b_c^2} + \frac{z_s^2}{2b_s^2} \frac{1}{1-v^2 \sin^2 \varphi} \right\}\right] d\varphi,$$

где $u^2 = 1 - b_c^2$, $v^2 = 1 - \frac{b_c^2}{b_s^2}$. Если $z_c = z_s = 0$, то

$$\mathcal{S}(0, 0, b_c, b_s, \eta) = \frac{\sqrt{1-b_c^2}\sqrt{1-b_s^2}}{2\pi} \frac{1}{b_s} \times \int_{\pi/2}^{\text{arctg}(-\eta b_c)} \frac{b_c^2 \sin^2 \varphi}{(1-u^2 \sin^2 \varphi)\sqrt{1-v^2 \sin^2 \varphi}} d\varphi$$

или

$$\mathcal{S}(0, 0, b_c, b_s, \eta) = \frac{1}{2\pi} \frac{b_c^2}{b_s} \sqrt{\frac{1-b_s^2}{1-b_c^2}} \times$$

$$\times \left\{ \left[\Pi(\varphi_1, u^2, v) - \Pi(\varphi_0, u^2, v) \right] - \left[F(\varphi_1, v) - F(\varphi_0, v) \right] \right\},$$

где $\Pi(\varphi, u^2, v) = \int_0^\varphi \frac{1}{(1-u^2 \sin^2 \varphi)\sqrt{1-v^2 \sin^2 \varphi}} d\varphi$ —

эллиптический интеграл 3-го рода; $F(\varphi, v) =$

$$= \int_0^\varphi \frac{1}{\sqrt{1-v^2 \sin^2 \varphi}} d\varphi$$
 — эллиптический интеграл

1-го рода и $\varphi_1 = \text{arctg}(-\eta b_c)$, $\varphi_0 = \pi/2$. Если вместо параметров b_c^2 и b_s^2 использовать величины u^2 , v^2 , то специальная функция может быть представлена в виде

$$\mathcal{S}(z_c, z_s, u, v, \eta) = \frac{1}{2\pi} u \sqrt{1-u^2} \sqrt{u^2-v^2} \times \int_{\pi/2}^{\text{arctg}(-\eta b_c)} \frac{\sin^2 \varphi}{(1-u^2 \sin^2 \varphi)\sqrt{1-v^2 \sin^2 \varphi}} \times \exp\left[-\frac{1-u^2 \sin^2 \varphi}{2(1-u^2)} \left\{ z_c^2 + \frac{1-v^2}{1-v^2 \sin^2 \varphi} z_s^2 \right\}\right] d\varphi.$$

Если использовать замену $x = \text{ctg}\varphi$ и ввести параметры $\kappa_c^2 = \frac{1-b_c^2}{b_c^2}$, $\kappa_s^2 = \frac{1-b_s^2}{b_s^2}$, то

$$\mathcal{S}(z_c, z_s, \kappa_c, \kappa_s, \eta) = \frac{\kappa_c \kappa_s}{2\pi} u \sqrt{1-u^2} \sqrt{u^2-v^2} \times \int_{\text{arctg}(\eta)}^{\pi/2} \frac{\sin^2 \varphi}{\sqrt{1+\kappa_c^2 \sin^2 \varphi} \sqrt{1+\kappa_s^2 \sin^2 \varphi}} \times \exp\left[-\left(\frac{z_c^2}{2} \frac{1+\kappa_c^2}{1+\kappa_c^2 \sin^2 \varphi} + \frac{z_s^2}{2} \frac{1+\kappa_s^2}{1+\kappa_s^2 \sin^2 \varphi} \right)\right] d\varphi,$$

где величины u^2 , v^2 были определены выше.

Используя введенную специальную интегральную функцию и примеры, нетрудно получить соответствующие расчеты вероятности ошибки при общих четырехпараметрических замираниях.

Таким образом, введена новая специальная интегральная \mathcal{S} -функция, которая включает в себя \mathcal{H} -функцию, функции Лапласа, Оуэна, Никольсона, (обобщенного) Маркума. Введенная \mathcal{S} -функция позволяет систематизировать вычисление вероятности ошибки в канале с общими четырехпараметрическими замираниями и белым шумом при когерентном приеме и проводить корректное сравнение сигнальных конструкций.

Литература

1. Справочник по специальным функциям / Под ред. А. Абрамовица и И. Стиган. М.: Наука, 1979. 832 с.
2. Савищенко Н. В. Многомерные сигнальные конструкции: их частотная эффективность и помехоус-

тойчивость приема: Монография / Под ред. Д. Л. Бураченко. СПб.: Изд-во Политехн. ун-та, 2005. 420 с.

3. Коржик В. И., Финк Л. М., Щелкунов К. Н. Расчет помехоустойчивости систем передачи дискретных сообщений: Справочник. М.: Радио и связь, 1981. 232 с.

4. Склад Б. Цифровая связь. Теоретические основы и практическое применение: Пер. с англ. М.: Вильямс, 2003. 1104 с.
5. Багушев С. В., Зайцев И. Е., Яковлев А. А. Перспективы развития сигнально-кодовых каналов для гауссовского канала связи // Зарубежная радиоэлектроника. 1990. № 1. С. 15–31.
6. Прокис Дж. Цифровая связь: Пер. с англ. / Под ред. Д. Д. Кловского. М.: Радио и связь, 2000. 800 с.
7. Кловский Д. Д. Передача дискретных сообщений по радиоканалам. М.: Радио и связь, 1982. 304 с.
8. Прудников А. П., Брычков Ю. А., Маричев О. И. Интегралы и ряды. Элементарные функции. М.: Наука, 1981. 800 с.
9. Прудников А. П., Брычков Ю. А., Маричев О. И. Интегралы и ряды. Специальные функции. М.: Наука, 1983. 752 с.



Красильников Н. Н., Красильникова О. И.

Мультимедиа-технологии в информационных системах. Представление и обработка изображений в компьютере: учебное пособие/Н. Н. Красильников, О. И. Красильникова. ГУАП. — СПб., 2007. — 132 с.: ил.
 ISBN 978-5-8088-0257-5

В учебном пособии изложены вопросы, связанные с представлением трехмерной и двумерной графики. Приведены статистические характеристики двумерных изображений. Рассмотрены вопросы оцифровки изображений и возникающие при этом искажения. Описаны методы линейной и нелинейной обработки изображений с целью повышения их качества.

Учебное пособие предназначено для студентов старших курсов, изучающих мультимедиа-технологии в рамках технических специальностей.

СИСТЕМНЫЙ АНАЛИЗ. ВЗГЛЯД НА ПРОШЛОЕ И НАСТОЯЩЕЕ

Проблемы рационального выбора в уникальных ситуациях существовали всегда, но по ряду причин в последние десятилетия важность их значительно увеличилась. Прежде всего, резко возрос динамизм окружающей среды и уменьшился период времени, когда принятые ранее решения остаются правильными. Во-вторых, развитие науки и техники привело к появлению большого числа альтернативных вариантов выбора. В-третьих, усложнился каждый из вариантов принимаемых решений. В-четвертых, увеличилась взаимозависимость различных решений и их последствий и т. д.

Необходимость решения этих проблем вызвала к жизни множество методов, подходов, которые постепенно накапливались, развивались, обобщались, образовав, в конце концов, определенную технологию анализа системности мира вообще и системности человеческого познания в частности.

Теоретической базой для такого анализа явилась возникшая в конце 40-х годов 20 века при решении военно-стратегических задач теория принятия решений. Методы теории принятия решений начали интенсивно развиваться, постепенно распространяясь и на область экономического управления.

Одной из первых отечественных монографий, где обобщался опыт построения и анализа сложных систем, накопленный к концу 60-х годов, является монография Н. П. Бусленко¹. В ней выделены два класса задач, возникающих в связи с исследованием сложных систем:

- 1) задача анализа, в ходе которой изучаются свойства системы в зависимости от ее структуры;
- 2) задача синтеза, которая сводится, в том числе, к выбору структуры и значений параметров.

Задачи синтеза оптимальных структур сложных систем обладают целым рядом существенных особенностей, что предьявляет определенные требования к возможным методам их решения. В частности, методы синтеза должны:

- 1) учитывать особенности задач построения оптимальных структур сложных систем, в том числе условия совместимости, нелинейный характер критериев, необходимость построения усеченного множества Парето и другие;
- 2) быть идейно близкими друг другу;
- 3) обладать высокой эффективностью, допускать получение как точного, так и приближенного (с заданной точностью) решения;
- 4) обеспечивать легкость программирования.

¹ Бусленко Н. П. Моделирование сложных систем. М.: Наука: Гл. ред. физ.-мат. лит., 1968. 356 с.

Исследования показали, что важнейшей задачей начального этапа является построение структуры системы. Поэтому в дальнейшем было проведено рассмотрение и представление структуры в виде графа, даны типичные структурные конфигурации, например комплексы, контуры. Раскрыта сущность структурного анализа системы, который, в частности, включает анализ связности; перечень входных, выходных полюсов; тип структуры и т. п.

Раскрыта проблема координации в многоуровневой системе, введен «постулат совместимости», который означает согласованность между целями системы и ее подсистем.

Подробное исследование и анализ структуры многоуровневой системы дали возможность выделить ее существенные особенности, такие как элементный состав, наличие связей, инвариантность во времени, а также установить, что наиболее удобным представлением структуры является граф (гиперграф). Кроме того, это позволило показать основные этапы синтеза топологической структуры: анализ элементов, связей, определение диаметра структуры, степени диаметра структуры, степени централизации, сложности.

Значительный вклад в развитие теории анализа и синтеза сложных систем внес А. Д. Цвиркун¹. В его работах рассматриваются структуры автоматизированных, информационно-управляющих систем организационного типа, производственно-транспортных систем и сетей вычислительных центров. Дана классификация моделей и методов синтеза структуры этих систем. В математическом плане они сводятся к задачам дискретного программирования, в частности, к задачам о назначении. С учетом специфики моделей предлагаются алгоритмы, основанные на идеях методов ветвей и границ.

В работе Н. Н. Моисеева «Математические задачи системного анализа»² сделан важный вывод о том, что использование методологии системного анализа при проектировании сложных систем привело к необходимости постановки задачи синтеза сложных систем с заранее заданными качествами, которая представляет собой новую и достаточно сложную проблему. Для ее решения необходимо создать новые методы, которые бы позволяли решать задачу преобразования информации о множестве всех допустимых вариантов будущей си-

¹ Цвиркун А. Д. Основы синтеза структуры сложных систем. М.: Наука, 1982. 200 с.

² Моисеев Н. Н. Математические задачи системного анализа. М.: Наука. Гл. ред. физ.-мат. лит., 1981. 488 с.

системы в информацию о конструктивном облике системы, наиболее соответствующей поставленным целям. Автор этой работы также считает, что основой, на которой происходит реализация методов решения таких задач, является теория нечетких множеств, предложенная американским ученым Л. Заде в 1965 году.

С тех пор благодаря усилиям «первопроходцев» и последователей опубликовано множество научных трудов, посвященных теоретическим и практическим аспектам этой теории.

В значительном количестве публикаций как зарубежных, так и отечественных авторов отражены не только теоретические аспекты, но и такие практические приложения рассматриваемой области как распознавание образов, принятие решений, системный анализ, инженерное искусство, процессы управления и т. д.

Первой отечественной монографией, посвященной изложению основных положений теории Л. Заде и ее развитию, явилась книга С. А. Орловского «Проблемы принятия решений при нечеткой исходной информации»¹. В ней сделана попытка показать, каким образом нечеткую, качественную информацию можно использовать в формализованных процедурах анализа и принятия решений. В предисловии к этому изданию академик Н. Н. Моисеев акцентировал внимание на то, что развитие идей методов ветвей и границ на основе теории нечетких множеств Л. Заде будет серьезным шагом в развитии методов последовательного анализа.

Использование и развитие основных положений теории нечетких множеств при построении экспертных систем, систем поддержки принятия решений, интеллектуальных систем, а также систем согласования решений, вырабатываемых различными группами специалистов, нашли свое отражение в трудах академика Э. А. Трахтенгерца². Причем особо отмечено, насколько плодотворно применение теории нечетких множеств на ранних стадиях проектирования сложных систем.

Дальнейшее исследование вопросов синтеза систем привело к необходимости развития научного направления, которое занимается разработкой математических моделей и методов решения задач с ограниченной неопределенностью, — интервального анализа.

Достаточно молодой и бурно развивающийся интервальный анализ находится на стыке вычислительной математики, информатики и кибернетики.

Приоритет в исследованиях интервального анализа принадлежит отечественному ученому академику Л. В. Канторовичу, идеи которого при-

менительно к задачам оптимизации развил А. А. Ватолин¹, сформулировавший для них определение множества решений. Среди работ, имеющих отношение к нестатистическому оцениванию параметров эмпирических зависимостей, выделяются работы А. П. Воцинина. Достаточно большое внимание аналитическому синтезу регуляторов в условиях неопределенности параметров объекта управления уделил Н. А. Хлебалин.

Анализ работ отечественных и зарубежных авторов, в которых рассматриваются вопросы исследования теории систем и системного анализа, свидетельствует о том, что:

- проблема синтеза оптимальных структур сложных систем особо острой стала в конце 60-х — начале 70-х годов. Предпосылкой к этому явились потребности практики в создании высокоэффективных систем управления информационными комплексами различных типов;

- широкое использование современных достижений теории графов позволило на достаточном уровне решить проблему структурного анализа системы;

- универсальных методов синтеза оптимальных структур сложных систем с заранее заданными качествами не существует, что объясняется, в том числе, высокой сложностью исследуемых моделей, их уникальностью, повышенными требованиями к качеству проекта, многокритериальностью, в общем случае нелинейностью целевых функций и ограничений;

- задачи синтеза оптимальной структуры, как правило, относятся к области дискретного программирования. Для их решения используют методы полного перебора, потоковые алгоритмы, методы случайного поиска с различными эвристическими правилами, комбинаторные методы, в том числе методы ветвей и границ.

Однако автономное использование классических методов дискретного программирования (которые не учитывают специфики многокритериальных задач) и методов теории принятия решений (позволяющих анализировать альтернативы по многим критериям в целом, не вникая в структуру комбинаторной модели) не давало возможности в полной мере решать рассматриваемую проблему. Путь к ее решению был найден профессором В. В. Сафроновым² на основе комплексного применения теории принятия решений и методов ветвей и границ, сочетания их отличительных свойств. В результате им разработан метод, названный обобщенным методом ветвей и границ, который позволяет по желанию исследователей стро-

¹ Орловский С. А. Проблемы принятия решений при нечеткой исходной информации. М.: Наука, 1981. 208 с.

² Трахтенгерц Э. А. Методы генерации, оценки и согласования решений в распределительных системах поддержки принятия решений // Автоматика и телемеханика. 1995. № 4. С. 3–52.

¹ Ватолин А. А. О задачах линейного программирования с интервальными коэффициентами // ЖВМ и МФ. Т. 24. № 11. 1984. С. 1629–1637.

² Сафронов В. В. Проблемы проектирования сложных технических систем и некоторые пути их решения // Доклады Академии военных наук. Сер. Аналитическая теория автоматического управления. 1999. № 1.

ить множество Парето, подмножество Парето с заданной мощностью с расположением элементов множества (подмножества) в порядке убывания приоритета (метод «жесткого» ранжирования); решать однокритериальные и сводимые к ним задачи.

Завершая анализ современного состояния исследований в области системного анализа и теории принятия решений, хотелось бы отметить, что основным методом системного анализа является расчленение большой неопределенности на более обозримые, лучше поддающиеся исследованию (что соответствует понятию анализ) при сохранении целостного системного (т. е. всестороннего и комплексного) представления об объекте исследования.

Традиционные математические методы описал еще Гёте:

*«Любой предмет желая изучить,
чтоб ясное о нем познать получить,
Ученый прежде душу изымает,
затем предмет на части расчленяет
И видит их.
Да жаль, духовная их связь тем временем
исчезла, унеслась.»*

Следует обратить внимание читателей, что как раз задачей системного анализа в отличие от традиционных математических методов и является сохранить эту «духовную связь» между объектами, составляющими единое целое, что позволяет сделать вывод о том, что системный анализ — это «здоровый смысл, подкрепленный определенными формализованными методами».

И еще, «что тревожит в последнее время, так это — чрезмерная засоренность отечественного языка теории управления иностранными терминами». Поэтому авторы считают необходимым процитировать слова видного ученого, доктора технических наук, профессора В. А. Подчукаева¹, высказанные им в предисловии к сборнику трудов

Международной научной конференции «Аналитическая теория автоматического управления»:

«...ползучая экспансия западных интерпретаторов задач советских (российских) ученых имеет своей целью изменение персонификации полученных ими результатов. В качестве примеров достаточно привести следующие факты. Теория автоматического управления — одна из целого ряда научных дисциплин, зародилась в России. Ее возникновение принято связывать с именем почетного члена Петербургской академии наук (1888) Ивана Алексеевича Вышеградского (1831–1895). Среди многих заслуг И. А. Вышеградского особое место занимает введение в учебный процесс курсового и дипломного проектирования, до сих пор являющихся отличительной особенностью русской высшей школы.

...Матрица управляемости получена А. Н. Крыловым на 30 лет раньше Р. Калмана, а термин Н. Н. Красовского «грубость свойства асимптотической устойчивости» заменен термином «робастность». Неуважение светлой памяти выдающихся отечественных ученых выливается порою в просто неприличные иностранные названия».

Отдельно следует остановиться на статье В. А. Подчукаева «Необходимые и достаточные условия оптимальности в линейно-квадратичной игре двух лиц» (1978). Уравнение Риккати, полученное в ней, лишь десятилетием позже было повторено на Западе. А сейчас поток публикаций, связанных с этим уравнением, называют теорией H^∞ .

На взгляд авторов, не изучать труды своих соотечественников, не замечать приоритета советской (российской) науки — непозволительно.

Ю. В. Ведерников,
канд. техн. наук, доцент
В. В. Матросов
начальник отделения
научно-вычислительного отдела
Михайловская военная
артиллерийская академия

¹ Подчукаев В. А. Программа дальнейших исследований в области анализа систем автоматического управления // Тр. Междунар. науч. конф. Сер. Аналитическая теория автоматического управления. 2000. С. 61–67.

**Правительство и Законодательное собрание Санкт-Петербурга
Министерство образования и науки РФ
Министерство РФ по информационным технологиям и связи
Санкт-Петербургский научный центр Российской академии наук
Отделение информационных технологий и вычислительных систем РАН
Санкт-Петербургский институт информатики и автоматизации РАН
при поддержке Совета Безопасности РФ и
Полномочного представителя Президента РФ в Северо-Западном федеральном округе
представляют**

**V САНКТ-ПЕТЕРБУРГСКАЯ МЕЖРЕГИОНАЛЬНАЯ КОНФЕРЕНЦИЯ
«ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ РЕГИОНОВ РОССИИ — ИБРР-2007»**

23–25 октября 2007 г.

**Место проведения конференции: Санкт-Петербургский институт информатики и автоматизации РАН
Адрес: 199178, Санкт-Петербург, В.О., 14-я линия, 39**

Направления работы

Государственная политика обеспечения информационной безопасности регионов России

Правовые аспекты информационной безопасности

Безопасные информационные технологии
Современные средства защиты информации
Информационная безопасность телекоммуникационных сетей

Подготовка и переподготовка кадров в области обеспечения информационной безопасности

Круглый стол: «Информационная безопасность — информационная культура XXI века»

Условия участия в конференции

Для участия в работе конференции необходимо выслать в адрес Оргкомитета:

1. Заявку предприятия (организации) на участие в работе конференции (образец формы заявки см.: <http://www.spiiras.nw.ru/conf.htm>).

2. Тезисы доклада объемом не более одной страницы формата А4 без формул, рисунков и ссылок на литературу. При оформлении тезисов следует придерживаться следующей схемы расположения текста:

Фамилия И. О. (строчные буквы)

Страна, город, организация (строчные буквы)

НАЗВАНИЕ ДОКЛАДА (прописные буквы)

<Текст тезисов доклада>

3. Электронный вариант тезисов доклада на дискете (для иногородних — предварительно переданный по электронной почте: lai@iias.spb.su или spiran@iias.spb.su) должен быть подготовлен в виде файла с расширением *.rtf в редакторе Word с использованием нормального шрифта Arial размера 12 через один интервал без шрифтовых выделений, переносов в словах и разбивки на страницы.

4. Копию платежного поручения предприятия (организации) с банковской отметкой о перечислении организационного целевого взноса Регио-

нальной общественной организации «Санкт-Петербургское общество информатики, вычислительной техники, систем связи и управления» (РОО СПОИСУ).

Издание трудов конференции

Планируется издание сборника докладов.

Финансирование подготовки и проведения конференции

Финансирование подготовки и проведения конференции осуществляется за счет целевых организационных взносов (добровольных пожертвований) учредителей и организаций-соустроителей; минимальный размер взноса составляет 2000 рублей.

В соответствии с утвержденной сметой расходов целевые взносы будут расходоваться на подготовку и проведение конференции, издание программы, трудов конференции и других информационных материалов.

Контрольные сроки

Заявки предприятий (организаций) на участие в конференции принимаются до 15 октября 2007 г. включительно.

Тезисы, поступившие в Оргкомитет до 21 сентября 2007 г., будут опубликованы до начала конференции.

Дополнительная информация и справки

Оргкомитет V Санкт-Петербургской межрегиональной конференции «ИБРР-2007»:

199178, Санкт-Петербург, В.О., 14-я линия, 39. Санкт-Петербургский институт информатики и автоматизации РАН

Тел.: (812) 328-19-19, 328-33-11

Факс: (812) 328-44-50

Эл. почта: lai@iias.spb.su; spiran@iias.spb.su

Сайт: <http://www.spiiras.nw.ru/conf.htm>

АНДРЕЕВ
Сергей
Дмитриевич



Аспирант кафедры безопасности информационных систем Санкт-Петербургского государственного университета аэрокосмического приборостроения. В 2006 году окончил Санкт-Петербургский государственный университет аэрокосмического приборостроения по специальности «Комплексная защита объектов информатизации». Является автором шести научных публикаций. Область научных интересов — телекоммуникационные системы, случайный множественный доступ, системы массового обслуживания.

ВЕЛИКОХАТЬКО
Юрий
Петрович



Научный сотрудник отдела надежности НИИ электротехнических устройств. В 1960 году окончил Высшее военно-морское училище им. М. В. Фрунзе. Является автором пяти научных публикаций. Область научных интересов — теория графов.

МИРОНОВ
Анатолий
Анатолевич



Доцент кафедры технологий приборостроения Санкт-Петербургского государственного университета информационных технологий, механики и оптики. В 1980 году окончил Ленинградский механический институт. В 1990 году защитил диссертацию на соискание ученой степени кандидата технических наук. Является автором 12 научных публикаций. Область научных интересов — автоматизированное проектирование.

НОВИКОВ
Федор
Александрович



Доцент кафедры прикладной математики Санкт-Петербургского государственного политехнического университета. В 1974 году окончил математико-механический факультет Ленинградского государственного университета. В 1983 году защитил диссертацию на соискание ученой степени кандидата физико-математических наук. Является автором 40 научных публикаций. Область научных интересов — прикладная математика, технологии программирования.

ПАВЛОВ
Владислав
Станиславович



Доцент кафедры вычислительных и электронных систем Санкт-Петербургского государственного университета аэрокосмического приборостроения. В 1987 году окончил Ленинградский институт авиационного приборостроения. В 1992 году защитил диссертацию на соискание ученой степени кандидата технических наук. Является автором 85 научных публикаций, в том числе двух монографий и семи авторских свидетельств на изобретения. Область научных интересов — локационные системы, пространственно-временная обработка информационных сигналов, синтез и анализ помехоустойчивых систем управления, теоретические основы функционирования, математические модели, структуры и алгоритмы радиоэлектронных систем.

САВИНОВ
Максим
Владимирович



Ассистент кафедры вычислительных и электронных систем Санкт-Петербургского государственного университета аэрокосмического приборостроения. В 2003 году окончил Санкт-Петербургский государственный университет аэрокосмического приборостроения. Является автором десяти научных публикаций. Область научных интересов — моноимпульсные пеленгационные системы.

САВИЩЕНКО

**Николай
Васильевич**



Профессор Военной академии связи.

В 1986 году окончил Новосибирский государственный университет, в 1989 году — Военную академию связи.

В 2006 году защитил диссертацию на соискание ученой степени доктора технических наук.

Является автором 93 научных публикаций.

Область научных интересов — математическое моделирование каналов связи, помехоустойчивость сигналов, многомерные сигнальные конструкции.

СЕМЁНОВ

**Сергей
Александрович**



Аспирант кафедры безопасности информационных систем Санкт-Петербургского государственного университета аэрокосмического приборостроения. В 2006 году окончил Санкт-Петербургский государственный университет аэрокосмического приборостроения по специальности «Комплексная защита объектов информатизации».

Является автором двух научных публикаций.

Область научных интересов — надежная передача информации, сжатие и передача видео в реальном времени.

СТЕПАНОВ

**Олег
Георгиевич**



Программист компании Jet-Brains s. r. o.

В 2006 году окончил Санкт-Петербургский государственный университет информационных технологий, механики и оптики. Магистр прикладной математики.

Область научных интересов — технологии программирования, объектно-ориентированное программирование, конечные автоматы, статический анализ программ.

СТЕПАНЯН

**Карлен
Багратович**



Архитектор проектов фирмы Astrosoft Development.

В 2003 году окончил Санкт-Петербургский государственный политехнический университет. Является автором двух научных публикаций.

Область научных интересов — спецификация и построение диаграмм.

ТИХОНОВ

**Эдуард
Прокофьевич**



Доцент кафедры биомедицинской электроники и охраны среды Санкт-Петербургского государственного электротехнического университета, член-корреспондент Метрологической академии.

В 1962 году окончил Ленинградский институт авиационного приборостроения.

Является автором более 190 научных публикаций, в том числе более 50 авторских свидетельств и патентов на изобретения.

Область научных интересов — кибернетика, информатика, моделирование, информационно-измерительные системы, биомедицинская инженерия.

ТЮРЛИКОВ

**Андрей
Михайлович**



Доцент кафедры информационных систем Санкт-Петербургского государственного университета аэрокосмического приборостроения.

В 1980 году окончил Ленинградский институт авиационного приборостроения по специальности «Информационные системы управления».

В 1986 году защитил диссертацию на соискание ученой степени кандидата технических наук. Является автором 60 научных публикаций.

Область научных интересов — многоабонентные системы связи, системы дистанционного обучения, протоколы передачи данных в реальном масштабе времени, алгоритмы сжатия видеоинформации.

**ШАЛЫТО
Анатолий
Абрамович**


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

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

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

Область научных интересов — системы логического управления, автоматное программирование.

**ШОПЫРИН
Данил
Геннадьевич**


Доцент кафедры компьютерных технологий Санкт-Петербургского государственного университета информационных технологий механики и оптики.

В 2002 году окончил Оренбургский государственный университет.

Является автором десяти научных публикаций.

Область научных интересов — объекто-ориентированное и автоматное программирование.

Уважаемые авторы!

Журнал «Информационно-управляющие системы» входит в «Перечень ведущих рецензируемых научных журналов и изданий, в которых должны быть опубликованы основные научные результаты диссертации на соискание ученой степени доктора и кандидата наук».

Мы будем рады сотрудничеству с Вами и надеемся, что Вы порекомендуете библиотеке Вашей организации подписаться на наш журнал.

При подготовке рукописей статей редакция просит Вас руководствоваться следующими рекомендациями.

Объем статьи (текст, таблицы, иллюстрации и библиография) не должен превышать эквивалента в 16 страниц, напечатанных на бумаге формата А4 на одной стороне через 1,5 интервала в Word шрифтом Times New Roman размером 13.

Обязательными элементами оформления статьи являются: индекс УДК, заглавие и аннотация (5–7 строк) на русском и английском языках, инициалы и фамилия автора (авторов), ученая степень, звание, полное название организации.

Простые формулы, например: $rd2 = St - 2\Phi1$ {<m>, набирайте в Word, **не используя** формульный редактор (Mathtype или Equation), для набора символов Ψ δ \geq \subseteq \Rightarrow \supset β и т. п. выбирайте на панели «Вставка», затем «Символ», затем ищите нужный знак в одном из разделов (Symbol, Symath, различные Euclid). Выполнение этого требования особенно важно для символов и формул, которые набраны в текстовой строке (при верстке в PageMaker набранные в Mathtype или Equation символы очень трудно выровнять, а при распечатке можно и потерять). Формулы, стоящие в отдельной строке, могут быть набраны как угодно. Только в том случае, если собственные средства Word не позволяют набрать формулу, используйте

те имеющийся в Word формульный редактор Mathtype или Equation; при наборе формул в формульном редакторе знаки препинания (.,;) в конце формулы набирайте вместе с формулой. Не используйте два редактора для набора одной формулы.

Для установки размера шрифта никогда не пользуйтесь вкладкой Other..., используйте вкладку Define; в формулах не отделяйте пробелами знаки: +, =, -.

При наборе символов в тексте помните, что символы, обозначаемые латинскими буквами, набираются светлым курсивом, русскими и греческими — светлым прямым, векторы — прямым полужирным шрифтом.

Иллюстрации в текст не заверстываются и предоставляются отдельными *исходными* файлами, поддающимися редактированию:

– рисунки, графики, диаграммы, блок-схемы изготавливаются в векторных программах: Visio 4, 5, 2002-2003 (*.vsd); Coreldraw (*.cdr); Excel; Word; AdobeIllustrator; AutoCad; Компас; Matlab (экспорт в формат *.ai);

– фото и растровые — в формате *.tif, *.png с максимальным разрешением (не менее 300 pixels/inch).

Обязательно составляйте подрисовочные подписи, не повторяющие дословно комментарии к рисункам в тексте статьи.

УДК 681.314

Алгоритмическое описание и сравнительный анализ свойств сигма-дельта АЦП (Часть 1)

Тихонов Э. П. Информационно-управляющие системы, 2007. № 4. С. 2–12.

Предложено аналитическое описание алгоритма работы сигма-дельта АЦП в виде нелинейного отображения, на основании которого осуществлено исследование его характеристик и выполнены аналитическими методами и посредством имитационного моделирования сравнительный анализ особенностей его функционирования.

Список лит.: 26 назв.

УДК 519.174

Однозначно раскрашиваемые графы с минимумом ребер

Великохатко Ю. П., Миронов А. А. Информационно-управляющие системы, 2007. № 4. С. 13–16.

Построен граф с количеством ребер, минимально допустимым для однозначно K -раскрашиваемого графа. Показано, как при помощи операций двух типов указанный граф можно преобразовать в множество всех графов этого класса (для произвольно заданного числа вершин). Доказано, что все такие графы $(K - 1)$ -связны. Найдено выражение для числа ребер в таких графах, а также даны уравнения, полезные для практических целей, в частности, для различных задач оптимизации.

Список лит.: 3 назв.

УДК 621.396.69

Синтез дискриминатора пеленгационной системы с эллипсообразным расположением точек приема

Павлов В. С., Савинов М. В. Информационно-управляющие системы, 2007. № 4. С. 17–21.

Проводится двумерный статистический синтез оптимального трехканального дискриминатора пеленгационной системы для случая, когда ее пространственные характеристики могут быть представлены через совокупность точек приема, расположенных по окату эллипса в плоскости пеленгования. Исследованы условия, при которых формируемые дискриминатором оценки угловых расхождений в азимутальной и угломестной плоскостях независимы между собой, с учетом поворота совокупности точек приема на произвольный угол.

Список лит.: 6 назв.

UDK 681.314

The algorithmic description and the comparative analysis of the sigma-delta analog-to-digital converter (Part 1)

Tikhonov E. P. IUS, 2007. N 4. P. 2–12.

We propose a non-linear mapping that describes the algorithm of work of the delta-sigma analog-to-digital converter. This model is used to investigate the characteristics of the converter and give a comparative analysis of its features.

Refs: 26 titles.

UDK 519.174

Uniquely colorable graphs with minimum number of edges

Velikohatiko Y. P., Mironov A. A. IUS, 2007. N 4. P. 13–16.

In this paper a uniquely K -colorable graph with minimum possible number of edges is being constructed. It is shown that the graph can be transformed into the set of all possible graphs of that kind (for any specified number of vertices) using only two kinds of transformations. It has been proven that all such graphs are $(K - 1)$ -connected. An expression for the number of edges in such graphs is derived as well as some equations that have practical applications, in particular, in solving various optimization problems.

Refs: 3 titles.

UDK 621.396.69

Synthesis of the discriminator of the direction finding for bearing system with an elliptic arrangement of reception points having an ellipse receiving points position

Pavlov V. S., Savinov M. V. IUS, 2007. N 4. P. 17–21.

The paper gives a two-dimensional statistical synthesis of the optimum three-channel discriminator for the direction finding system in the case when its spatial characteristics can be presented as the set of reception points located on an ellipse in the plane of direction finding. We investigate the conditions under which estimations of angular mismatches formed by the discriminator in azimuthal and elevation planes are independent between themselves, taking into account the rotation of reception points through an arbitrary angle.

Refs: 6 titles.

УДК 681.3.06

Предметно-ориентированный язык автоматного программирования на базе динамического языка RUBY

Степанов О. Г., Шалыто А. А., Шопырин Д. Г. Информационно-управляющие системы, 2007. № 4. С. 22–27.

Решается задача преобразования диаграмм переходов, применяемых в SWITCH-технологии, в исполняемый код. Предлагается использование динамических языков программирования, позволяющих добиться изоморфности диаграммы и соответствующего программного кода, что ведет к уменьшению количества ошибок при указанном преобразовании. На базе динамического языка Ruby с использованием разработанной библиотеки STROBE создан предметно-ориентированный язык автоматного программирования.

Список лит.: 14 назв.

УДК 004.434

Язык описания диаграмм

Новиков Ф. А., Степанян К. Б. Информационно-управляющие системы, 2007. № 4. С. 28–36.

Предлагается язык описания диаграмм, позволяющий формально определить графический синтаксис (нотацию) диаграмм заданного типа и связать нотацию с семантикой, заданной в форме набора классов. Язык может быть применен для автоматической генерации визуализаторов и графических редакторов диаграмм различных типов.

Список лит.: 7 назв.

УДК 004.7

Методики оценки параметров радиоканала

Андреев С. Д., Семенов С. А., Тюрликов А. М. Информационно-управляющие системы, 2007. № 4. С. 37–43.

Ставится задача измерения пропускной способности и вероятности потери пакета в радиоканале. Уточняется, что в ряде практических приложений необходимо измерять значения пропускной способности с высокой дискретизацией. Предлагаются две методики получения искомых значений, удовлетворяющих выдвинутым требованиям, и обсуждаются границы их применимости.

Список лит.: 9 назв.

УДК 681.3.06

An object-oriented language of automaton programming based on the dynamcal language RUBY

Stepanov O. G., Shalyto A. A., Shopyrin. IUS, 2007. N 4. P. 22–27.

This paper solves the problem of transforming transition diagrams used in SWITCH technology into executable code. We suggest using dynamic programming languages to solve this problem because these languages allow to achieve an isomorphism between the source diagram and the resulting code. This leads to decreasing the number of errors that appear during such transition. We also present an extension library STROBE and a domain-specific automata programming language based on the dynamic programming language Ruby.

Refs: 14 titles.

УДК 004.434

A diagram definition language

Novikov F. A., Stepanyan K. B. IUS, 2007. N 4. P. 28–36.

A Diagram Definition Language (DiaDeL) is proposed. The paper contains textual specifications for graphical notation and its binding to semantics. The semantics is defined as a set of classes. DiaDeL could be used for automatic generation of various diagram visualizers and editors.

Refs: 7 titles.

УДК 004.7

Methods of estimation of radiochannel parameters

Andreev S. D., Semenov S. A., Tyurlikov A. M. IUS, 2007. N 4. P. 37–43.

The paper addresses the problem of the wireless channel throughput and packet error rate estimation. It is shown that some practical applications require the throughput estimation values obtained with high sampling rate. Two methods are introduced to obtain the sought values with high sampling rate and the limits of their applicability are discussed.

Refs: 9 titles.

УДК 621.39

Помехоустойчивость модемов с двумерными сигнальными конструкциями по точным формулам вероятности ошибки в канале без замираний и с общими четырехпараметрическими замираниями

Савищенко Н. В. Информационно-управляющие системы, 2007. № 4. С. 44–54.

Рассматривается помехоустойчивость модемов с двумерными сигнальными конструкциями в канале без замираний и с общими замираниями и белым шумом при когерентном приеме. Введена новая специальная интегральная \mathcal{S} -функция, которая включает в себя \mathcal{H} -функцию, функции Лапласа, Оуэна, Никольсона, (обобщенного) Маркума. Введенная \mathcal{S} -функция позволяет систематизировать вычисление вероятности ошибки в канале с общими четырехпараметрическими замираниями и белым шумом при когерентном приеме.

Список лит.: 9 назв.

UDK 621.39

Noise immunity of modems with two-dimensional signal constructions based on precise formulas of fadingless channel-error-probability and with common four-parameter fadings

Savischenko N. V. IUS, 2007. N 4. P. 44–54.

We discuss the noise immunity of modems with two-dimensional signal constructions in fadingless channels, channels with common fadings and with white noise during coherent reception. A new special integral S-function with includes the H-function, Laplace's, Owen's, Nicolson's, (generalized) Marcum's fuctions is introduced. The S-function allows for a systematic calculation of error probability in the channel with common four-parameter fadigs and white noise during coherent reception.

Refs: 9 titles.

ПАМЯТКА ДЛЯ АВТОРОВ

Поступающие в редакцию статьи проходят обязательное рецензирование.

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

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

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

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