НАУЧНЫЙ ЖУРНАЛ

ТЕЛЕКОММУНИКАЦИИ И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

№2-2017

(*Дата издания: декабрь 2017 г.*)

УЧРЕДИТЕЛЬ:

ОРДЕНА ТРУДОВОГО КРАСНОГО ЗНАМЕНИ ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧЕРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «МОСКОВСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ СВЯЗИ И ИНФОРМАТИКИ» (МТУСИ)

РЕДАКЦИОННАЯ ПОДГОТОВКА:

Отдел организации научно-исследовательской работы студентов (ОНИРС МТУСИ)

СОДЕРЖАНИЕ

«Цифровые технологии радиосвязи и телерадиовещания»

«NGN: сетевые технологии и системы телекоммуникаций»

СОВРЕМЕННАЯ СВЕРХУЗКОПОЛОСНАЯ СИСТЕМА ПЕРЕДАЧИ ДАННЫХ «НА КРИСТАЛЛЕ» LORA

Болдина Валерия Игоревна
студентка группы БРА1401 МТУСИ
boldinavaleriya@gmail.com

Фролов Алексей Андреевич
уч. секретарь НИЧ МТУСИ
a.a.frolov@mtuci.ru

Одним из перспективных направлений развития инфокоммуникационных сетей являются сети «Интернета вещей» (IoT – Internet of Things). Концепцией «Интернета вещей» является объединение в общую (единую) сеть физических объектов, имеющих доступ к сети общего пользования и выполняющих функции анализа параметров окружающей среды, анализа обрабатываемых, передаваемых данных и взаимодействие с окружающей средой. Такие сети называют сетями межмашинного взаимодействия (M2M-Machine-to-Machine). Современным решением проблемы построения сетей M2M и IoT являются сети LoRaWAN, физический уровень которых построен по принципам технологии LoRa.

Ключевые слова: *Интернет вещей, система «на кристалле», технология LoRa, энергоэффективная передача данных, сверхузкополосная система.*

История появления технологии LoRa

Исследовательский центр IBM Research и Semtech Corporation в начале 2015 года представили новый открытый энергоэффективный сетевой протокол LoRaWAN (Long Range Wide Area Networks), который позволяет организовать обмен данными между устройствами на достаточно большом расстоянии при помощи технологии LoRa.

Системы связи M2M, построенные на основе технологии LoRa, имеют преимущества перед системами Wi-Fi, Bluetooth, WiMaX [12-18] и сотовыми сетями передачи данных.

Созданный в начале 2015 г., LoRa Alliance стандартизировал для малопотребляющих глобальных сетей (Low Power Wide Area Networks, LPWAN) протокол сетей межмашинного взаимодействия LoRaWAN [7].

LoRaWAN – это открытый протокол для высокоемких сетей (в одной сети до 1 000 000 устройств) с большим радиусом действия (до 10-15 км на открытой местности) и низким энергопотреблением. LoRaWAN-сеть организована как сеть типа звезда и включает различные классы (A, B и C) узлов для оптимизации компромисса между скоростью доставки информации и сроком работы при батарейном питании.

Реализация сети на основе LoRaWAN происходит за счет полной двухсторонней связи между узлами и обладает специальными методами шифрования для обеспечения надежности и безопасности системы.

Сеть LoRaWAN можно представить в виде оконечных устройств (точек, узлов) данные с которых передаются (в зашифрованном виде!) на шлюзы, далее на сервер сети провайдера и далее на сервер приложений провайдера, откуда информация поступает к пользователю [6, 7].

Спецификация определяет 3 класса оконечных устройств LoRaWAN, отличающиеся друг от друга режимами приема:

- A (узел передает данные на шлюз короткими посылками по заданному графику);
- B (узел включает приемник по графику, заданному сервером);
- C (окно приема открыто постоянно и закрывается только на период кратковременной передачи данных).

Устройства данных классов являются двунаправленными. Класс A является базовым и должен поддерживаться всеми устройствами.

Диапазон областей применений данной технологии огромен: от домашней автоматизации и интернета вещей (Internet of Things, IoT) («умного дома») до промышленности и умных городов.

Анализ характеристик LoRa

В рабочем диапазоне часто 860-1020 МГц (США, Европа) система LoRa позволяет демодулировать сигналы на уровне 20 дБ ниже уровня шумов, тогда как большинство систем с частотной манипуляцией (frequency shift keying, FSK) могут корректно работать с сигналами на уровне не ниже 8-10 дБ над уровнем шумов.

Дальность связи системы LoRa в реальных условиях на реальной местности может быть:

- на открытой местности и при высоком расположении антенны LoRa обеспечивает дальность больше заявленных 30 км;
- на пересеченной местности она падает до 1-2 км даже на минимальной скорости.

Большую дальность связи система LoRa позволяет обеспечить не только за счет параметров передатчика системы, но и благодаря высокой чувствительности приемника (-148 дБм). При этом приемопередатчик системы LoRa потребляет достаточно мало электроэнергии по сравнению с приемопередатчиками систем Wi-Fi, WiMAX, Bluetooth, ZigBee и т.п.

Рассматриваемая система занимает полосы частот шириной от 125 до 500 кГц. Известны диапазоны частот, в которых применяется система LoRa: в Европе – 860 МГц; в США – 915 и 780 МГц в странах Азии [11]. В России для организации межмашинного взаимодействия в диапазоне 0,1-10 ГГц выделены диапазоны частот в пределах 433,92 МГц; 915 МГц; 2,45 ГГц и 5,8 ГГц [5]. Для работы в каждом из этих диапазонов оборудование рассматриваемой системы имеет соответствующие режимы работы с соответствующими характеристиками.

Диапазоны частот 2,45 и 5,8 ГГц находятся в "нелицензируемой" части радиочастотного спектра. Отсутствие надобности в получении лицензии делает эти полосы частот привлекательными для применения систем LoRa. В данных диапазонах частот применяются системы межмашинного взаимодействия ZigBee, Bluetooth, WiFi и др. Они уступают системе LoRa по энергоэффективности и дальности действия. Среди рассмотренных систем межмашинного взаимодействия система LoRa является сверхузкополосной. Основные параметры известных систем M2M и системы LoRa представлены в табл. 1.

Таблица 1

Параметр	LoRa	ZigBee [8]	Bluetooth [2]	Wi-Fi [2]
Частотный диапазон, МГц	860-1020 2450±1% 5800±3%	686 / 915 / 2400	2400-2483	2412-2484 5,15-5,35; 5,65-6,425
Битовая скорость, кбит/с	0,24-37,5	20 / 40 / 250	721	11000 / 54000
Тип модуляции сигнала	LoRa	BPSK / QPSK	FHSS	DSSS
Ширина полосы пропускания, кГц	125-500	5200	1000	40000 / 160000
Длительность сигнала	5 мс	2 нс	4 мкс	800 нс
Выходная мощность, дБм	20	-32..0	20 / 4 / 0	
Дальность	До 30 км	200 м	100 м	100 м
Коэффициент широкополосности	0,00002041 (СУП)	0,002164 (УП)	0,0004165 (УП)	0,007273 (УП) / 0,031496 (ШП) [5]
Коэффициент расширение спектра	6-12	6-26	20	1-7
Чувствительность приемника, дБм	-117..-148	-92..-85	-90	-65 / -79

*УП – узкополосная система, ШП – широкополосная система, СУП – сверхузкополосная система

Структурная схема системы LoRa

Особенность технологии передачи LoRa в том, что в ней совмещено расширение спектра за счёт помехоустойчивого кодирования и применения сигналов с большой базой.

На рисунке 1 представлена структурная схема системы LoRa. В передатчике данные кодируются с помощью FEC-кодера и поступают на блок управления цифрового формирователя ЛЧМ-сигнала.

Цифровой формирователь ЛЧМ-сигнала построен на основе цифрового вычислительного синтезатора (ЦВС), в котором формируются квадратурные компоненты выходного сигнала с ЛЧМ-модуляцией. База такого сигнала, в зависимости от режима работы, варьируется от 64 до 4096.

Радиоприёмное устройство LoRa построено на основе согласованного фильтра (СФ) ЛЧМ-сигнала, порогового устройства (ПУ) и декодера Витерби. Сигнал, поступающий на приемную систему, усиливается в малошумящем усилителе (МШУ). Далее с помощью экстрактора [3] и вычитателя компенсируются помехи, сосредоточенные в рабочем частотном канале.

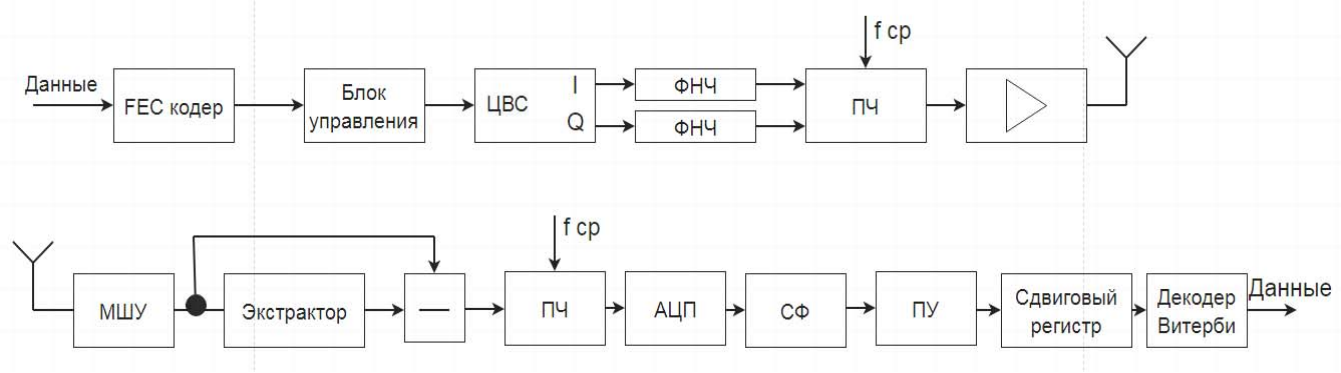


Рис. 1. Структурная схема системы LoRa

После этого сигнал переносится в область низких частот с помощью преобразователя частоты (ПЧ) и осуществляется приведение сигнала к цифровому виду с помощью (АЦП). После согласованной фильтрации ЛЧМ-сигнала принятые биты записываются в сдвиговый регистр, где формируется последовательность бит для декодирования.

Применение ЛЧМ-сигнала для передачи данных позволяет сделать приёмник устойчивым к отклонению частоты от номинального значения и снизить требования к тактовому генератору.

Применение кодов, корректирующих ошибки, позволяет повысить отношение сигнал/шум и увеличить устойчивость работы в условиях импульсных помех.

Особенностью данной схемы приемника является экстрактор, который компенсирует сосредоточенные помехи. Это позволяет системе LoRa работать в одном диапазоне с такими системами как GSM, UMTS, LTE, WiFi, Bluetooth, ZigBee с большим радиусом действия и низким энергопотреблением.

Описанные выше особенности системы LoRa позволяют организовать связь на расстоянии до 30-ти км, при этом экономично расходуя батарею питания (в случае с автономными узлами сети).

Характеристики оборудования LoRa

Готовые решения для организации сети LoRa, представленные на рынке, ориентированы в основном на Европу, США и работают в диапазонах 863 и 902 МГц [4].

Компания Semtech представила семейство RF-трансиверов SX127x, поддерживающее технологию LoRa, для нового рынка M2M/IoT. Эти приемопередатчики работают в диапазоне 860-1000 МГц и 137-960 МГц.

В таблице 2 представлены основные характеристики RF-трансиверов семейства SX127x [1, 9].

Таблица 2

Наименование	SX1272	SX1273	SX1276	SX1277	SX1278	SX1279
Диапазон работы частот, МГц	860-1020	860-1020	137-1020	137-1020	137-525	137-960
Коэффициент распространения спектра	6-12	6-9	6-12	6-9	6-12	6-12
Ширина полосы пропускания, кГц	125-500	125-500	7,8-500	7,8-500	7,8-500	7,8-500
Скорость передачи при использовании модуляции LoRa, кбит/с	0,24-37,5	1,7-37,5	0,018-37,5	0,11-37,5	0,018-37,5	0,018-37,5
Чувствительность передатчика, дБм	-117…-137	-117…-130	-111…-148	-111…-139	-111…-148	-111…-148
Выходная мощность, дБм	+20					
Доступные типы модуляции	FSK, GFSK, MSK, GMSK, OOK, LoRa					

На рисунке 2 представлено приемопередающее устройство «на кристалле» системы LoRa, построенное в соответствии со структурной схемой, рассмотренной на рис. 1.

Структурная схема трансивера SX1272/3 содержит типовые блоки:

- Приемный или передающий тракт (радиоприемный тракт реализован по схеме с однократным квадратурным преобразованием на низкую промежуточную частоту);
- Схема формирования частот;
- Интерфейс ввода/вывода с конфигурационными регистрами (подсистеме питания).

Такая схема построения позволяет реализовать лучшие характеристики устройства по чувствительности и избирательности по соседнему каналу по сравнению со схемой прямого преобразования.

Выходная мощность системы LoRa равна 20 дБм. Благодаря применению помехоустойчивого кодирования можно достичь значений коэффициента расширения спектра от 6 до 12, при этом чувствительность приемника увеличивается на 19 дБ, относительно параметров систем Wi-Fi, ZigBee, Bluetooth и др. Системе LoRa обеспечивает связь до 30 км. Такая дальность действия достигается при длительности посылки порядка 5 мс. При таких параметрах система LoRa обеспечивает энергоэффективную передачу данных и позволяет работать автономным узлам сети продолжительное время (несколько лет на одном аккумуляторе типа AA) [8].

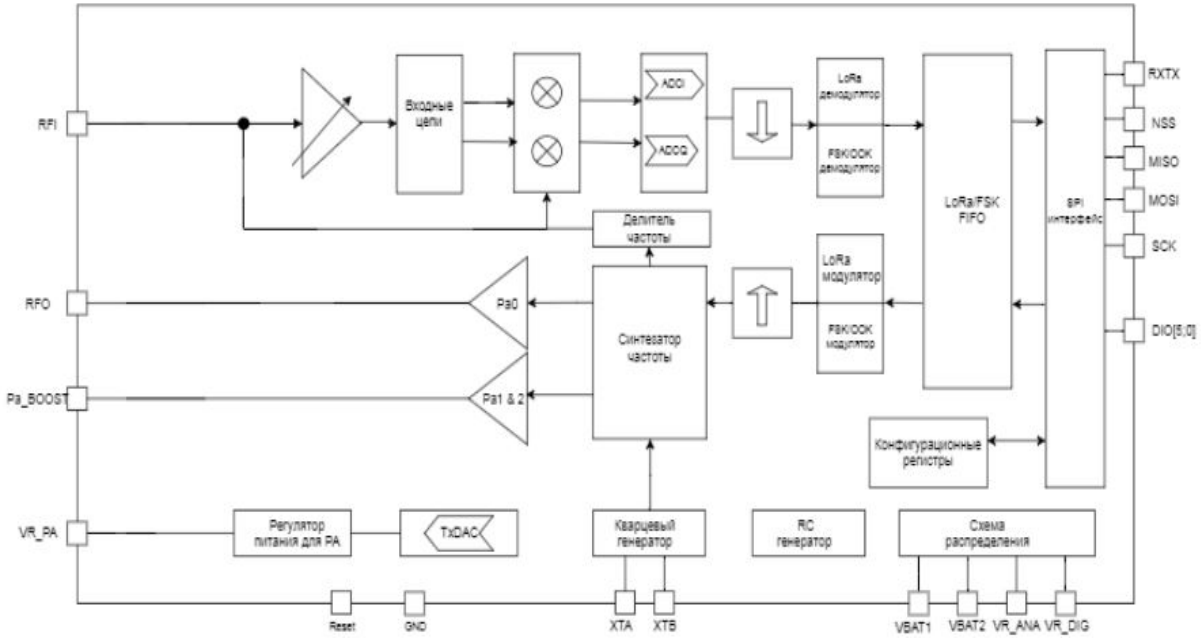


Рис. 2. Приемопередающее устройство «на кристалле» SX1272/3

Применение устройств серии SX127x является универсальным. Внутренние регистры памяти позволяют динамически изменять рабочую частоту, её девиацию, битрейт, вид модуляции, выходную мощность и многие другие параметры, а также устанавливать режимы работы всех периферийных блоков, что позволяет использовать один и тот же беспроводной модуль для решения разных задач.

Выводы

1. Технология LoRa перспективна для применения в сетях M2M и IoT, так как позволяет повысить их эффективность.
2. Система LoRa обеспечивает дальность до 30 км на открытой местности и до 1-2 км на пересеченной местности.
3. Использование нелицензируемой части радиочастотного спектра диапазонов 2,45 и 5,8 ГГц позволяет упростить развертывание сетей M2M и IoT и исключить затраты на получение лицензий на использование диапазона частот.
4. Благодаря своей высокой чувствительности (-148 дБм) и возможности работы в диапазонах 2,4 и 5 ГГц система LoRa идеально подходит для работы с низким потреблением электроэнергии, высокой проницаемостью сигнала и большой дальностью, а применение ЛЧМ-сигнала делают модемы LoRa простыми в реализации.

5. Использование помехоустойчивого кодирования и наличие экстрактора в структурной схеме приемника делает систему устойчивой к помехам, как внутри диапазона используемых частот, так и вне его.

6. Применение для передачи данных сигналов с дискретно-частотной модуляцией представляет большой интерес для дальнейшего исследования системы LoRa.

7. Необходимо провести исследование возможности совместного использования спектра системой LoRa с известными системами в диапазонах 2,4 и 5 ГГц.

Литература

1. *Верхулевский К.* «Технология LoRa компании Semtech: новый импульс развития «Интернета вещей» // Беспроводные технологии 2015, № 3. С. 8-14.
2. Приказ Министерства связи и массовых коммуникаций Российской Федерации от 14 сентября 2010 г. № 124 об утверждении правил применения оборудования радиодоступа. Часть I. Правила применения оборудования радиодоступа для беспроводной передачи данных в диапазоне от 30 МГц до 66 ГГц (в ред. Приказа Минкомсвязи России от 23.04.2013 № 93).
3. *Сикарев А.А., Лебедев О.Н.* Микроэлектронные устройства формирования и обработки сложных сигналов. М.: Радио и связь, 1983. 216 с.
4. Решение ГКРЧ от 7 мая 2007 г. № 07-20-03-001.
5. *Фролов А.А.* Влияние узкополосных и широкополосных помех на многочастотную импульсную СШП-систему радиодоступа // Электросвязь. 2014. № 7. С. 32-35.
6. *Josh Blum*, "LoRa modem with LimeSDR", blog on MYRIADRF,10.6.2016, https://myriadrf.org/blog/lora-modem-limesdr/
7. LoRa-Alliance, „LoRaWAN 101 – A Technical Introduction“, https://www.lora-alliance.org/What-Is-LoRa/Technology, May 2017.
8. RC232 user manual. www.radiocrafts.com
9. SEMTECH, “LoRa Modulation Basics“, Application Note AN1200.22, May 2015.
10. Интернет ресурс: https://habrahabr.ru/company/rtl-service/blog/304312/ (Дата обращения 28.09.2017).
11. Интернет ресурс: http://lorawan.lace.io/faqs/lora/ (Дата обращения 28.09.2017).
12. Фролов А.А. Применение сверхширокополосных систем для решения проблемы дефицита РЧС // Вестник связи. 2012. № 9. С. 12-16.
13. *Косичкина Т.П., Сперанский В.С., Спирин А.П., Фролов А.А.* Когнитивные сверхширокополосные радиосистемы как метод повышение эффективности использования радиочастотного спектра // T-Comm: Телекоммуникации и транспорт. 2015. Т. 9. № 12. С. 37-43.
14. *Сперанский В.С., Фролов А.А.* Сложные дискретные частотные сверхширокополосные сигналы // REDS: Телекоммуникационные устройства и системы. 2014. Т. 4. № 1. С. 78-82.
15. *Сперанский В.С., Спирин А.П., Фролов А.А., Косичкина Т.П.* Перспективы развития сверхширокополосных систем связи в направлении когнитивного радио // Системы синхронизации, формирования и обработки сигналов. 2015. Т. 6. № 1. С. 9-11.
16. *Фролов А.А.* Сверхширокополосная система радиодоступа с совмещением многочастотной и импульсной технологией // T-Comm: Телекоммуникации и транспорт. 2013. Т. 7. № 10. С. 100-102.
17. *Фролов А.А., Шинаков Ю.С.* Исследование и разработка многочастотной сверхширокополосной системы с ДЧ сигналами // T-Comm: Телекоммуникации и транспорт. 2015. Т. 9. № 6. С. 28-33.
18. *Фролов А.А.* Анализ современных стандартов: MCWILL, TD-SCDMA, WCDMA, IEEE 802.15.3A для применения в СШП-системах // T-Comm: Телекоммуникации и транспорт. 2012. Т. 6. № 9. С. 144-148.

РАСЧЕТ ЩЕЛЕВЫХ ИЗЛУЧАТЕЛЕЙ С МНОГОСЛОЙНЫМ УКРЫТИЕМ МЕТОДОМ ИНТЕГРАЛЬНЫХ УРАВНЕНИЙ

Ястребцова Ольга Игоревна
МТУСИ, аспирант
yastrebtsova@rambler.ru

Чебышев Вадим Васильевич
МТУСИ, д.т.н., профессор кафедры ТЭДиА
tedia@mtuci.ru

Рассматривается решение задачи возбуждения щели в проводящем экране с многослойным укрытием на основе метода интегральных уравнений. Приводится построение элементов тензорной функции Грина многослойной структуры с магнитным током с использованием соотношений двойственности. В квазистатическом приближении определяется поле возбуждения щели и проводится обращение задачи к интегральному уравнению Фредгольма первого рода для полного магнитного тока щели. Приведен результат расчета диаграммы направленности полуволновой щели под слоем диэлектрика по предлагаемой методике. Проводится сравнение полученных результатов с результатами эксперимента, что позволяет сделать вывод об адекватности метода интегральных уравнений для решения рассматриваемой задачи.

Ключевые слова: *микрополосковые антенны, многослойные среды, метод интегральных уравнений, щелевые излучатели, уравнения Фредгольма*.

Малогабаритные невыступающие щелевые антенны и антенные решетки находят широкое применение в различных системах связи. Применение многослойных укрытий для щелевых излучателей существенно изменяет амплитудно-фазовые, направленные и частотные свойства последних [1, 2, 5-7].

Предлагается метод расчета щелевых излучателей с многослойным укрытием, основанный на использовании интегральных уравнений для магнитных токов, моделирующих щелевые излучатели. Метод основан на наиболее общем подходе при решении электродинамических задач для неоднородных сред, который состоит в построении интегральных представлений полей в плоских многослойных средах с использованием формализма представления векторных потенциалов поля с помощью тензорной функции Грина с последующим ее обращением к интегральному уравнению первого рода для магнитного тока.

Рассматривается возбуждение некоторым первичным полем $\left(\mathbf{E}^0,\mathbf{H}^0\right)$ узкой щели в проводящем экране (рис. 1) с покрытием в виде плоской слоисто-однородной среды (Рис. 2а). Используя зеркальное отображение, исходная задача сводится к задаче возбуждения узкой ($2b \ll \lambda, a/b > 1$) магнито-проводящей ленты $S_{пр}$ магнитным током с поверхностной плотностью $\mathbf{j}_s^M\left(M_0\right), M_0 \in S_{пр}$ слоистой среды удвоенной толщины (рис. 2б). Случай криволинейной щели может быть рассмотрен по аналогии с криволинейными полосковыми излучателями [3].

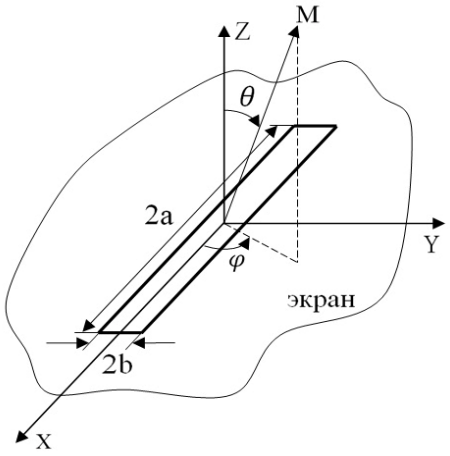


Рис. 1. Геометрия узкой щели в проводящем экране

Предполагается гармоническое изменение поля во времени по закону $\exp(i\omega t)$. Слоистая среда характеризуется параметрами $\dot{\varepsilon}_n(z), \mu_0, n = 1...N$. В пределах слоя параметры среды постоянны.

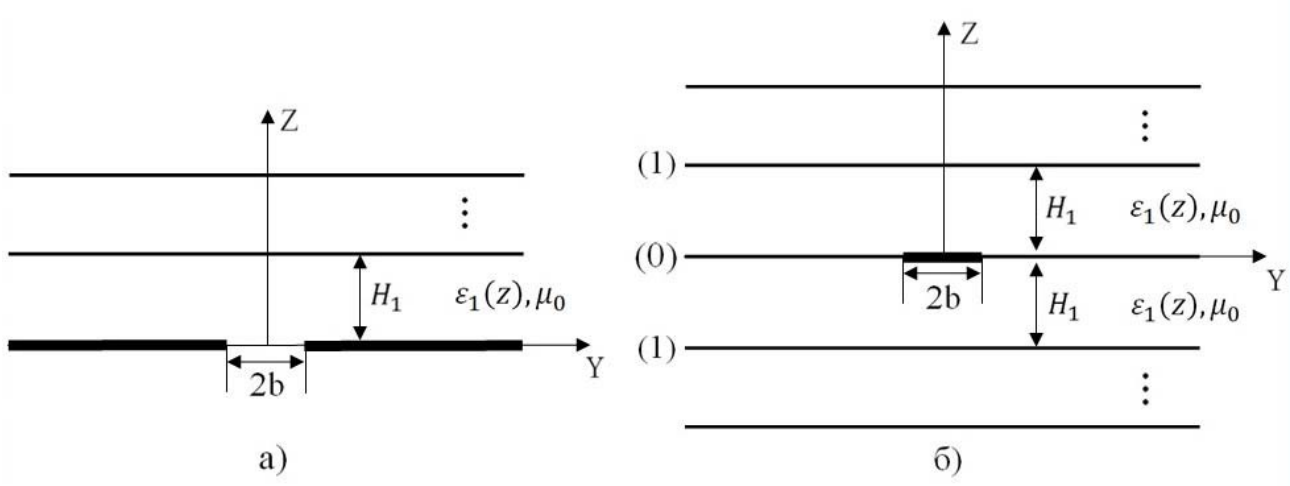


Рис. 2. Покрытие в виде плоской слоисто-неоднородной среды (а) и зеркальное отображение исходной задачи (б)

Поле $(\mathbf{E}, \mathbf{H})$, создаваемое магнитным током, будем характеризовать векторным потенциалом $\mathbf{A}^M$, который по аналогии с решением задачи для однородной среды, представим в виде:

$$\mathbf{A}^M(M) = \frac{\mu_0}{4\pi} \iint_{S_{пр}} \mathbf{j}_s^M(M_0) \hat{G}^M(M, M_0) d\sigma_{M_0}, \tag{1}$$

где M, M_0 – точки наблюдения и истока, $\hat{G}^M$ – тензорная функция слоистой среды для магнитных источников, которая в матричной форме имеет вид [4]:

$$\hat{G}^M = \begin{vmatrix} G_0^M & 0 & 0 \\ 0 & G_0^M & 0 \\ \varepsilon(z)\dfrac{\partial g^M}{\partial x} & \varepsilon(z)\dfrac{\partial g^M}{\partial y} & \dfrac{\varepsilon(z)}{\mu_0} G_1^M \end{vmatrix}, \tag{2}$$

где $G_0^M(M, M_0), g^M(M, M_0), G_1^M(M, M_0)$ – элементы тензорной функции Грина $\hat{G}^M(M, M_0)$. Поле $(\mathbf{E}, \mathbf{H})$ магнитного источника с помощью векторного потенциала $\mathbf{A}^M$ вычисляется из соотношений:

$$\mathbf{E} = -\frac{1}{\varepsilon(z)} \operatorname{rot} \mathbf{A}^M, \ \mathbf{H} = -i\omega \mathbf{A}^M - \frac{i}{\omega\mu_0} \operatorname{grad}\left(\frac{1}{\varepsilon(z)} \operatorname{div} \mathbf{A}^M \right) \tag{3}$$

Для тока линейной щели $\mathbf{j}_s^M = \mathbf{x}_0 j_x^M$

$$A_x^M(M) = \frac{\mu_0}{4\pi} \iint_{S_{пр}} j_x^M(M_0) G_0^M(M, M_0) d\sigma_{M_0}$$

$$A_z^M(M) = \frac{\mu_0}{4\pi} \iint_{S_{пр}} j_z^M(M_0) \frac{\partial g^M(M, M_0)}{\partial x} d\sigma_{M_0} \tag{4}$$

Граничное условие на ленточном магнитном проводнике $S_{пр}$ сформулируем для полного магнитного поля:

$$\left[\left(\mathbf{H} + \mathbf{H}^0 \right) \times \mathbf{z}_0 \right] = 0 \text{ на } S_{пр} \tag{5}$$

Подставив (1) и (2) в (3), получим интегро-дифференциальное уравнение для тока $j_x^M, M_0 \in S_{пр}$, которое можно представить в виде

$$\left(\frac{\partial^2}{\partial x^2}+k_1^2\right)\iint\limits_{S_{пр}} j_x^M\left(M_0\right)\left[G_0^M+\frac{\partial g^M}{\partial z}\right]d\sigma_{M_0}=-\frac{k_1}{2}\iint\limits_{S_{пр}} j_x^M\left(M_0\right)\frac{\partial g^M}{\partial z}d\sigma_{M_0}-\frac{k_1^2}{i\omega}\frac{4\pi}{\mu_0}H_x^0, \tag{6}$$

где $k_1=\omega\sqrt{\varepsilon_1\mu_0}$.

Рассматривая уравнение (6) для $M\in\Gamma$, где $\Gamma\left(x,y=0\right)$ – ось ленты $S_{пр}$, имеем обыкновенное дифференциальное уравнение второго порядка, которое в обобщенном виде,

$$\left(\frac{d^2}{dx^2}+k_1^2\right)A_\Gamma=-F$$

Используя известную формулу обращения для выделенного дифференциального оператора, решение этого уравнения имеет вид:

$$A_\Gamma=\int\limits_\Gamma F\left(x'\right)G\left(x,x'\right)dx'+C_1\psi_1\left(x\right)+C_2\psi_2\left(x\right)$$

где $G\left(x,x'\right)$ – функция Грина обращения. Для системы функций $\psi_1=\sin k_1x$, $\psi_2=\cos k_1x$ функция Грина будет иметь вид:

$$G\left(x,x'\right)=\frac{\sin k_1\left|x-x'\right|}{2k_1}$$

В результате из (6) получим интегральное уравнение:

$$\iint\limits_{S_{пр}} j_x^M\left(M_0\right)K\left(M,M_0\right)d\sigma_{M_0}=$$

$$=-i\,2\pi W_1\int\limits_\Gamma H_x^0\left(u\right)\sin k\left|u-x\right|dx+C_1\sin k_1x+C_2\cos k_2x, \tag{7}$$

где $W_1=\sqrt{\mu_0/\varepsilon_1}$, ε_1 – параметр первого слоя покрытия щели. В (7) ядро уравнения имеет вид:

$$K\left(M,M_0\right)=G_0^M\left(M,M_0\right)+\frac{\partial g^M\left(M,M_0\right)}{\partial z}-\frac{1}{2k}\int\limits_\Gamma\sin k_1\left|u-x\right|\frac{\partial g^M\left(M,M_0\right)}{\partial z}dx \tag{8}$$

Узкую щель в проводящем экране можно характеризовать полным магнитным током I^M, который для поля E_{ξ_1} щели определим как

$$I^M=2\int\limits_{-b}^{b}E_{\xi_1}\left(\xi,x\right)d\xi \tag{9}$$

Соотношение (9) для полного тока I^M щели является интегральной характеристикой поверхностного тока $j_s^M\left(M_0\right),M_0\in S_{пр}$. Учитывая особенность этого тока на ребре, имеем,

$$j_s^M\left(M_0\right)=\frac{I^M\left(\Gamma\right)}{\sqrt{b^2-y^2}},\ M_0\in S_{пр},\Gamma\in\left(0,x\right) \tag{10}$$

где $I^M\left(\Gamma\right)$ – полный магнитный ток щели.

Определим элементы G_0^M и $\frac{\partial g^M}{\partial x}$, определяющие ядро (8) интегрального уравнения (7) для магнитного тока. Эти элементы, как следует из [3], можно определить по известным элементам тензора Грина для электрического источника, если использовать соотношения двойственности $G_0^M\to G_1^Э,G_0^Э\to G_1^M$ при замене $\varepsilon\to\mu$, $R^H\to -R^E$, $W^E\to W^H$.

В результате из (5) для однослойной структуры (Рис. 2б), выделяя дипольную особенность элементов тензора, при $z_0=0,0\le z\le H_1$ имеем

$$G_0^M=\frac{e^{-ik_1R}}{R}+\int\limits_0^\infty f_G\left(\lambda\right)\frac{\lambda}{\eta_1}J_0\left(\lambda\rho\right)d\lambda \tag{11}$$

где $R(M,M_0)=\sqrt{\rho^2+z^2}$, $\rho=\sqrt{(x-x_0)^2+(y-y_0)^2}$, $\eta_1=\sqrt{\lambda^2-k_1^2}$.

$$f_G(\lambda)=-R_0^E\frac{1-R_1^E e^{-2\eta_1 H_1}}{1+R_1^E R_0^E e^{-2\eta_1 H_1}}-R_1^E e^{-2\eta_1 H_1}\frac{1-R_0^E}{1+R_1^E R_0^E e^{-2\eta_1 H_1}}$$

$$\left.\frac{\partial g^M(R)}{\partial z}\right|_{z=0}=G_0^M(R)+\int_0^\infty f_g(\lambda)J_0(\lambda\rho)\,d\lambda, \qquad (12)$$

где $f_g(\lambda)=\dfrac{1-R_0^E-R_1^E(1-R_0^E)e^{-2\eta_1 H_1}}{1-R_1^E R_0^E e^{-2\eta_1 H_1}}\dfrac{\lambda}{\eta_1}+\dfrac{\lambda}{\eta_1}\dfrac{R_0^E-R_1^E(1-R_0^E)e^{-2\eta_1 H_1}}{1-R_1^E R_0^E e^{-2\eta_1 H_1}}-\dfrac{\lambda}{\eta_1}\dfrac{R_0^H-R_1^H(1-R_0^H)e^{-2\eta_1 H_1}}{1-R_1^H R_0^H e^{-2\eta_1 H_1}}$

Коэффициенты отражения от слоев (0), (1) $R_0^{E,H}$ и $R_1^{E,H}$ (Рис. 2б) определяются в (12) и (13) по методике [4].

При численном интегрировании в (11), (12) предполагается ограничение верхнего предела в несобственном интеграле в силу его быстрой сходимости. Представление элементов (11), (12) с выделенной особенностью определяет интегральное уравнение (7) с ядром (8) как интегральное уравнение Фредгольма первого рода для магнитного тока j_s^M щелевого излучателя в слоистой среде.

Используем представление (10) для полного магнитного тока щелевого излучателя. Тогда при вычислении кратного интеграла в уравнении (7), используя квадратурную формулу Эйлера с одним узлом [4], можно получить одномерное интегральное уравнение для полного тока щели I^M,

$$\int_\Gamma I^M(x_0)G(x,x_0)\,dx_0=-i\frac{U}{\pi k_1 b}\sin k_1|x|+C_1\sin k_1 x+C_2\sin k_1 x, \qquad (13)$$

где ядро уравнения из (8), полагая $\xi=|x-x_0|$, имеет вид

$$G(\xi)=G_0(\xi)+\left.\frac{\partial g(\xi,z)}{\partial z}\right|_{z=0}-\frac{1}{2}\int_\Gamma \sin|x-u|\left.\frac{\partial g(|u-x_0|,z)}{\partial z}\right|_{z=0}du \qquad (14)$$

$$G(\xi)=\frac{e^{-i\xi}}{\sqrt{\xi^2+d^2}}\left[(1+i\xi)\frac{2}{\pi}F\left(\frac{\pi}{2},\alpha\right)-i\sqrt{\xi^2+d^2}\right]+\int_\Gamma f_G(\lambda)\frac{\lambda}{\eta_1}J_0(\lambda\xi)\,d\lambda,$$

где $F\left(\dfrac{\pi}{2},\alpha\right)$ – полный эллиптический интеграл первого рода [4], $\alpha=\dfrac{d}{\sqrt{\xi^2+d^2}}$. Остальные функции можно получить заменой ρ на ξ. При $\xi\ll 1$ имеем представление

$$F\left(\frac{\pi}{2},\alpha\right)\cong\ln 4-\ln\frac{\xi}{\sqrt{\xi^2+d^2}}+0(\xi^2)$$

Следовательно, ядро (14) в уравнении (13) имеет – логарифмическую (слабую) особенность, так что это уравнение является одномерным интегральным уравнением Фредгольма первого рода. Коэффициенты C_1 и C_2 в (13) определяются из дополнительного условия отсутствия «стекания» магнитного тока с концов щели.

Для численного решения интегрального уравнения (13) наиболее приспособлен принцип саморегуляризации [3], реализующий особенность ядра (14) уравнения. Метод состоит в дискретизации уравнения (13) методом коллокаций и сведении уравнения к хорошо обусловленной системе линейных алгебраических уравнений (СЛАУ) для значений тока в узлах дискретизации с последующим ее решением каким-либо методом вычислительной математики. Наиболее экономичный алгоритм численного решения интегрального уравнения предполагает кусочно-квадратичную аппроксимацию тока на шаге дискретизации. Алгоритм численного решения приведен в [3]. По результатам вычисления тока щели при заданном потенциале U на ее входе можно определить ее характеристики.

Определим поле излучения и диаграмму направленности щели с покрытием. Поле линейной щели характеризуется составляющими векторного потенциала $\left(A_x^M, A_z^M\right)$ (4), где элементы тензора G_0^M и $\dfrac{\partial g^M}{\partial x}$ из [3] представим в виде,

$$G_0^M\left(M,M_0\right)=\frac{1}{2}\int_{-\infty}^{\infty}g_0^M\left(\lambda,z_0\right)e^{-\eta_0 z}H_0^{(2)}\left(\lambda\rho_{MM_0}\right)\lambda d\lambda$$

$$\frac{\partial g^M}{\partial x}\left(M,M_0\right)=\frac{1}{2}\int_{-\infty}^{\infty}g_1^M\left(\lambda,z_0\right)e^{-\eta_0 z}H_0^{(2)}\left(\lambda\rho_{MM_0}\right)\lambda d\lambda \qquad (15)$$

где $H_0^{(2)}\left(\lambda\rho\right)$ – функции Ханкеля.

В свою очередь, для функций $g_0^M\left(\lambda,z_0\right)$, $g_1^M\left(\lambda,z_0\right)$ в (11), зависящих от вида слоистой среды, при $0\le z_0<H_1, z>H_1$ можно получить из [3].

Несобственные интегралы в (11) вычисляются в приближении дальней зоны щели, и при $\lambda=k\sin\theta$ получим $G_0^M\cong g_0\left(\theta\right)\frac{\mathrm{e}^{-ikR}}{R}$, $\frac{\partial g^M}{\partial x}\cong-\sin\varphi\, g_1\left(\theta\right)\frac{\cos\theta}{\sin\theta}\frac{\mathrm{e}^{-ikR}}{R}$, $R\to\infty$.

Используем известные соотношения для составляющих поля магнитного вибратора в дальней зоне и получим

$$H_\theta=B\cos\theta\cos\varphi\left[\cos\varphi g_0\left(\theta\right)+ig_1\left(\theta\right)\right]\int_{-L}^{L}I^M\left(x_0\right)e^{-ikx_0\sin\theta\cos\varphi}dx_0$$

$$H_\varphi=iB\cos\theta\sin\varphi g_1\left(\theta\right)\int_{-L}^{L}I^M\left(x_0\right)e^{-ikx_0\sin\theta\cos\varphi}dx_0,$$

где B – нормирующий коэффициент.

На рисунке 3 приведен результат расчета диаграммы направленности полуволновой линейной щели с покрытием в виде слоя с $\varepsilon_{r1}=2.8$ и $H_1=0{,}1\lambda$ по указанной методике, в сравнении с результатом эксперимента на частоте 2.7 ГГц и с расчетом диаграммы на основе HFSS. Отметим существенное отличие в расчете диаграммы направленности щели, рассчитанной на основе прикладного пакета HFSS, от диаграммы, рассчитанной по предлагаемой методике.

Рис. 3. Диаграммы направленности полуволновой линейной щели с диэлектрическим покрытием в плоскости H

Таким образом, в работе была рассмотрена задача возбуждения линейной щели в экране с многослойным укрытием. Для этого было проведено построение тензорной функции Грина для многослойной среды, имеющей вид зеркального отображения укрытия щели в экране, и определены соотношения двойственности для расчета элементов тензора. Далее было проведено обращение задачи к одномерному интегральному уравнению Фредгольма первого рода для полного магнитного тока щели. Приведенные численные результаты свидетельствуют о различии расчета диаграммы направленности щелевого излучателя по предлагаемой методике с расчетом диаграммы направленности на основе пакета прикладных программ HFSS.

Литература

1. *Magill E., Wheeler H.* Wide-angle impedance matching of a planar array antenna by a dielectric sheet. IEEE Transactions on Antennas and Propagation. vol. 4. iss. 1. 1996, pp. 49-53.
2. *Красюк В.Н*. Антенны СВЧ с диэлектрическими покрытиями. Ленинград: Судостроение, 1986. 164 с.
3. *Чебышев В.В.* Основы проектирования антенных систем. М.: Горячая линия – Телеком. 2016. 149 с.
4. *Дмитриев В.И., Захаров Е.В.* Метод интегральных уравнений в вычислительной электродинамике. М.: Макс Пресс. 2008. 307 с.
5. *Ястребцова О.И., Чебышев В.В*. Электродинамический анализ волн, распространяющихся в многослойных средах микрополосковых антенн // T-Comm: Телекоммуникации и транспорт. 2016. Т. 10. № 8. С. 3-8.
6. *Чебышев В.В., Ястребцова О.И*. Свойства микрополоскового вибратора в многослойной среде из метаматериалов // T-Comm: Телекоммуникации и транспорт. 2015. Т. 9. № 7. С. 47-52.
7. *Чебышев В.В., Лисицына Ю.А.* Частотные свойства микрополосковых вибраторов со слоистой подложкой // T-Comm: Телекоммуникации и транспорт. 2012. Т. 6. № 10. С. 123-125.

ПОЛЬЗОВАТЕЛЬСКИЕ АСПЕКТЫ БЕЗОПАСНОСТИ В СЕТЯХ LTE

Корионов Игорь Павлович
студент группы БСУ1401 МТУСИ
tyiiia_abr@mail.ru

Орлов Владимир Георгиевич
МТУСИ, к.т.н., доцент кафедры ТиЗВ
ovg250846@icloud.com

Представлены характеристики современных радиотехнологий мобильного доступа. Рассмотрены механизмы реализации взаимной аутентификации и процедура АК в LTE. Приведены процедуры и алгоритмы формирования вектора аутентификации. Описаны схемы получения ключей шифрации и целостности информации для пользовательских пакетов и сигнализации. Приведены алгоритмы проверки целостности сообщений и шифрации данных.

Ключевые слова: *безопасность в сетях LTE, аутентификация, вектор аутентификации, генерация ключей безопасности и целостности, алгоритмы шифрования/дешифрования, контроль целостности сообщений.*

На сегодняшний день LTE является наиболее перспективной и быстро развивающейся технологией мобильного широкополосного доступа. Интенсивное распространение LTE и постепенный переход от сетей 2G/3G свидетельствуют о том, что данная технология будет повсеместно использоваться в беспроводных системах для передачи данных, включая так называемый «тяжёлый контент», услуги IP-TV с HD качеством, on-line сервисы с IoT и M2M приложениями, а также для мультимедийного широкополосного вещания eMBMS с трансляцией ТВ-контента в режиме реального времени (LTE-Advanced) [6, 7]. С точки зрения использования такого дефицитного ресурса, как радиочастотный спектр, а также по спектральной эффективности LTE существенно превосходит наиболее скоростной вариант UMTS – HSPA+, (табл. 1) [1].

Таблица 1

Характеристики современных радиотехнологий мобильного доступа

Характеристики	WCDMA (UMTS)	HSPA	HSPA+	LTE	LTE-Andvanced (ITM-Advanced)
Максимальная скорость на линии вниз	384 Кб/с	14 Мб/с	28 Мб/с	100 Мб/с	1 Гб/с
Максимальная скорость на линии вверх	128 Кб/с	5,7 Мб/с	11 Мб/с	50 Мб/с	500 Мб/с
Задержка (полный круг)	150 мсек	100 мсек	50 мсек	10 мсек	Менее 5 мсек
Релиз 3GPP	99/4	5/6	7	8	10
Год начала развертывания	2003/4	2005/8	2008/9	2009/10	2011/12
Метод одновременного доступа	CDMA	CDMA	CDMA	OFDMA/SC-FDMA	OFDMA/SC-FDMA

Особенностью LTE является то, что в отличие от стандарта мобильной связи 3G, в сетях LTE весь трафик проходит не по двум разным сетям (голосовой по сети с коммутацией каналов через MSC, а данные - по сети данных через узлы маршрутизации данных и обслуживания абонентов GGSN/SGSN), а через единую архитектуру EPC (Evolved Packet Core) по протоколу IP, что влечёт за собой все угрозы связанные с IP. Открытая, основанная на IP, распределенная архитектура LTE позволяет злоумышленникам использовать мобильные устройства и сети со спамом для организации кражи данных и прослушивания пользователей, IP-спуфинга, распространения вредоносного ПО, атак DDoS и многих других вариантов кибератак и преступлений. С учётом этого архитектура LTE была спроектирована и разработана 3GPP с учетом принципов безопасности, основывающихся на пяти группах функций безопасности [5]:

1. Безопасность доступа к сети – обеспечивается безопасный доступ UE к EPC и защита от возможных атак в радиоканале посредством защиты целостности и шифрования между USIM, ME, E-UTRAN и объектами EPC.

2. Безопасность сетевого домена – защита сетевых элементов опорной сети от возможных атак и защита обмена сигнальными и пользовательскими данным.

3. Безопасность пользовательского домена для управления доступом к мобильным станциям. Обеспечивается совместной аутентификацией USIM и ME, прежде чем они смогут получить доступ друг к другу, с использованием секретного PIN-кода.

4. Безопасность домена приложения – набор функций безопасности, которые позволяют приложению в UE и поставщику услуг установить безопасную связь на уровне приложения. Прозрачность и конфигурация безопасности позволяют пользователю контролировать наличие функций безопасности в операциях обмена данными.

В целом стандарт LTE включает мощные криптографические методы взаимной аутентификации между сетевыми элементами с механизмами безопасности, встроенными в его архитектуру [5].

Взаимная аутентификация и соглашение о ключах
(AKA – Authentication and Key Agreement)

Процедура запускается сетью при подключении абонентского терминала (UE) к сети для взаимной аутентификации абонента и сети, и формирования промежуточного ключа $K_{ASME.}$ UE идентифицируется посредством IMSI (IMSI и K хранятся как в HSS, так и на SIM карте UE), после чего MME, получив запрос, обращается к HSS за информацией для аутентификации (Auth Data Request), передавая при этом IMSI, SNID, Network Type. HSS определив K, соответствующий полученному IMSI, формирует с помощью односторонних функций f1 – f5 параметры вектора аутентификации AV (рис. 1), где:

- MAC (Message Authentication Code), используется UE при контроле подлинности обслуживающей сети;
- KASME (Key Access Management Entity), генерируется HSS с помощью алгоритма KDF (Key Derivation Function);
- XRES (Expected Response), используется MME при установлении подлинности UE;
- CK (Ciphering Key) – ключ закрытия информации;
- IK (Integrity Key) – ключ целостности информации;
- AK (Anonymity Key), используется для шифрации SQN;
- AMF (Authentication Management Field), используется для указания типа сети;
- SQN (Squence Number) – счетчик процедур AKA, предотвращающий повторное использование вектора аутентификации AV;
- IMSI – Международный идентификатор UE (SIM карты);
- K – абонентский ключ;
- SNID (Serving Network Identity) – идентификатор обслуживающей сети,

равный PLMN ID данной сети.

В результате определяется вектор аутентификации AV = {RAND, XRES, AUTN, KASME} и параметр аутентификации сети AUTN = {SQN + AK, AMF, MAC}, (рис. 1) [5].

Сформированный вектор аутентификации AV (Auth Data Response) передаётся из HSS по запросу в MME, где для UE с номером IMSI формируется сообщение Auth Request, содержащее RAND и AUTN, которое MME передаёт по эфиру UE.

Получив вектор аутентификации AV для UE с номером IMSI, MME формирует сообщение Auth Request, содержащее RAND и AUTN, и предает его UE. Получив данное сообщение, UE обеспечивает формирование данных, необходимых для аутентификации, используя односторонние функции f1 – f5, такие же, как в HSS, (рис. 2) [2].

На основании сформированной информации UE идентифицирует сеть путём сравнения xMAC с полученным MAC. В случае их отличия UE передает MME сообщение (Auth Failure) с указанием причины отсутствия аутентификации. Равенство xMAC = MAC означает, что сеть успешно аутентифицирована, после чего UE отправляет MME сообщение (Auth Response), содержащее RES [3].

Обмен информацией между UE и MME осуществляется в соответствии с протоколом NAS. Получив данное сообщение от UE, MME производит аутентификацию UE, сравнивая RES с ранее полученным от HSS xRES. Если они не равны, MME направляет UE сообщение (Auth Reject) об отрицательном результате аутентификации. При равенстве xRES = RES процесс взаимной аутентификации UE и MME считается успешным. Следует отметить, что обмен информацией между UE и MME осуществляется в соответствии с протоколом NAS.

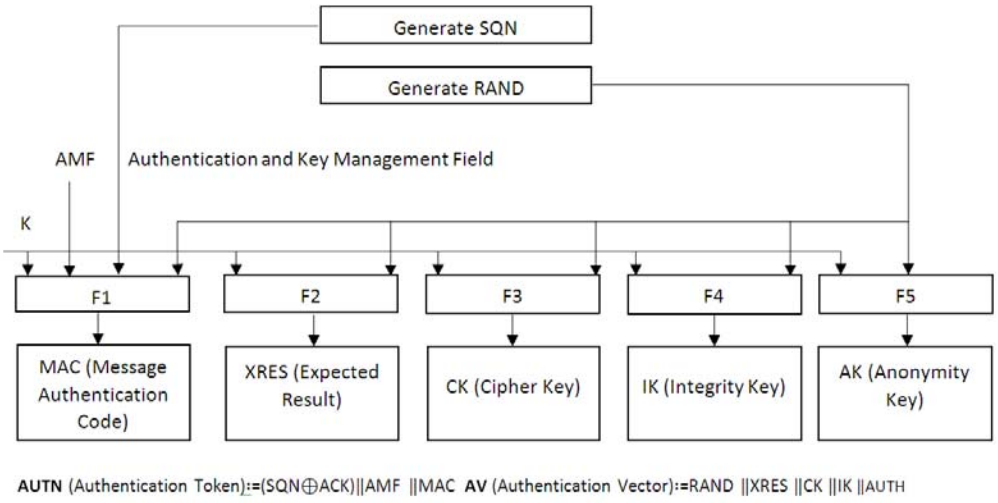


Рис. 1. Формирование AV и AUTN в HSS

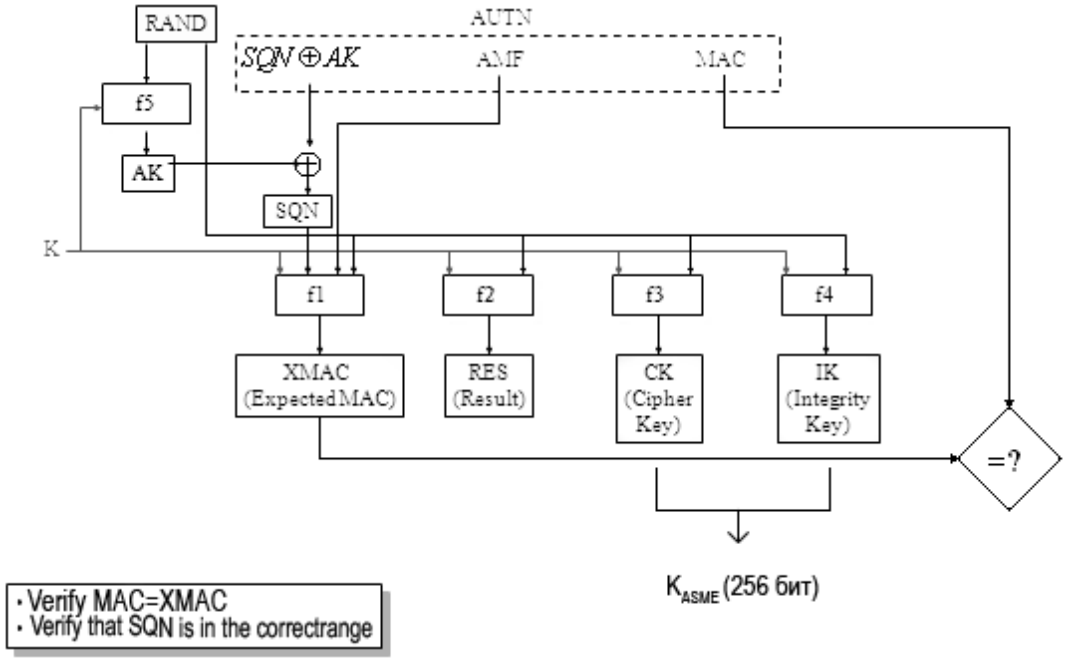


Рис. 2. Преобразование информации аутентификации в UE и идентификация сети

Шифрование и контроль целостности данных

При успешном завершении процедуры аутентификации и АКА MME, eNB и UE производят генерацию ключей, используемых для шифрации и проверки целостности получаемых сообщений (рис. 3).

Процедуры безопасности реализуются, как в плоскости управления, так и в пользовательской плоскости (рис. 4). В радиоканале защита обеспечивается, как для сигнального трафика, так и для пользовательских пакетов. При этом данные сигнализации разделяют на сквозные сигнальные сообщения между UE и MME (NAS – Non Access Stratum) и сигнальные сообщения между eNB протокола RRC (AS – Access Stratum), (рис. 4).

Для шифрации и защиты целостности предусматривается использование следующих базовых алгоритмов, [5, 8, 9]:

- UEA2 (UMTS Encryption Algorithm и UIA2 (UMTS Integrity Algorithm), разработанные для стандартов 3G;
- AES (Advanced Encryption Standard).

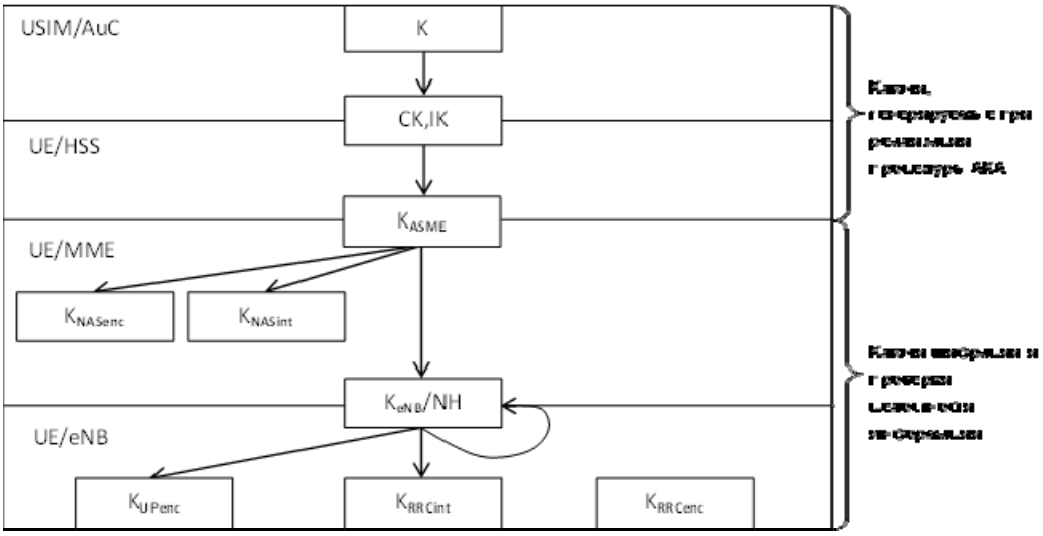


Рис. 3. Иерархия ключей

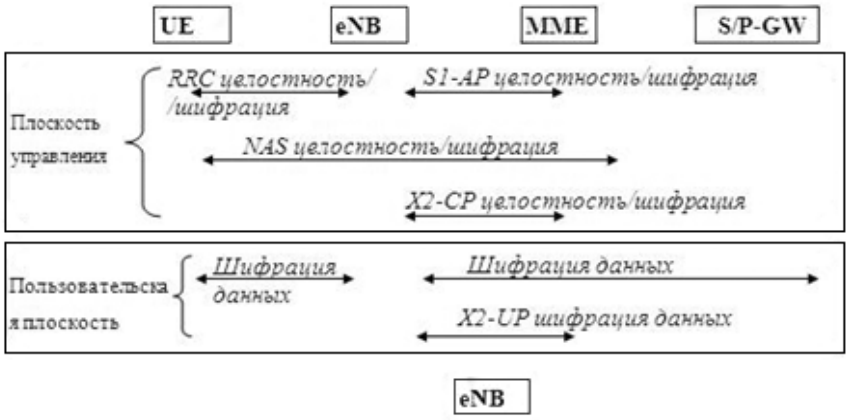


Рис. 4. Реализация процедур безопасности

Исходным ключом для всей цепочки процедур формирования ключей является K_{ASME} (рис. 6). KDF генерирует ключи длиной 256 бит, у которых с помощью функции Trunc исключают 128 старших бит, в результате чего получают рабочие ключи длиной 128 бит (рис. 6а). При этом формирование NAS ключей в UE и MME выполняется параллельно, однако UE начинает генерацию ключей по команде MME после выбора им алгоритма (рис. 6). Идентификатор алгоритма (Alg ID) и индикатор ключа K_{ASME} KSI (Key Set Identifier) содержатся в команде MME (NAS Security Mode Command). UE информирует MME о завершении генерации NAS ключей путём передачи сообщение NAS Security Mode Complete, после получения, которого UE и MME будут производить обмен зашифрованной информацией с проверкой ее целостности.

Шифрование и проверки целостности информации, передаваемой между eNB – UE (рис. 4) выполняются с помощью протокола RRC [5]. Схема получения ключей шифрации и целостности для трафика отличается от схемы генерации ключей для NAS тем, что в качестве исходного параметра служит вторичный промежуточный ключ K_{eNB} (256 бит). Этот ключ генерируется UE и MME и также использует KDF. Исходными входными параметрами для его генерации служат: K_{ASME}, счетчик сигнальных сообщений NAS вверх, прежнее значение K_{eNB}, идентификатор соты и номер частотного канала в направлении вверх. По завершении генерации K_{eNB} его значение из MME передается в eNB. Таким образом, при каждой периодической локализации UE происходит изменение K_{eNB} [4].

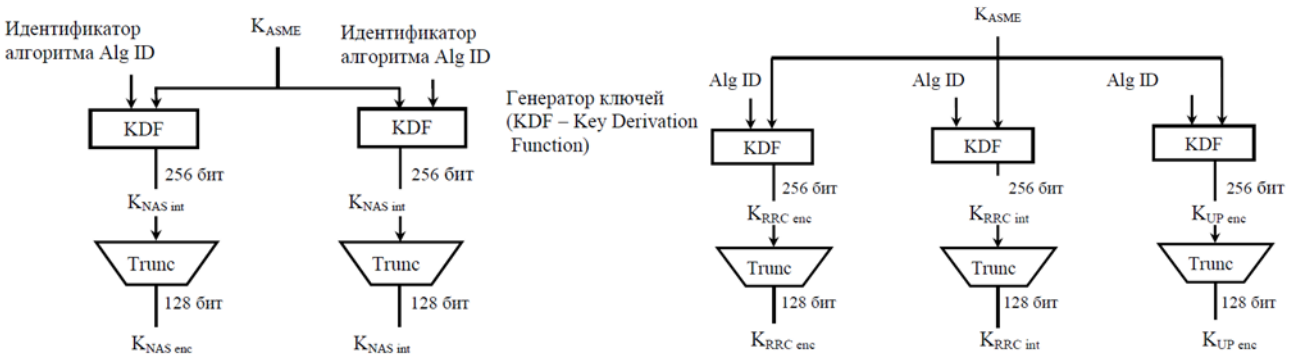


Рис. 6. Формирование ключей шифрования и целостности для NAS сигнализации (а), и для UE – eNB (б)

На рисунке 7 представлен процесс шифрации и дешифрации сообщений с использованием алгоритма EEA (EPS Encryption Algorithm).

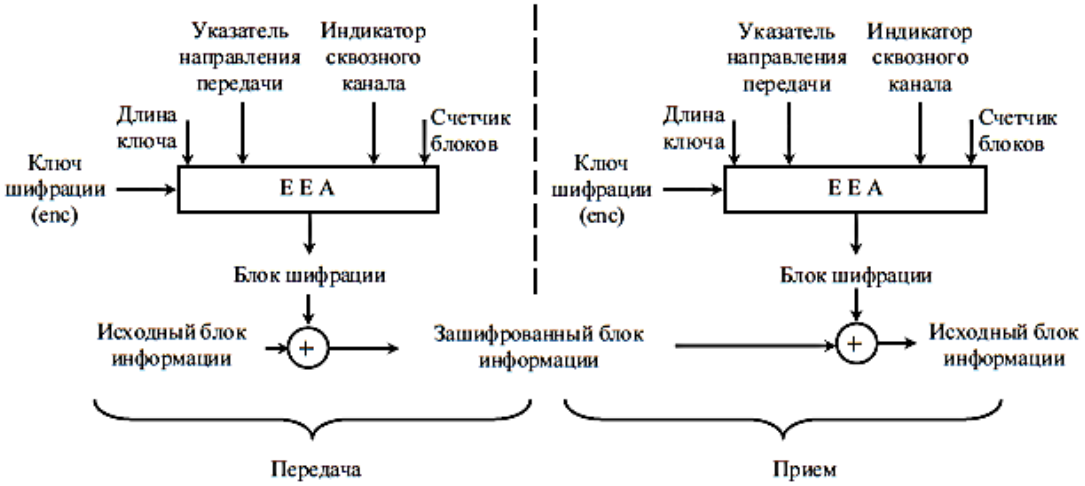


Рис. 7. Алгоритм шифрации в E-UTRAN

Исходными параметрами в этом процессе являются ключ шифрации (128 бит), счетчик блоков (32 бита), идентификатор сквозного канала (5 бит), указатель направления передачи (1 бит) и длина шифрующего ключа LENGTH. В соответствии с алгоритмом шифрации EEA формируется шифрующее число KEYSTREAM BLOCK, которое при передаче складывают по модулю два с шифруемым исходным блоком информации PLAINTEXT BLOCK. При приёме для дешифрации повторно выполняется та же операция.

Проверка целостности передаваемых сообщений (рис. 8) базируется на генерации и присоединении к передаваемому пакету числа MAC (Message Authentication Code) (32 бита), которое вычисляется на передающей стороне. На приёмной стороне с помощью того же алгоритма EIA (EPC Electronic Industries Alliance) вычисляется xMAC. Вычисление MAC и xMAC производится при одних и тех же исходных данных и при сравнении их равенство свидетельствует о целостности полученного пакета.

Рис. 8. Алгоритм проверки целостности в E-UTRAN

Заключение

Таким образом, криптостойкое шифрование в сетях LTE исключает возможность доступа злоумышленников к пользовательской информации и данным сигнализации, как в радиоканале, так и в сетевой зоне. Проверка целостности обнаруживает любую попытку злоумышленника ретранслировать или изменить сигнальные сообщения и защищает систему от атаки типа «man in the middle», когда злоумышленник, перехватывая последовательность сообщений сигнализации, модифицирует и повторно передает их, пытаясь получить доступ к информации абонентов сети.

Литература

1. *М.С. Лохвицкий, Н.С. Мардер*. Сотовая связь: от поколения к поколению. М.: Издательство ИКАР, 2014. 236 с.
2. 3GPP TS 36.101 v10.0.0; User Equipment (UE) radio transmission and reception.
3. 3GPP TS 36.321 v9.3.0; Medium Access Control (MAC) protocol specification.
4. 3GPP TS 24.301 v10.0.0; Non-Access-Stratum (NAS) protocol for Evolved Packet System (EPS); Stage 3.
5. 3GPP TS 33.401 v9.4.0; 3GPP System Architecture Evolution (SAE); Security architecture.
6. *Пушкарев А.В., Орлов В.Г.* Эволюция технических средств формирования и доставки ТВЧ на мобильные терминалы пользователей // T-Comm: Телекоммуникации и транспорт. 2015. Т. 9. № 1. С. 11-16.
7. *Орлов В.Г., Пушкарев А.В.* Перспективы развития мобильного видео // T-Comm: Телекоммуникации и транспорт. 2013. Т. 7. № 9. С. 115-117.
8. *Мазуркевич Д.О., Орлов В.Г.* Эволюция систем безопасности сетей сотовой связи разных поколений // T-Comm: Телекоммуникации и транспорт. 2011. Т. 5. № 1. С. 38-40.
9. *Орлов В.Г., Мазуркевич Д.О.* Алгоритмы шифрования в публичных беспроводных сетях // T-Comm: Телекоммуникации и транспорт. 2011. Т. 5. № 10. С. 62-64.

МОДЕЛИРОВАНИЕ РАДИОСИСТЕМЫ С МНОГОПОЛЯРИЗАЦИОННЫМ УПЛОТНЕНИЕМ

Редькович Владислав Сергеевич
аспирант МТУСИ кафедры ЭМСиУРЧС
225_vlad@mail.ru

Пустовойтов Евгений Леонтьевич
МТУСИ, к.т.н., доцент кафедры ЭМСиУРЧС
pustovoitov@niir.ru

Целью данной статьи является представление результатов проверки возможности предложенного в [1, 2] многополяризационного уплотнения радиочастотного спектра (РЧС) с разделением на приемной стороне более двух радиосигналов с одинаковой несущей частотой, но имеющих взаимно неортогональные поляризации. Упомянутая проверка проведена на примере моделирования радиосистемы с четырехкратным уплотнением, которое достигается за счет использования четырех индивидуальных поляризаций для каждого из четырех полезных радиосигналов, передаваемых из четырех разных точек в одну. При этом предполагалось, что на прием каждого из них не воздействуют внесистемные мешающие радиосигналы, а поляризации и уровни полезных радиосигналов не изменяются в процессе их передачи. Разумеется, такое идеализированное положение практически возможно лишь в космическом пространстве или искусственно созданной среде распространения радиосигналов. Однако, с учетом крайней новизны рассматриваемого подхода к значительному повышению эффективности использования РЧС радиосистемами проведение проверки реализуемости поляризационного разделения на приемной стороне более двух радиосигналов с одинаковой несущей частотой, но имеющих неортогональные поляризации, вполне оправдано.

Ключевые слова: *поляризация, уплотнение, радиочастотный спектр, канал, многополяризационное.*

Моделирование работы радиосистемы с многополяризационным уплотнением РЧС произведено с помощью программного пакета Matlab применительно к радиосистеме, структурная схема которой приведена на рис. 1. Эта радиосистема включает в себя:

– четыре радиопередатчика, передающих одно из четырех заранее подготовленных сообщений;

– высокочастотные тракты (ВЧ тракты) четырех приемников (ПРМ), каждый из которых настроен на прием радиосигнала определенной поляризации;

– устройство обработки совокупности суммарных радиосигналов, поступающих на входы каждого из приемников, которое выделяет полезные радиосигналы для каждого из приемников**.**

Основой математической модели рассматриваемой системы радиосвязи в соответствии с рис. 1 являются четыре линии радиосвязи между пунктами А и С, а также устройство обработки совокупности четырех суммарных сигналов, поступающих с выходов линейных частей ВЧ трактов каждого из приемников.

В пункте А расположены четыре передатчика, работающие на одной и той же частоте (f_0) и различающиеся поляризациями (φ) с шагом 45 градусов: 0, 45, 90, 135 градусов. В пункте С расположены четыре приемника, каждый из которых рассчитан на прием радиосигналов с одной из четырех используемых поляризаций, а также устройство обработки суммы результирующих радиосигналов Usum(t)i, поступающих из высокочастотных трактов каждого из четырех приемников, в котором производится разделение радиосигналов Ui(t) от каждого передатчика.

Моделирование радиосистемы с многополяризационным уплотнением производилось следующим образом.

Рассматривался случай аналоговой радиосистемы с фазовой модуляцией. Информационные сигналы, поступающие на входы четырех передатчиков в пункте А, представляли собой заранее записанные голосовые сообщения, осциллограммы которых изображены на рис.2 . Моделировалась фазовая модуляция этими информационными сигналами несущего колебания частотой 80 МГц с помощью специальной встроенной функции фазовой модуляции Matlab-pmmod. В итоге на частоте 80 МГц получили четыре фазомодулированных радиосигнала:

```
U1 = pmmod(x1(1,:),fc,fs,phasedev);
U2 = pmmod(x2(2,:),fc,fs,phasedev);
U3 = pmmod(x3(3,:),fc,fs,phasedev);
U4 = pmmod(x4(4,:),fc,fs,phasedev).
```

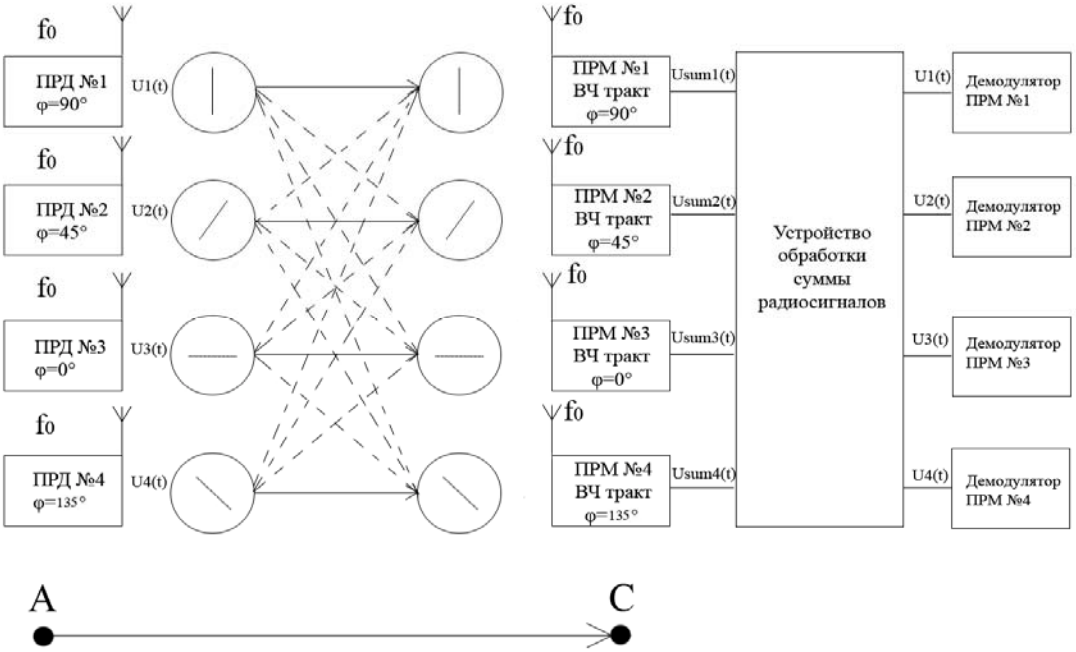


Рис. 1. Структурная схема рассматриваемой системы радиосвязи

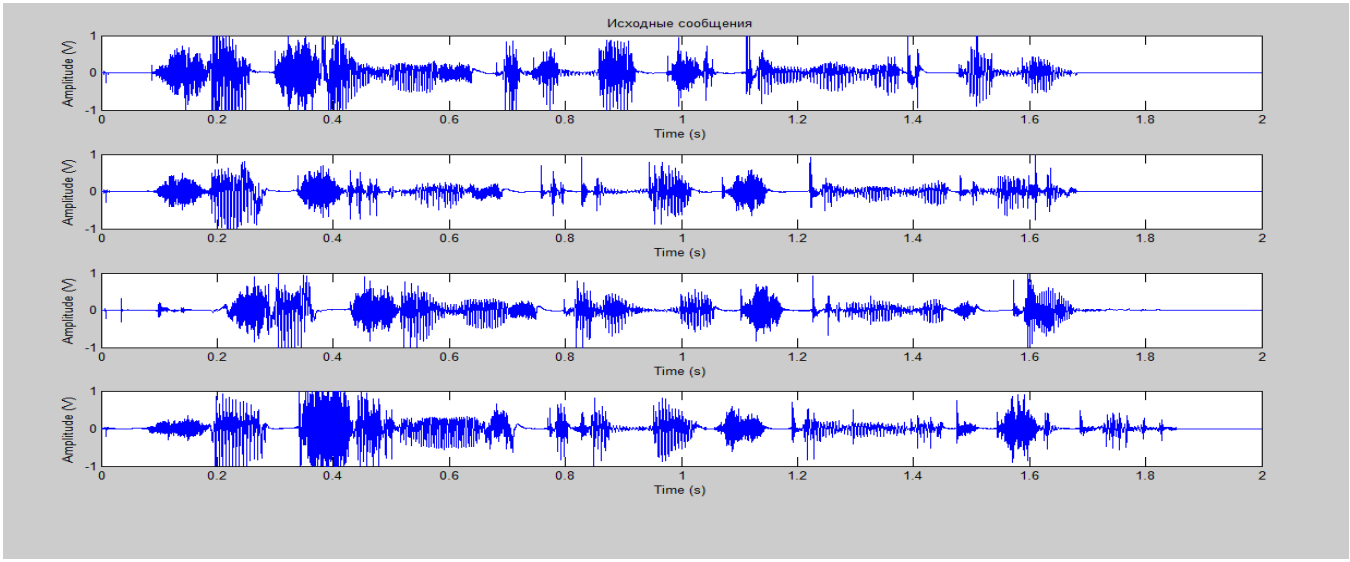


Рис. 2. Информационные звуковые сигналы на входах четырех передатчиков

Далее моделировалось поступление каждого из этих радиосигналов в индивидуальный радиоканал передачи с одной из четырех используемых поляризаций, после чего анализировалось их совместное воздействие на четыре входа устройства обработки суммы радиосигналов, предназначенного для выделения на каждом из своих выходов одного их 4-х переданных радиосигналов. Такое моделирование основано на следующих соображениях.

Проанализируем воздействие четырех радиосигналов с разными поляризациями на приемник одной из них, которую примем за нулевую. При этом будем считать радиосигнал с “нулевой” поляризацией (φ =0) полезным сигналом (ПС), а остальные – мешающими сигналами (МС) по отношению к ПС. На рис.3 показаны положения плоскостей поляризаций всех радиосигналов. При этом на вход приемника ПС поступит сумма следующих сигналов:

$$U_{ПС} = A_1 \cdot \cos\left(\omega_0 t + \Omega\psi_1(t)\right)$$

$$U_{МС1} = A_2 \cdot \cos\left(\omega_0 t + \Omega\psi_2(t)\right) \cdot \cos\varphi_1$$

$$U_{МС2} = A_3 \cdot \cos\left(\omega_0 t + \Omega\psi_3(t)\right) \cdot \cos\varphi_2$$

$$U_{МСi} = A_i \cdot \cos\left(\omega_0 t + \Omega\psi_i(t)\right) \cdot \cos\varphi_i,$$

где: $U_{ПС}$ – полезный сигнал; $U_{МСi}$ – i-й мешающий сигнал; A_i – амплитуда i – го сигнала; ω_0 – несущая частота сигнала; $\Omega\psi_i(t)$ – фазовая модуляция i – го сигнала; φ_N – разность углов между плоскостями поляризации N – го МС и ПС

Аналогичным образом можно описать сумму полезного и мешающих радиосигналов для приемников сигналов других поляризаций. Учитывая суммы радиосигналов на входе каждого из 4-х приемников, получим систему уравнений (1), заменив для краткости записи $A_n \cdot \cos\left(\omega_0 t + \Omega_n(t)\right) \cdot \cos\varphi_n$ на $U_n(t) \cdot \cos\varphi_n$:

$$\begin{cases} U_1(t) + U_2(t) \cdot \cos(45°) + U_3(t) \cdot \cos(90°) + U_4(t) \cdot \cos(135°) = U_{sum1}(t) \\ U_1(t) \cdot \cos(45°) + U_2(t) + U_3(t) \cdot \cos(45°) + U_4(t) \cdot \cos(90°) = U_{sum2}(t) \\ U_1(t) \cdot \cos(90°) + U_2(t) \cdot \cos(45°) + U_3(t) + U_4(t) \cdot \cos(45°) = U_{sum3}(t) \\ U_1(t) \cdot \cos(135°) + U_2(t) \cdot \cos(90°) + U_3(t) \cdot \cos(45°) + U_4(t) = U_{sum4}(t) \end{cases} \quad (1)$$

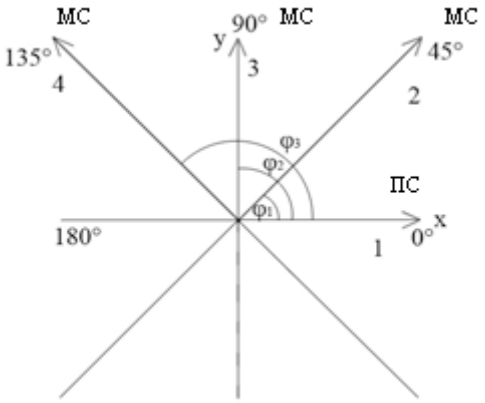


Рис. 3. Расположение плоскостей вектора E для четырех поляризаций с относительным сдвигом в 45 градусов

С помощью системы линейных уравнений (1) запишем выражения для суммарных радиосигналов на входах каждого приемника $U_{sum\,i}(t)$ или пропорциональных им радиосигналов на выходах линейных частей ВЧ трактов приемников:

$U_{sum1}(t) = U_1(t) + U_2(t) \cdot 0.707 + U_3(t) \cdot 0 + U_4(t) \cdot (-0.707)$

$U_{sum2}(t) = U_1(t) \cdot 0.707 + U_2(t) + U_3(t) \cdot 0.707 + U_4(t) \cdot 0$

$U_{sum3}(t) = U_1(t) \cdot 0 + U_2(t) \cdot 0.707 + U_3(t) + U_4(t) \cdot 0.707$

$U_{sum4}(t) = U_1(t) \cdot (-0.707) + U_2(t) \cdot 0 + U_3(t) \cdot 0.707 + U_4(t)$

Далее эти радиосигналы поступают на один из входов устройства обработки суммы радиосигналов, где выделяются полезные сигналы для каждого приемника (см. рис. 1). Математическая модель устройства обработки суммы радиосигналов, структурная схема которого дана в [1, 2], реализуется следующим способом:

Полезный радиосигнал каждого приемника на выходе устройства обработки суммы радиосигналов находится по формулам [1]:

$U_1(t) = U_{sum1}(t) \cdot K11 + U_{sum2}(t) \cdot K12 + U_{sum3}(t) \cdot K13 + U_{sum4}(t) \cdot K14$

$U_2(t) = U_{sum1}(t) \cdot K21 + U_{sum2}(t) \cdot K22 + U_{sum3}(t) \cdot K23 + U_{sum2}(t) \cdot K24$

$U_3(t) = U_{sum1}(t) \cdot K31 + U_{sum2}(t) \cdot K32 + U_{sum3}(t) \cdot K33 + U_{sum2}(t) \cdot K34$

$U_4(t) = U_{sum1}(t) \cdot K41 + U_{sum2}(t) \cdot K42 + U_{sum3}(t) \cdot K43 + U_{sum2}(t) \cdot K44,$

где: K_{ij} – весовой коэффициент; первая цифра i обозначает номер радиосигнала $U_{ri}(t)$, который должен быть получен на выходе сумматора устройства обработки совокупности радиосигналов, а вторая цифра j – номер суммарного радиосигнала $U\,sum_n(t)$, поступающего из радиолинии с номером n. Весовые коэффициенты находятся из системы уравнений (1) в соответствии с [1] и с помощью функции **rref** рассчитаны

в Mathcad. Выделенные на выходах устройства обработки суммы сигналов радиосигналы подвергаются операции демодуляции. После демодуляции каждого радиосигнала принятый информационный сигнал записывается в аудиофайл формата MP3(***output1.mp3, output2.mp3, output3.mp3, output4.mp3)*** и выдаются графики временных зависимостей мгновенных значений переданных и принятых информационных сигналов.

Временные зависимости принятых звуковых информационных сигналов представлены на рис. 4.

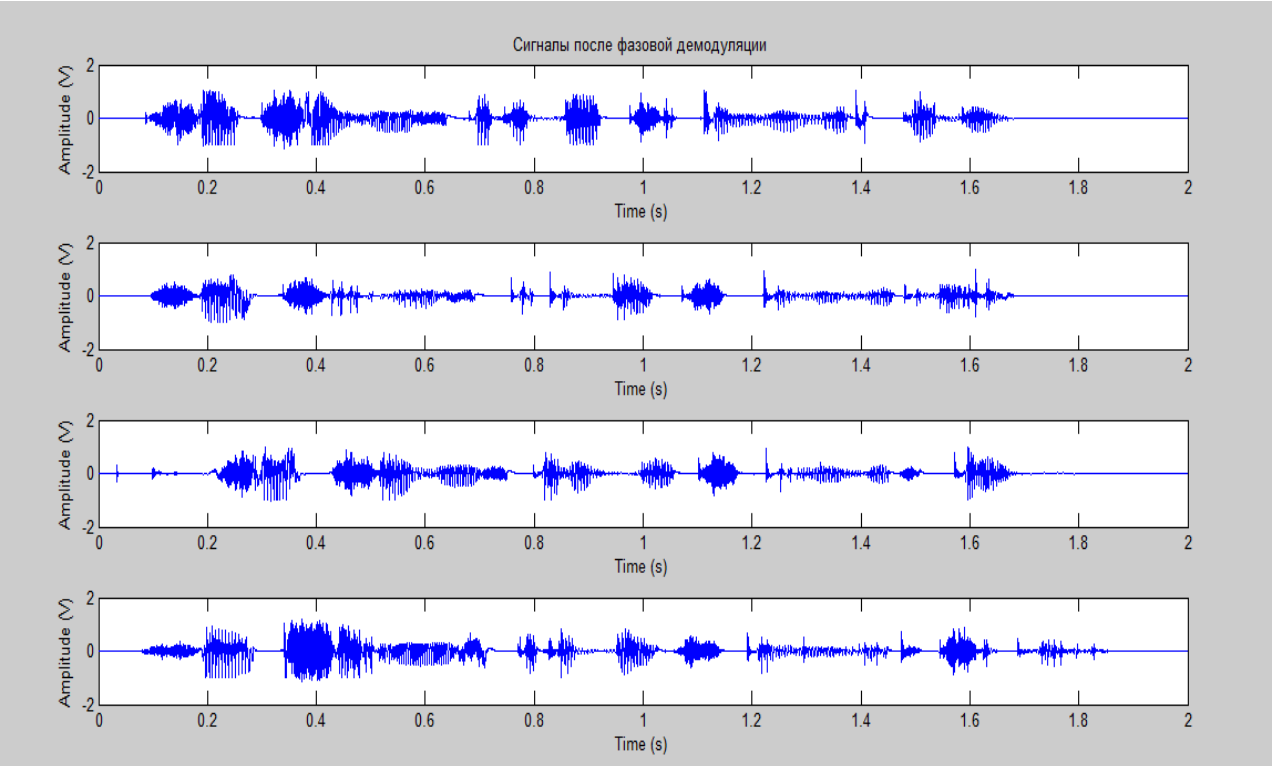


Рис. 4. Звуковые сигналы, полученные на выходах приемников

Выводы

Моделирование радиосистемы с четырехкратным многополяризационным уплотнением спектра показало принципиальную реальность многополяризационного разделения радиосигналов при их передаче на одной и той же несущей частоте при условии стабильности положения поляризаций в процессе распространения радиосигналов. В результате моделирования с использованием в качестве передаваемых сообщений звуковых файлов получены принятые звуковые файлы для каждого приемника, качество которых после прослушивания соответствовало субъективной оценке авторов на 5/5. Полученные результаты дают основания для продолжения как теоретических, так и экспериментальных исследований радиосистем с поляризационным уплотнением спектра.

Литература

1. *Пустовойтов Е.Л.* Многополяризационное уплотнение радиосистем // Электросвязь, 2017. №2.

2. *Пустовойтов Е.Л.* Способ многополяризационного уплотнения радиочастотного спектра в радиосистеме, патент №2609595, 2017 г.

3. https://www.mathworks.com/ (дата обращения: 30.11.2017).

АЛГОРИТМ ИЗМЕНЕНИЯ ЧАСТОТЫ ДИСКРЕТИЗАЦИИ СИГНАЛА ЗВУКОВОГО ВЕЩАНИЯ

Точеный Юрий Михайлович
tocyura@yandex.ru
студент группы МРА1601 МТУСИ

Попов Олег Борисович
МТУСИ, к.т.н., профессор кафедры ТиЗВ
olegp45@yandex.ru

Исследован алгоритм изменения частоты дискретизации звукового вещательного сигнала. Проведен анализ существующих способов изменения частоты дискретизации. Предложен алгоритм изменения частоты дискретизации в частотной области, обеспечивающий уменьшение искажений по сравнению с существующими. Определены виды оконных функций и длительность выборки, обеспечивающие снижение шумов передискретизации до -92 дБ.

Ключевые слова: *преобразование частоты дискретизации (ПЧД), шумы передискретизации, манипуляция отсчетами*.

Структурная схема канала звукового вещания состоит из трех трактов (рис. 1). Из тракта формирования программ, тракта первичного распределения программ и тракта вторичного распределения программ образуется единый электрический канал звукового вещания [1, 4-7]. На практике тракт первичного распределения, как правило, доставляет сигнал до регионального или республиканского тракта формирования программ (радиодома), где он формируется заново с включением местных программ и вновь попадает в тракт первичного распределения, и так несколько раз.

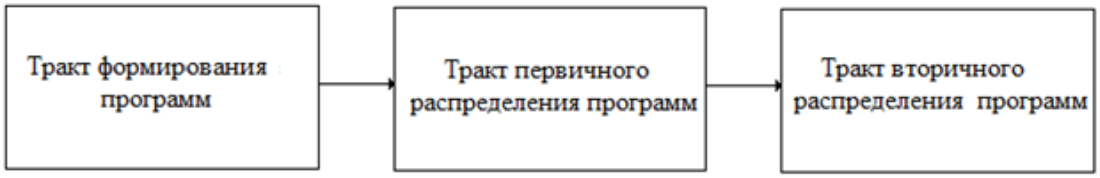


Рис. 1. Структурная схема системы звукового вещания

В каждом тракте используется своя частота дискретизации ($F_д$) определяемая верхней частотой передаваемого сигнала или особенностями звеньев тракта. В тракте формирования используются $F_д$ 48 и 44,1 кГц. В тракте первичного распределения 32, 24 и 16 кГц для передачи сигнала с верхней частотой 15, 10 и 6,4 кГц, соответственно. Преобразование $F_д$ сопровождается накоплением искажений.

На первом этапе внедрения в канале звукового вещания цифровых устройств, преобразование $F_д$ осуществлялось с переходом к аналоговому сигналу с помощью цифро-аналогового преобразователя (ЦАП) и аналого-цифрового преобразователя (АЦП) с новой частотой дискретизации. Для оценки искажений при квантовании гармонического сигнала используют соотношение сигнал/шум квантования $ОСШ_{кв}$, выражение которого представлена ниже [2]:

$$ОСШ_{кв} = 10\lg\frac{12\cdot\left(U_c/\sqrt{2}\right)^2}{\left(U_c/2^{n-1}\right)^2} = 6\cdot n + 1.8\ [\text{дБ}] \qquad (1)$$

где n – количество разрядов, U_c – амплитуда гармонического сигнала. В зависимости от количества разрядов квантователя, отношение сигнал шум будет разным, а так как процесс квантования осуществляется несколько раз в тракте звукового вещания, то данные искажения будут накапливаться при каждой передискретизации. Кроме того, дополнительные шумы 1,5 – 4 дБ добавляются за счет неточности исполнения АЦП и ЦАП схем.

В настоящее время используется цифровое преобразование частоты дискретизации (ПЧД). Способ изменения частоты дискретизации во временной области заключается в применении общих кратных частот

двух сопрягаемых трактов. Это реализуется с помощью добавления L-1 нулевых отсчетов между каждым исходным отсчетом, что приводит к увеличению частоты дискретизации исходного сигнала *x(m)* до минимальной кратной частоты выходного сигнала [2].

$$x_1(m) = \begin{cases} x_1(m/L), m = 0, \pm L, \pm 2L.. \\ 0, \text{при других } m \end{cases} \quad (2)$$

Далее с помощью интерполяционного фильтра, который имеет коэффициенты импульсного отклика *h(k)*, вычисляются добавленные отсчеты:

$$x_2(m) = \sum_{k=1}^{N} h(k) \cdot x_1(m-k) \quad (3)$$

где $N = k \cdot (L-1)$ – количество требуемых коэффициентов, а *k* – количество опорных отсчетов интерполяции. Последней процедурой является выборка отсчетов выходного сигнала с требуемой частотой дискретизации, кратной минимальной общей частоте дискретизации - выборка каждого *M*-го отсчета:

$$y(m) = x_2(M \cdot m) \quad (4)$$

В конечном результате получаем требуемую частоту дискретизации выходного сигнала:

$$F_{Дy} = (L/M) \cdot F_{Дx} \quad (5)$$

где $F_{Дy}$ - частота дискретизации выходного сигнала, а $F_{Дx}$ - частота дискретизации входного сигнала.

При выполнении компьютерного моделирования прохождения звукового сигнала через тракт звукового вещания, между двумя радиодомами с F_o 48-32-48 кГц было показано, что искажения соответствуют ГОСТУ Р 52742-2007 только при передаче синусоидального сигнала. При передаче реального вещательного сигнала ошибка превышает допустимое значение 2% (рис. 2).

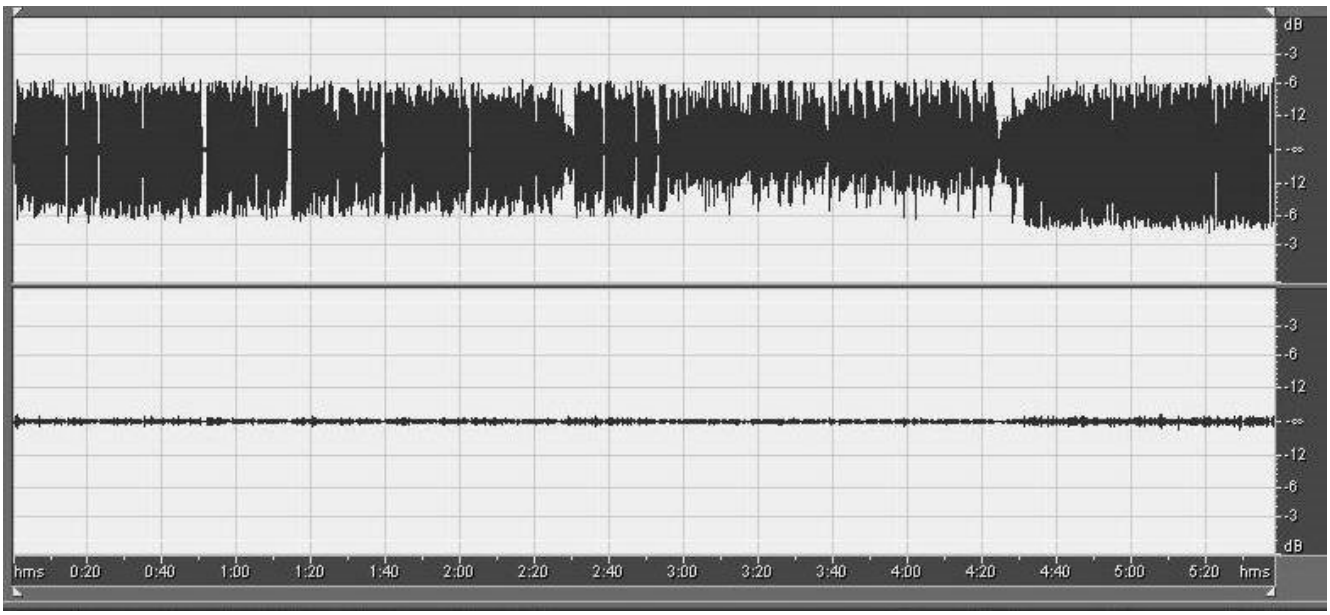


Рис. 2. Исходный сигнал и сигнал ошибки передискретизации

На рисунке 3 представлен график спектра звукового сигнала и шумов ошибки. Видно, что спектр шумов ПЧД имеет максимум в области высоких частот не маскируемых сигналом, а потому будет особенно заметен слушателю. Кроме того, ошибки ПЧД особенно велики в моменты нестационарностей сигнала – атак, что особенно заметно для слушателя.

Кафедрой ТиЗВ МТУСИ разработан способ ПЧД с промежуточным переходом в частотную область [3], что позволяет осуществлять манипуляции с нулевыми коэффициентами, на частотах выше верхней частоты самого сигнала, что дает возможность значительно уменьшить искажения при передискретизации. Упрощенный алгоритм ПЧД в частотной области представлен на рис. 4.

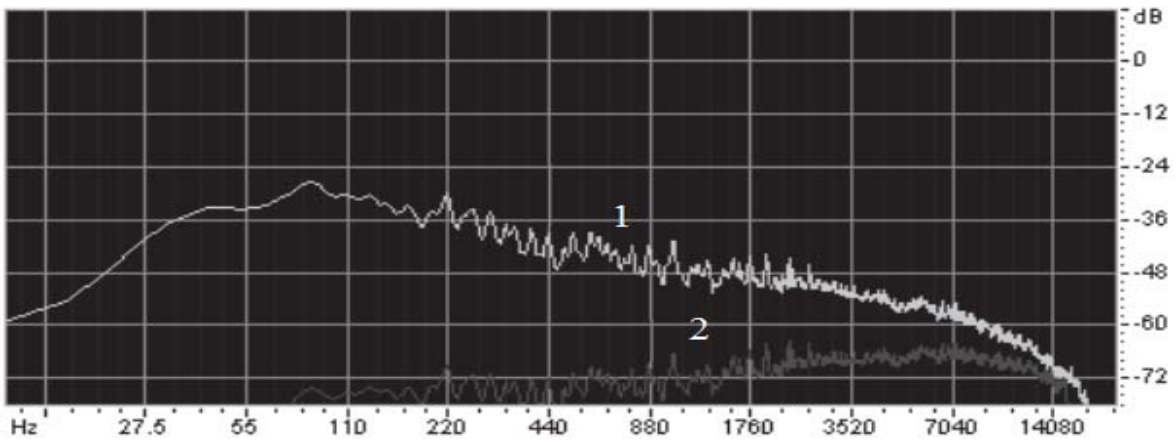


Рис. 3. Спектр сигнала: 1 – исходный сигнал, 2 – шумы передискретизации

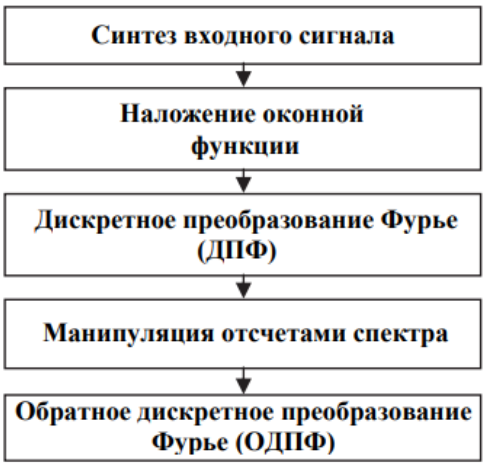


Рис. 4. Структурная схема ПЧД с промежуточным переход в частотную область

Для перехода в частотную область используется дискретное преобразование Фурье (ДПФ). Так как количество отсчетов во временной и коэффициентов частотной областях совпадает, при удалении или добавлении необходимого количества частотных коэффициентов изменяется частота дискретизации [3].

Шумы ПЧД во многом определяются выбором оконной функции при реализации ДПФ. Нами были исследованы шумы ПЧД при использовании следующих оконных функций: прямоугольное окно, треугольное окно, окно Хемминга, окно Ханна и окно Наттолла (Кайзера-Бесселя).

По результатам исследования, приведённым на рис. 5, наименьшие шумы ПЧД (СПМ) обеспечиваются при использовании окна Наттола.

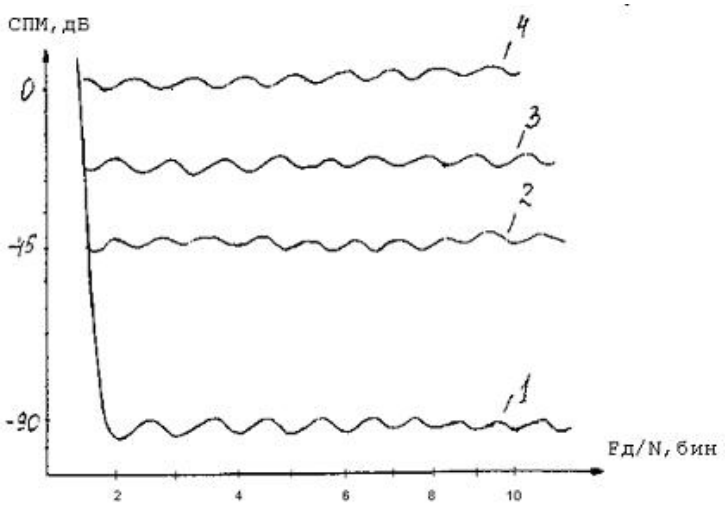


Рис. 5. СПМ при использовании:
1 – окно Наттолла, 2 – Окно Хемминга, 3 – треугольное окно, 4 – прямоугольное окно

Окно Наттола не обеспечивает единичного коэффициента передачи при любом проценте перекрытия оконных функций, поэтому после ОДПФ необходимо использовать компенсирующую оконную функцию.

Шумы ПЧД зависят и от длительности выборки на которой производится преобразование, проведенное нами компьютерное моделирование алгоритма на реальном звуковом сигнале показало (рис. 6), что уже при использовании 400 отсчетов отношение сигнал/шум ниже допуска ГОСТ, а при длительности выборки более 2000 отчетов достигает – 92 дБ.

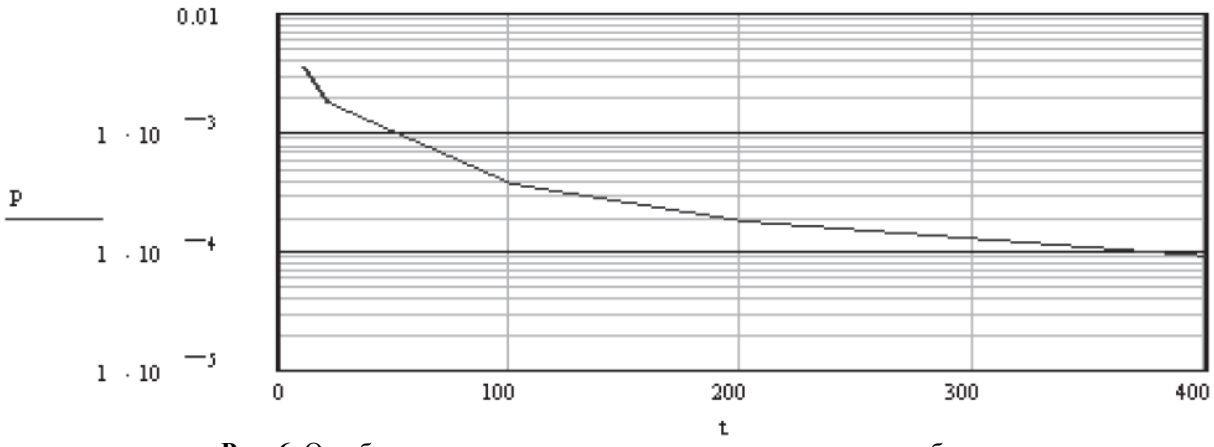


Рис. 6. Ошибка предискретизации в зависимости от длины выборки

Выводы

- Оценка качества преобразования частоты дискретизации с помощью стационарного синусоидального сигнала, в соответствии с ГОСТ Р 52742-2007 недостаточна и не гарантирует отсутствия искажений при передаче реального вещательного сигнала.
- Предложенный кафедрой ТиЗВ способ изменения частоты дискретизации в частотной области позволяет минимизировать искажения до величины нормируемой ГОСТ (-60 дБ) на реальном вещательном сигнале.
- Проведенные нами исследования показали возможность снижения ошибки передискретизации до уровня -92 дБ за счет выбора оптимальной оконной функции, которой оказалось окно Наттола.
- Определено, что для достижения ошибки на уровне -92 дБ длительность выборки на которой производится преобразование частоты дискретизации должна составлять не менее 2000 точек.

Литература

1. ГОСТ Р 52742-2007. Каналы и тракты звукового вещания. Основные параметры качества. Методы измерении.
2. *Рихтер С.Г.* Цифровое радиовещание обработка и измерения сигналов в трактах звукового вещания. М.: Инсвязьиздат, - 2010.
3. *Абрамов В.А., Попов О.Б., Рихтер С.Г.* Патент РФ №2405262 «Способ изменения скорости передачи цифрового звукового сигнала телерадиовещания и устройство для его осуществления» Опубл. БИ №33 27.11.2010. 13 с.
4. *Попов О.Б., Рихтер С.Г.* Цифровая обработка сигналов в трактах звукового вещания. Москва, 2007. Сер. Учебное пособие для высших учебных заведений.
5. *Абрамов В.А., Венедиктов М.Д., Попов О.Б., Рихтер С.Г.* Результаты обработки сигналов цифрового радиовещания // T-Comm: Телекоммуникации и транспорт. 2012. Т. 6. № 10. С. 4-6.
6. *Абрамов В.А., Попов О.Б., Рихтер С.Г.* Аудиопроцессорная обработка сигналов цифрового вещания и ее последствия // T-Comm: Телекоммуникации и транспорт. 2016. Т. 10. № 6. С. 17-20.
7. *Абрамов В.А., Попов О.Б., Ождихин Г.М., Рихтер С.Г.* Оценка качества обработки звуковых сигналов в радиовещательных студиях // T-Comm: Телекоммуникации и транспорт. 2013. Т. 7. № 9. С. 6-8.

ОПРЕДЕЛЕНИЕ ХАРАКТЕРА ВОЛНЕНИЯ В ТОЧКЕ РАСПОЛОЖЕНИЯ СУДНА С ПОМОЩЬЮ ИЗМЕРЕНИЙ ДОПЛЕРОВСКИХ СДВИГОВ ЧАСТОТЫ СУДОВОГО ПЕРЕДАТЧИКА

Давыдов Артем Владимирович
магистрант группы М151601(70) МТУСИ
artem.daw@yandex.ru

Репинский Владимир Николаевич
МТУСИ, к.т.н. доцент кафедры ИСУА
repinski@rambler.ru

При движении морского подвижного объекта (МПО) в акватории, если период волны заранее известен, можно рассчитать период качки корабля и частоту максимального отклонения передающей антенны. Расчет частоты в рамках допустимой погрешности дает возможность сделать вывод о размерах МПО, находящегося за пределами видимости.

Ключевые слова: *морской подвижный объект, доплеровский сдвиг, период качки, частота максимального отклонения антенны, морское волнение.*

В настоящее время проблемы управления морскими подвижными объектами являются особенно актуальными. Главные задачи навигации – обеспечение условий безопасного мореплавания при увеличивающейся интенсивности судоходства, осуществление сложных маневров, прогнозирование возможных нестандартных ситуаций. Основными причинами неопределенности динамики МПО являются неточность и неполнота информации об объекте и условиях его функционирования, а также погрешность измерений, используемых для формирования управляющих воздействий в процессе движения, неполнота знаний о внешних возмущениях.

К ним следует отнести резкие перепады волновых воздействий на МПО, течения и др.

Задача расчета движения объекта решается методами и приборами мореходной навигации, которые позволяют определить местоположение и ориентацию движущегося объекта относительно принятой системы координат, величину и направление скорости движения, направление и расстояние до места назначения и т.д.

К задачам навигации также относится определение оптимального маршрута движения, под которым понимается требование обеспечения максимальной безопасности и экономичности вывода объекта в заданную точку пространства в определенный момент времени с установленной точностью.

Особенности доплеровского сдвига

Основным фактором развития технических средств судовождения являются достижения в области навигационного приборостроения. Они обусловлены возросшим числом проблем в навигации, связанных с ростом числа судов, увеличением их скоростей и размеров.

Для обеспечения безопасности судоходства в первую очередь важна точность показаний. Если учитывать статистику мировых показателей возросшей аварийности судов в последнее время, можно сделать вывод, что требуется повысить безопасность движения судов в акватории. Поэтому необходимо получать более детальную информацию о судах: положение в пространстве, направление движения и размер объекта [2].

В расчет принимаются сам МПО и несколько наземных станций слежения, расположенных на большом расстоянии друг от друга. Наземные станции обеспечивают прием телеметрической информации, следят за траекторией движения и используются для анализа сигнала с целью примерного расчета габаритов судна.

В данной работе предлагается решение для расчета размеров МПО по доплеровским особенностям изменения частоты его передающей антенны (рис. 1). Ведется расчет максимальной частоты отклонения передающей антенны (v_{max}) в условиях морского волнения относительно ее стационарного положения.

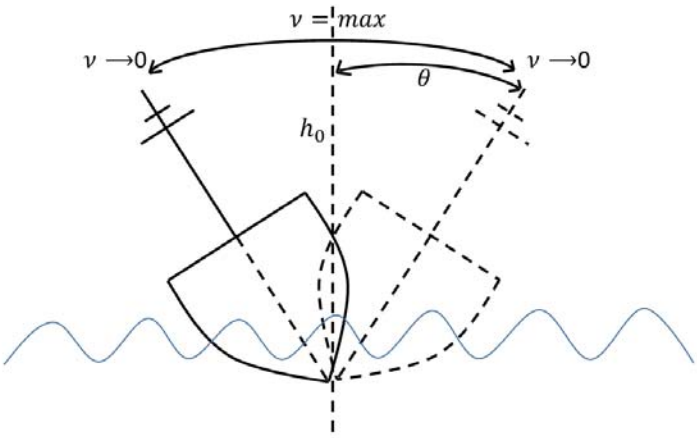


Рис. 1. Колебания МПО в условиях волнения

Движение источника излучения, находящегося под активным воздействием полупроводящей среды, в той или иной степени отражается на параметрах излученного им радиосигнала. Это в первую очередь доплеровское изменение частоты, изменение поляризации отраженного сигнала (из-за смещения области отражения), изменение уровня сигнала из-за изменения угла падения луча на ионосферу, а также из-за электризации антенн и элементов их крепления при большой относительной скорости «Антенна – электризующая среда».

Для оценки перечисленных изменений параметров и последующего построения алгоритма обнаружения и идентификации МПО необходимо рассмотреть параметры траектории движения антенных систем МПО, которые, в свою очередь, определяются колебаниями корпуса МПО.

Свободные колебания судна

Свободная качка на тихой воде. Дифференциальное уравнение бортовой качки МПО, соответствующее рис. 1, имеет вид: [3]

$$-(I_x + \Delta I_x)\theta'' - N\theta' - Dh_0\theta = 0, \quad (1)$$

где I_x – момент инерции МПО;

ΔI_x – присоединенный момент инерции (влияние воды);

θ – угол крена;

$N\theta'$ – момент демпфирующих сил (сил сопротивления, направленных против угла крена);

$Dh_0\theta$ – восстанавливающий момент;

h_0 – метацентрическая высота МПО;

$(Ix + \Delta Ix)\theta''$ – момент сил инерции, по принципу Даламбера направленный в сторону, обратную угловому ускорению.

Решение уравнения (1) в предположении малой килеватости МПО, то есть без учета нелинейности сопротивления среды (воды), получается гармоническим с частотой

$$\omega_b = \sqrt{\frac{Dh_0}{I_x + \Delta I_x}} \quad (2)$$

и временем периода полного бортового колебания

$$\tau_b = 2\pi\sqrt{\frac{I_x + \Delta I_x}{Dh_0}} = \frac{K_i B}{\sqrt{h_0}}, \quad (3)$$

где B – ширина корпуса МПО, коэффициент K_i, зависящий от параметров МПО и от его загрузки, рассчитывается отдельно для каждого судна. Ориентировочное значение K_i в зависимости от типа МПО и его загрузки находится в пределах 0,62–0,86.

Период гармонической килевой качки рассчитывают по формуле

$$\tau_k = 2\pi T\sqrt{\frac{0{,}98^2\left(1+\frac{0{,}8\,B\,B}{T}\right)}{g\alpha}} \approx (2{,}7-3)T, \quad (4)$$

где T – осадка МО в метрах; g, α – коэффициенты полноты корпуса судна.

Вынужденные колебания морского объекта

Вынужденные колебания судна совершаются под действием сил, обусловленных волнением, ветровым воздействием. Рассмотрим основные механизмы возбуждения вынужденных колебаний МПО.

Бортовая качка при волнении. Решение неоднородного дифференциального уравнения колебаний МПО имеет следующий вид:

$$\theta_m = \aleph_{\frac{T}{\lambda_0}} \aleph_{\frac{B}{\lambda_0}} \sqrt{\frac{(1-q)^2+4\mu_\gamma^2}{2\mu_\gamma^2}}, \quad (5)$$

где μ_γ – относительный безразмерный коэффициент сопротивления, обычно находящийся в пределах 0,05–0,18; $q = \frac{\Delta I_{yy}}{I_{xx}+\Delta I_{xx}}$ – безразмерный коэффициент, равный 0,15–0,20; $\aleph_{\frac{T}{\lambda_0}} \aleph_{\frac{B}{\lambda_0}}$ – редукционные коэффициенты МПО, определяемые формой смоченной части корпуса.

Для определения влияния качки на доплеровский сдвиг частоты необходимо рассчитывать данные, учитывая два вида качки: бортовую и килевую.

Для этого следует принять во внимание, что перемещения антенны будут гармоническими и мгновенные значения будут вычисляться по формулам [1]

$$\alpha = A_k \sin\left(\frac{2\pi t}{T_\alpha} + A_0\right), \quad (6)$$

$$\gamma = \Gamma_k \sin\left(\frac{2\pi t}{T_\gamma} + \Gamma_0\right), \quad (7)$$

где A_k – амплитуды бортовой качки; Γ_k – амплитуда килевой качки; T_α – период бортовой качки; T_γ – период килевой качки; A_0 – начальная фаза бортовой качки; Γ_0 – начальная фаза килевой качки.

В таком случае можно вычислить величину доплеровского сдвига:

$$\Delta F_\alpha = fh\frac{d\alpha}{dt}\cos\nu, \quad (8)$$

$$\Delta F_\gamma = fh\frac{d\gamma}{dt}\cos\theta, \quad (9)$$

где ν – угол между линейной скоростью движения антенны, вызванной бортовой качкой; θ – угол между линейной скоростью движения антенны, вызванной килевой качкой; h – высота антенны; f – частота передатчика.

Вывод

С помощью приведенных формул можно провести оценку доплеровского сдвига и в дальнейшем использовать это для определения размеров морского подвижного объекта.

Литература

1. *Мирошников В.В., Нестеренко В.Б., Завальнюк И.П., Завальнюк О.П.* Моделирование качки судна при разных условиях загрузки: науч // Восточно-Европейский журнал передовых технологий. 2014. 70 с. ISSN 1729-3774,
2. *Сорочинский В.А., Якшевич Е.В.* Спутниковые системы морской навигации. М.: Транспорт, 2007. 200 с.
3. Спутниковые навигационные системы [Электронный ресурс]. Режим доступа: http://sea-library.ru/gmdss/74-sputnikovye-navigatsionnye-sistemy.html. Спутниковые навигационные системы, свободный. – Морская библиотека.

ИНФОРМАЦИОННЫЕ АСПЕКТЫ СЖАТИЯ ДАННЫХ ИСТОЧНИКА НЕПРЕРЫВНЫХ СООБЩЕНИЙ

Малиночкин Вячеслав Сергеевич
студент группы БЗС1401 МТУСИ
Pyton.Mr.Malina@yandex.ru

Санников Владимир Григорьевич
МТУСИ, к.т.н., профессор кафедры ОТС
tes_mtuci@mail.ru

Рассматривается математическая формализация задачи эффективного сжатия данных источника непрерывных гауссовских сообщений. При среднеквадратической мере искажения и модели сообщения со спектром, содержащим n-кратные полюса, впервые решается задача минимизации скорости создания сообщений (эпсилон-энтропии) непрерывного источника. Рассмотренная методика может быть использована в задачах минимизации объема памяти запоминающих устройств или минимизации скорости передачи сообщений по идеальному каналу связи.

Ключевые слова: *непрерывное сообщение, сжатие данных, случайное кодирование, эпсилон-энтропия, идеальный канал связи, функция скорость-искажение.*

Введение

В теории и технике телекоммуникаций актуальна проблема максимального уменьшения (сжатия) объема (данных) цифрового представления различного вида непрерывных сообщений (речевых, звукового вещания, телевизионных и др.) без потери для получателя их информационного содержания [1, 9]. Известно, что точное представление случайного непрерывного сообщения $A(t)$ обладает бесконечной энтропией [2], что не является конструктивным. Однако, из-за ограниченности разрешающей способности информационно-измерительных систем, реальной чувствительности приемных устройств и органов чувств человека на практике не требуется точного представления сообщения. Достаточно воспроизвести его с *ограниченной точностью*, характеризуемой некоторым малым параметром ε (эпсилон). Наиболее часто в качестве меры точности представления непрерывного сообщения с нулевым средним в момент t используют среднеквадратичное приближение $A(t)$ его эпсилон приближением $A_\varepsilon(t)$

$$\overline{\varepsilon^2} = \mathrm{M}\left[A - A_\varepsilon\right]^2 = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(x - x_\varepsilon)^2 W(x, x_\varepsilon)\,dx\,dx_\varepsilon\,, \qquad (1)$$

где $\overline{\varepsilon^2} = D_\varepsilon \le \varepsilon_0^2$, D_ε – дисперсия погрешности ε, ε_0^2 – достаточно малая допустимая величина, M – символ статистического усреднения, $W(x, x_\varepsilon) = W(x_\varepsilon)W_{A|A_\varepsilon}(x \mid x_\varepsilon)$ – совместная функция плотности вероятности (ФПВ) случайных величин A_ε и A [3].

Количество взаимной информации, содержащееся в A_ε относительно A определяется соотношением [2, 4]

$$I(A_\varepsilon; A) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} W_{A_\varepsilon A}(x, x_\varepsilon)\log_b \frac{W_{A_\varepsilon A}(x, x_\varepsilon)}{W_A(x)W_{A_\varepsilon}(x_\varepsilon)}\,dx\,dx_\varepsilon\,. \qquad (2)$$

А.Н. Колмогоров ввел понятие *эпсилон-энтропия – как минимальное количество взаимной информации*, необходимое для определения A с заданной точностью ε [5]

$$H_\varepsilon(A) = \min_{W_{A|A_\varepsilon}(x|x_\varepsilon)} I(A; A_\varepsilon)\,, \qquad (3)$$

где минимум берется по всем условным ФПВ $W_{A|A_\varepsilon}(x \mid x_\varepsilon)$, для которых $D_\varepsilon \le \varepsilon_0^2$, т.е. по всем возможным способам цифрового представления (кодирования) сообщения A_ε с заданной ФПВ $W(x_\varepsilon)$.

Минимизация количества взаимной информации (2), а тем более отыскание оптимальных методов эффективного кодирования непрерывных сообщений является в общем случае нерешенной задачей. Известны лишь частные случаи [6], связанные с оптимальным квантованием гауссовской случайной величины A, а также с определением скорости создания сообщений $I_\varepsilon^{'} = H_\varepsilon(A)/T$, где T – длительность кодовых символов, для гауссовского стационарного процесса с нулевым средним и известной спектральной плотностью мощности (СПМ) $G_A(f)$ при среднеквадратической мере искажения (1). Гипотетическая система передачи непрерывного сообщения при этих условиях может быть представлена в виде следующих функциональных блоков: источник непрерывного случайного сообщения $A(t)$, случайное эффективное кодирование со скоростью $I_\varepsilon^{'}$, цифровой канал связи, случайное эффективное декодирование, ε – оценка непрерывного сообщения $A_\varepsilon(t)$. Случайные процедуры кодирования и декодирования между собой взаимно связаны. Рассмотрим эти вопросы детальнее.

Параметрические уравнения при случайном кодировании

Пусть сообщение $A(t)$ представляет собой случайный, стационарный гауссовский процесс с СПМ $G_A(f), f \geq 0$, и передается по цифровому каналу в виде последовательности цифровых символов. Если эта последовательность двоичная, то она характеризуется битовой скоростью передачи V_b, измеряемой в бит/с. Главной характеристикой цифровой системы является зависимость среднеквадратической погрешности передачи (СКП) сообщения от скорости передачи битов. Наилучшей (при отсутствии помех в канале связи) является такая система, которая обеспечивает наименьшую СКП передачи сообщения при заданной скорости передачи битов или же минимальную скорость передачи битов при заданной СКП передачи сообщения.

В реальной системе связи выполняется условие $V \geq I_\varepsilon^{'}$, где $I_\varepsilon^{'}$ – информационная производительность источника сообщения или скорость создания сообщений источником, равная эпсилон-энтропии в единицу времени. *Равенство $V = I_\varepsilon^{'}$ достигается при равновероятных и независимых битах***.** Таким образом, задача свелась к определению эпсилон-энтропии непрерывного сообщения. В теории эпсилон-энтропии эта задача решается следующим образом [6]. Область частот $f \geq 0$ разбивается на малые интервалы df. В каждом таком интервале дисперсия сообщения равна $dD_A = G_A(f)df$. При известной СПМ шума цифрового представления $G_\varepsilon(f), f \geq 0$, дисперсия шума в элементарной полоске равна $dD_\varepsilon = G_\varepsilon(f)df$. Для обеспечения минимума эпсилон-энтропии шум (ε - погрешность) должен быть гауссовским случайным процессом.

Применив теорему отсчетов к частотно-ограниченным процессам элементарных полосок, с учетом $I_\varepsilon^{'} = H_\varepsilon(A)/T$, приращение информационной производительности непрерывного источника можно представить в виде [5, 6]

$$dI_\varepsilon^{'} = \min\frac{dH_\varepsilon(A)}{dt} = \min\left[\frac{1}{2}\log_2\frac{dD_A}{dD_\varepsilon}\right]2df = \min\left[\log_2\frac{G_A(f)}{G_\varepsilon(f)}\right]df .$$

Проинтегрировав это соотношение по частоте, получаем

$$I_\varepsilon^{'} = \min\left[\int_0^\infty \log_2\frac{G_A(f)}{G_\varepsilon(f)}df\right]. \qquad (4)$$

Минимизация этого выражения должна осуществляться определением экстремальной СПМ шума цифрового представления $G_\varepsilon(f), f \geq 0$. Это вариационная задача, которая решается при следующем ограничении: ввиду того, что шум является частью сообщения, должно выполняться неравенство

$$G_\varepsilon(f) \leq G_A(f). \qquad (5)$$

Помимо этого по условию нормировки имеем

$$\int_0^\infty G_\varepsilon(f)df = \overline{\varepsilon^2} = D_\varepsilon. \qquad (6)$$

Решая вариационную задачу определения экстремальной функции $G_\varepsilon(f)$, приходим к следующему соотношению

$$G_\varepsilon(f)=\begin{cases}G_A(v)=\lambda=const,\ 0\le f<v;\\ G_A(f), \qquad\qquad v\le f<\infty.\end{cases} \tag{7}$$

При этом дисперсия погрешности цифрового представления определяется так

$$D_\varepsilon=\lambda\cdot v+\int_v^\infty G_A(f)df\,, \tag{8}$$

где v – проекция на ось частот точки пересечения постоянной λ с СПМ $G_A(f)$.

Уравнения (4) и (8) задают параметрическое семейство уравнений для определения функции скорость-искажение при эффективном статистическом кодировании непрерывного сообщения $A(t)$.

Модель источника непрерывных сообщений

Для получения численных результатов решения параметрических уравнений (4-8) требуется задаться моделью источника непрерывных сообщений. В современной теории телекоммуникаций для формирования непрерывных сообщений все чаще используется метод формирующего фильтра [3, 9]. Здесь для получения случайного непрерывного сообщения как отклика формирующего фильтра с заданными характеристиками (импульсной реакцией или комплексным коэффициентом передачи) на вход фильтра подается стандартный случайный процесс с известными вероятностными характеристиками. Используя данный метод, примем в качестве модели случайного сообщения $A(t)$ отклик формирующего линейного фильтра с коэффициентом передачи

$$K(j\omega)=\frac{1}{\left[1+\left(j\omega/\alpha\right)\right]^n},\quad f\ge 0\,, \tag{9}$$

на вход которого действует белый гауссовский шум $\xi(t)$ с характеристиками

$$M\{\xi(t)\}=0;\quad M\{\xi(t_1)\xi(t_2)\}=\frac{1}{2}G_0\delta(t_2-t_1);\quad G_0=\text{const}, \tag{10}$$

где $\delta(t_2-t_1)$ - дельта–функция Дирака, $\Delta\omega_\gamma=\alpha=2\pi F_\gamma$.

Односторонняя СПМ этого сообщения, очевидно, равна

$$G_A(f,n)=\frac{1}{2}G_0K^2(f)=\frac{G_0}{2\left[1+\left(f/F_n\right)^2\right]^n}. \tag{11}$$

Дисперсия сообщения, с учетом табличного интеграла, приводится к виду

$$D_A=2\int_0^\infty G_A(f,n)df=\int_0^\infty\frac{G_0}{\left[1+(f/F_n)^2\right]^n}df=G_0F_n\int_0^\infty\frac{dx}{\left[1+x^2\right]^n}=\frac{G_0F_n\pi}{2}\frac{(2n-3)!!}{(2n-2)!!}. \tag{12}$$

Пусть полоса частот спектра сообщения фиксирована и равна $\Delta\omega_\gamma=\alpha=2\pi F_\gamma$. Найдем величину F_n в (12) из условия ослабления амплитудного спектра сообщения на величину $1/\gamma$ при любом порядке n формирующего фильтра. В результате имеем

$$F_n=F_\gamma/\gamma_n,\quad \gamma_n=\sqrt{\gamma^{2/n}-1}\,, \tag{13}$$

где при ослаблении в a дБ $\gamma=10^{a/20}$. Например, при $\gamma=\sqrt{2}$, $a=3$ дБ.

Функция скорость–искажение

Для определения функции скорость-искажение вначале подставим (7) и (11) в (4). В результате находим $I_\varepsilon^{'}=\int_0^v\log_2\left[\frac{G_A(f,n)}{\lambda}\right]df=\int_0^v\log_2\left[\frac{1+\left(v/F_n\right)^2}{1+\left(f/F_n\right)^2}\right]^n df=\frac{nF_n}{\ln 2}\left\{\frac{v}{F_n}\ln\left[1+\left(v/F_n\right)^2\right]-\int_0^{v/F_n}\ln\left[1+x^2\right]dx\right\}$. Затем с учетом табличного интеграла вида [7]

$$\int_0^{v/F_n}\ln\left[1+x^2\right]dx=\frac{v}{F_n}\ln\left[1+\left(v/F_n\right)^2\right]-\frac{2v}{F_n}+2\,\text{arctg}\left(\frac{v}{F_n}\right),$$

вычисляем относительную информационную производительность непрерывного источника, характеризующую спектральную эффективность системы цифровой передачи непрерывного сообщения

$$C_\varepsilon(n) = \frac{I_\varepsilon^{'}(n)}{F_n} = \frac{2n}{\ln 2}\left\{\frac{v}{F_n} - \operatorname{arctg}\left(\frac{v}{F_n}\right)\right\} \quad \text{[бит/с·Гц]}. \tag{14}$$

Теперь с учетом (8) ÷ (10) находим дисперсию погрешности цифрового представления непрерывного сообщения

$$D_\varepsilon = \lambda v + \int_v^\infty G_A(f)df = \lambda v + \int_0^\infty G_A(f)df - \int_0^v G_A(f)df = D_A\left[1 - \frac{1}{D_A}\left(\int_0^v G_A(f)df - \lambda v\right)\right].$$

Отсюда получаем относительную СКП цифрового представления сообщения

$$\delta_\varepsilon = \frac{D_\varepsilon}{D_A} = \left[1 - \frac{1}{D_A}\left(\int_0^v G_A(f)df - \lambda v\right)\right], \tag{15}$$

где $\lambda = \dfrac{G_0}{\left[1+\left(v/F_n\right)^2\right]^n}$, $\quad \displaystyle\int_0^v G_A(f)df = G_0F_n\int_0^{v/F_n}\frac{df}{\left[1+x^2\right]^n}$.

Последний интеграл табличный [7] и приводится к следующему виду

$$\int_0^v G_A(f)df = G_0F_n\left[\frac{v}{F_n(2n-1)}\sum_{k=1}^{n-1}\frac{\prod_{i=1}^{k}\left[(2n-2i+1)/(n-i)\right]}{2^k\left[1+(v/F_n)^2\right]^{n-k}} + \frac{(2n-3)!!}{(2n-2)!!}\operatorname{arctg}\left(\frac{v}{F_n}\right)\right].$$

Подставляя полученные соотношения в (15), после ряда преобразований, окончательно для относительной СКП получаем следующую зависимость

$$\delta_\varepsilon = 1 - \frac{2}{\pi}\left\{\operatorname{arctg}\left(\frac{v}{F_n}\right) - \frac{v}{F_n}\frac{c_n}{\left[1+\left(v/F_n\right)^2\right]^n}\left[1 - \frac{1}{(2n-1)}\sum_{k=1}^{n-1}c_{n,k}\left[1+\left(v/F_n\right)^2\right]^k\right]\right\}, \tag{16}$$

где введены обозначения:

$$c_n = \frac{(2n-2)!!}{(2n-3)!!}, \quad c_{n,k} = \frac{1}{2^k}\prod_{i=1}^{k}\frac{(2n-2i+1)}{(n-i)}.$$

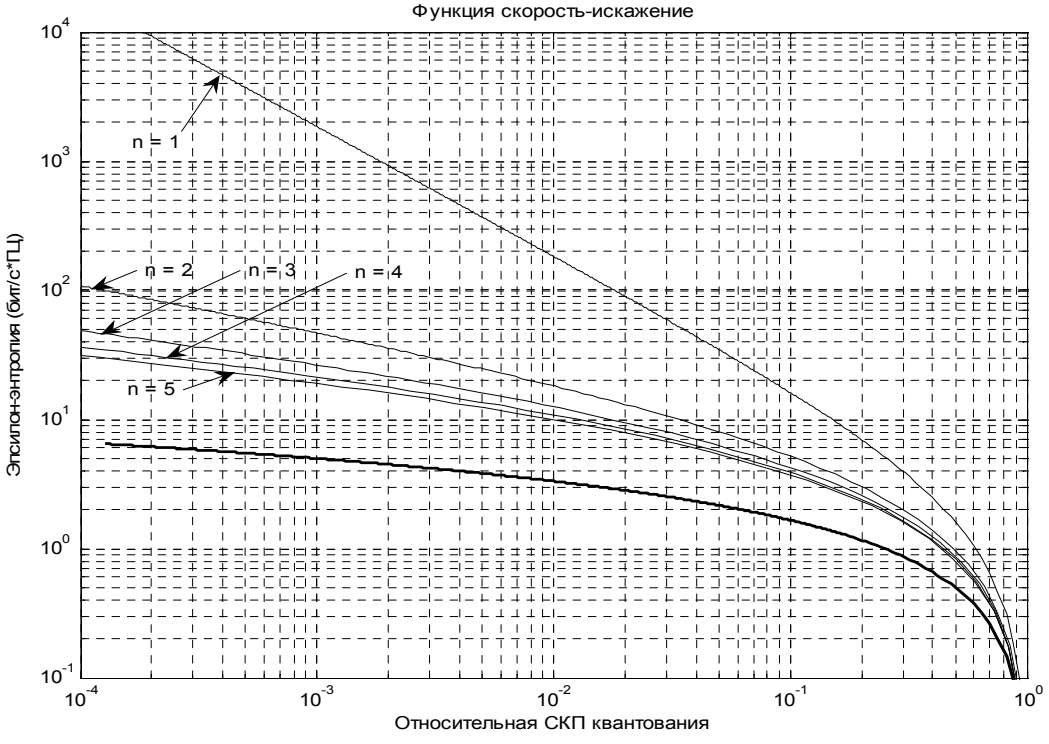


Рис. 1. Функция скорость-искажение для непрерывного гауссовского сообщения

Собственно *функция скорость-искажение* (*ФСИ*) вида: $C_\varepsilon(n) = \varphi(\delta_\varepsilon)$, характеризуется теперь параметрической системой из двух уравнений (14) и (16). Графики ФСИ получаются путем изменения параметра ν в пределах от 0 до ∞ и фиксации в этих точках величин $C_\varepsilon(n)$ в (14) и δ_ε в (16). На рисунке 1 приведены ФСИ для различных порядков n формирующего фильтра сообщения.

На рисунке 1 верхняя ФСИ, соответствующая $n = 1$ (фильтр Баттерворта первого порядка), совпадает с зависимостью, полученной в работе [6]. Самая нижняя ФСИ соответствует значению $n \to \infty$. В этом случае формирующий фильтр – есть идеальный фильтр нижних частот (ИФНЧ) [2]. Графики для $2 \le n < \infty$ являются новыми и получены впервые.

Функция скорость-искажение позволяет при фиксированной среднеквадратичной погрешности квантования определить минимальное число бит на отсчет при кодировании дискретно-непрерывного сообщения, что очень важно в системах сжатия данных, отображающих речевые, звукового вещания и телевизионные сообщения. Так, например, при относительной СКП цифрового представления гауссовского сообщения равной $\delta_\varepsilon = D_\varepsilon / D_A = 0,001$, величина $C_\varepsilon(n)$ при различных n, соответственно, равны $C_\varepsilon(1) = 1800$, $C_\varepsilon(2) = 47$, $C_\varepsilon(3) = 26$, $C_\varepsilon(4) = 21$, $C_\varepsilon(5) = 18$, …, $C_\varepsilon(\infty) = 5$ [бит/с·Гц].

Литература

1. *Орищенко В.И., Санников В.Г., Свириденко В.А.* Сжатие данных в системах сбора и передачи информации. М.: Радио и связь, 1985. 184 с.
2. *Шеннон К.* Работы по теории информации и кибернетике / Пер. с англ. Под ред. Р.Л. Добрушина и О.Б Лупанова с предисловием А.Н. Колмогорова. М.: Изд. ИЛ, 1963. 830 с.
3. *Санников В.Г.* Основы теории систем инфокоммуникаций: учеб. пособие. М.: Горячая линия – Телеком, 2017.
4. *Стратонович Р.Л.* Теория информации. М.: Советское радио, 1975. 424 с.
5. *Колмогоров А.Н.* Теория передачи информации. Сессия АН СССР, 15-20 октября 1956 г. Пленарные заседания. Изв. АН СССР, 1957. С. 66-99.
6. *Величкин А.И.* Передача аналоговых сообщений по цифровым каналам связи. М.: Радио и связь, 1983. 240 с. (Статистическая теория связи. Вып. 19).
7. *Прудников А.П., Брычков Ю.А., Маричев О.И.* Интегралы и ряды. М.: Наука. ГР ФМЛ. 1981. 798 с.
8. *Санников В.Г.* Методы кодирования речевых сигналов: учеб. пособие. М.: Московский технический университет связи и информатики, 2003.
9. *Безруков И.М., Волчков В.П.* Исследование помехоустойчивости цифровой системы связи с канальным прекодером и финитной посимвольной передачей // Телекоммуникации и информационные технологии. 2016, Т.1, №1. С. 145-149. (http://www.srd-mtuci.ru/attachments/article/249/ТИТ-1-2016.pdf).

ОЦЕНКА ДОСТОВЕРНОСТИ ИНСТРУМЕНТАЛЬНОЙ ДИАГНОСТИКИ НАЛИЧИЯ ЗАБОЛЕВАНИЙ ПО ИЗМЕНЕНИЮ РЕЧИ В СЕТЯХ СВЯЗИ

Титова Надежда Дмитриевна
студент группы БСС 1501
nadya.titova23@yandex.ru

Денисова Мария Алексеевна
студент группы БСС 1501
marija.masha1801@yandex.ru

Терехов Алексей Николаевич
МТУСИ, к.т.н., доцент кафедры ОТС
a.n.terekhov@mtuci.ru

Определение вероятности достоверности инструментальной диагностики наличия заболевания, выполненной в данной статье, определяет актуальность результатов исследований. Поскольку, именно вероятность, с которой выполнена инструментальная диагностика заболевания в условиях воздействия мешающих факторов может, например, стать решающим аргументом для своевременного обращения пациента к врачу. При подготовке статьи выполнены исследования, посвященные влиянию различного рода шумов и помех на вероятность инструментальной диагностики наличия заболевания. Предложен вероятностный подход, применимый не только для разработанного метода инструментальной диагностики заболевания, но и для любых других, что позволяет сравнить методы и выбрать лучший. Выработаны рекомендации и предложены реализации, позволяющие определить вероятностный оптимальный порог, обеспечивающий оптимальную вероятность инструментальной диагностики наличия заболевания. Полученные результаты основаны, в том числе, на исследованиях изменений параметров речи, включая разборчивость сложносоставных числительных, и их корреляции с длительностью пауз между их элементами, свойственных для сетей с коммутацией пакетов. Установлено, что для проведения имитационного моделирования предложено учитывать воздействие всех существенных окружающих шумов, субъективных факторов, а также параметров сетей и устройств связи, влияющих на вероятность инструментальной диагностики наличия заболевания.

Ключевые слова: *достоверность; параметры речи; инструментальная диагностика; наличие заболевания; оптимальный порог; мешающие факторы.*

Определение наличия заболевания предложено осуществлять решающим устройством, на вход которого поступает процесс $z(t)$, представляющий собой математического ожидания параметров детерминированных сигналов, накопленных благодаря «закону Яровой» - $s(t)$ и мешающих факторов (субъективных факторов, изменяющих параметры речи; влияние каналов связи; акустических шумов помещения и.т.д. [3, 6-8]) – $x(t)$. К объективным причинам отнесены влияние акустического окружения, специфика функционирования устройств и сетей связи, исследование качества которых рассмотрено в работе [2].

$$z(t) = s(t) + x(t), \quad (1)$$

где $z(t)$ – сигнал на входе решающего устройства; $s(t)$ - текущий речевой сигнал; $x(t)$- мешающие факторы.

Одной из первоочередных задач, решаемых при инструментальной диагностики наличия заболевания, является то, что влияние мешающих факторов может приводить к тому, что вместо истинного решения о том, что у пациента присутствует отклонение - соответствующее наличию заболевания, будет принято неверное решение о том, что пациент здоров и наоборот.

Обозначим принятие решения о наличии заболевания – «1», а о том, что пациент здоров – «0». Определим, что в случае принятия решения об отсутствии заболевания – «0» , а пациент болен – «1», то такая ошибка называется «пропуск сигнала». Вероятность принятия решения об отсутствии заболевания – «0» при наличии заболевания – «1» обозначим $p(0/1)$. Если принято решение о наличии заболевания – «1» , а на самом деле пациент здоров – «0», то данная ошибка называется «ложная тревога». Вероятность принятия решения о болезни – «1» при здоровом пациенте – «0» обозначим – $p(1/0)$.

Рассчитаем зависимость $p(1/0)$ и $p(0/1)$ от выбора порога V принятия решения о том, что пациент болен или здоров. Примем, что закон распределения мешающих факторов $x(t)$, поражающего сигнал, является нормальным. Вероятность $p(1/0)$ равна вероятности $p(z(t)>V)$ и равна вероятности $p(x(t)>V)$, поскольку, если пациент здоров то процесс $z(t)=x(t)$.

Поскольку [1] функция плотности вероятности (ФПВ) помехи $x(t)$ – гауссова с дисперсией σ^2, т.е.:

$$W(x)=\frac{1}{\sigma\sqrt{2\pi}}\,e^{-\frac{x^2}{2\sigma^2}}$$

то искомая вероятность равна:

$$p(1/0)=p(x>V)=\int_{V}^{\infty}\frac{1}{\sigma\sqrt{2\pi}}\,å^{-\frac{x^2}{2\sigma^2}}dx=\frac{1}{\sqrt{2\pi}}\int_{V/\sigma}^{\infty}å^{-\frac{ó^2}{2}}dy=1-\mathbf{F}\left(\frac{V}{\sigma}\right),\qquad(2)$$

$$ãäå\quad \mathbf{F}\left(\frac{V}{\sigma}\right)=\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{V/\sigma}å^{-\frac{ó^2}{2}}dy\quad -èí òåðàë\quad Ëàïëàñà.$$

Вероятность принятия решения о том, что пациент здоров в случае наличия заболевания $p(0/1)=p(z(t)<V)=p(U_c+x(t)<V)=p(x(t)<V-U_c)$ может быть представлена в виде:

$$p(0/1)=p(x<V-U_ñ)=\int_{-\infty}^{V-U_ñ}\frac{1}{\sigma\sqrt{2\pi}}\,å^{-\frac{x^2}{2\sigma^2}}dx=\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\frac{V-U_ñ}{\sigma}}å^{-\frac{ó^2}{2}}dy=\mathbf{F}\left(\frac{V-U_ñ}{\sigma}\right)\qquad(3)$$

Зависимость $p(1/0)$ и $p(0/1)$ от порога принятия решения V приведена на рисунке (кривые 1 и 2, соответственно) для произвольно выбранного отношения $U_c/\sigma=2$.

Полной характеристикой правильности постановки диагноза о наличии или отсутствии заболевания является средняя вероятность ошибки:

$$p=p(1)p(0/1)+p(0)p(1/0),\qquad(4)$$

где $p(1)$, $p(0)$ – априорные вероятности наличия заболевания – 1 или отсутствия заболевания – 0 соответственно, определяемых отклонением математического ожидания речевых параметров от среднего значения. На рисунке 1 представлены зависимости средней вероятности ошибки постановки диагноза для $p(1)=p(0)=0.5$ (кривая 3); и для $p(1)=0.2$, $p(0)=0.8$ (кривая 4).

Величина порога, при которой средняя вероятность ошибки постановки диагноза минимальна, называется оптимальным порогом $V_{опт}$, значение которого можно определить решением уравнения [1]:

$$\frac{dð}{dV}=\frac{d}{dV}\left[ð(0)ð(1/0)+ð(1)ð(0/1)\right]=\frac{d}{dV}\left[ð(0)[1-F(\frac{V}{\sigma})]+ð(1)\,F(\frac{V-U_ñ}{\sigma})\right]=0\qquad(5)$$

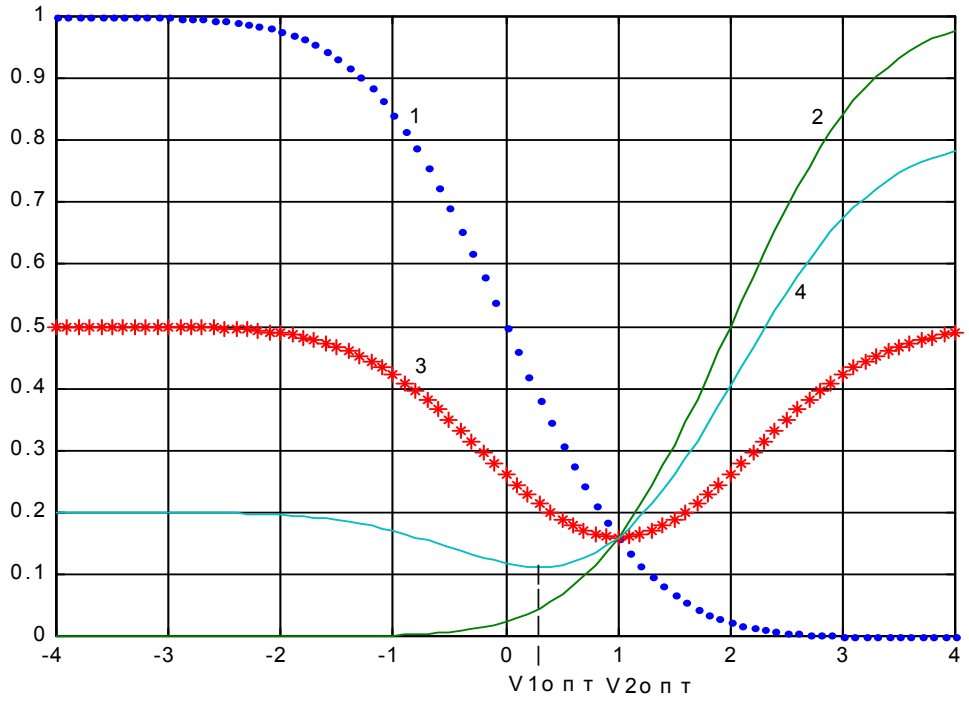


Рис. 1. Зависимость средней вероятности ошибки от априорных вероятностей заболеваний и здорового состояния

Решение этого уравнения для мешающих факторов дает следующее выражение, позволяющее определить оптимальное значения порога:

$$V_{îïò} = \frac{U_ñ}{2} - \frac{\sigma^2}{U_ñ} ln \frac{ð(1)}{ð(0)} \quad (6)$$

На рисунке 1 отмечены оптимальные значения порогов $V1_{опт}$ и $V2_{опт}$ для двух кривых, соответствующих разным значениям априорных вероятностей, определяемых отклонением математического ожидания речевых параметров от среднего значения, наличия или отсутствия заболевания. Для установления соответствия между голосом и речью, зафиксированными на эталонной фонограмме и фонограмме-образце возможно использование функции корреляции [4, 5]. Функция корреляции характеризует степень статистической зависимости двух значений случайного процесса, разделенных интервалом времени τ.

В данном случае представляется возможным рассмотреть модель как один стационарный случайный процесс, разделенный во времени.

На рисунке 2 представлен стандартный вид функции корреляции. Интервал корреляции случайного процесса, характеризует ширину графика функции корреляции, в случае если: $|\tau| \le \tau_k$ – то значения коррелированны; $|\tau| > \tau_k$ – то значения не коррелированны.

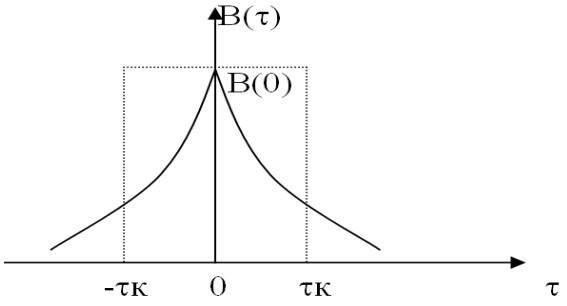


Рис. 2. Стандартный вид функции корреляции

Для эргодического стационарного случайного процесса с нулевым средним значением функция корреляции зависящего только от разности $\tau = t_2 - t_1$, представлено на рис. 2.

Выводы

1. Для дифференцирования наличия или отсутствия заболеваний необходимо использовать предложенный алгоритм выделения оптимального порога, обеспечивающего адаптацию к статической выборки статистических параметров.

2. Снижение ошибок вычислительного характера возможно за счет применения проверенного математического аппарата определения априорной вероятности правильности постановки диагноза о наличии заболеваний.

3. Обеспечит достоверность определения наличия заболевания, не увеличивая необходимый объём вычислительных ресурсов, возможно при использовании корреляционного анализа.

Литература

1. *Сухоруков А.С.* Теория цифровой связи. Учебное пособие. Ч.2/ МТУСИ. М., 2008. 53 с.

2. *Григорьев И.Д., Орлов В.Г.* Исследование качества связи *MANET*-сети на основе *VDL*-4 с использованием механизмов канального уровня // Телекоммуникации и информационные технологии. 2016, №.1. С. 105-110. *(http://www.srd-mtuci.ru/attachments/article/*249*/ТИТ*-1-2017*.pdf)*

3. *Рысин Ю.С., Терехов А.Н.* Алгоритм определения факторов, влияющих на качество восприятия телефонных услуг связи // Электросвязь. № 3. 2016. С. 65-68.

4. *Терехов А.Н., Рысин Ю.С.* Некоторые пороги восприятия запаздывающих акустических сигналов (эхо-сигналов) // T-Comm: Телекоммуникации и транспорт. № 4. 2015. С. 51-53.

5. *Попов О.Б., Рихтер С.Г., Терехов А.Н., Чернышева Т.В.* Методы оценки качества в каналах телерадиовещания. Учебное пособие для вузов. М.: Горячая линия – Телеком, 2016. 232 с.

6. *Рысин Ю.С., Терехов А.Н.* Алгоритм оценки влияния негативных факторов на качество телефонного общения // T-Comm: Телекоммуникации и транспорт. 2012. Т. 6. № 10. С. 96-98.

7. *Венедиктов М.Д., Рысин Ю.С., Терехов А.Н.* Программно аппаратный комплекс для оценки параметров сетей, обеспечивающих телефонные услуги связи // T-Comm: Телекоммуникации и транспорт. 2013. Т. 7. № 8. С. 39-43.

8. *Терехов А.Н.* Метод интегральной оценки качества телефонного общения при модернизации сетей и/или введении перспективных услуг связи // T-Comm: Телекоммуникации и транспорт. 2016. Т. 10. № 4. С. 67-72.

АГЛОРИТМ ЭФФЕКТИВНОГО АРУР ДЛЯ СИСТЕМ МАССОВОГО ОПОВЕЩЕНИЯ

Симаков Никита Андреевич
студент группы МРА1601 МТУСИ
nickita.simakoff@yandex.ru

Абрамов Валентин Александрович
МТУСИ, к.т.н., доцент кафедры ТиЗВ
vabramov44@mail.ru

Приведен алгоритм АРУР для систем массового оповещения. Проведен анализ систем массового оповещения, в которых применяются автоматические регуляторы уровня звукового сигнала с огибающей. Также приведены результаты исследований и сделаны выводы.

Ключевые слова: *авторегулятор уровня звукового сигнала (АРУР), относительная средняя мощность (ОСМ), комплексная огибающая Гильберта, μ- характеристика, аудио процессор АРГО.*

Жизнь современного общества невозможна без массового оповещения о надвигающихся стихийных бедствиях, катастрофах и крупных транспортных и промышленных авариях. Оповещение населения начинается с подачи громких звуковых сигналов, которые представляют собой речевое сообщение, конкретизирующее вид и место опасности.

При звуковом оповещении применяют все известные методы первичной обработки сигналов звукового информационного вещания, повышающее громкость и разборчивость при ограниченных мощностях усилителей и звуковых излучателей:

- повышение с помощью авторегулятора уровня относительной средней мощности;
- динамическая регулировка спектра передаваемого сигнала.

Инерционность АРУР приводит к нежелательным изменениям СЗВ, снижению разборчивости, искажениям тембра, поэтому используется обработка не самого сигнала, а его огибающей сформированной с использованием ортогонального по Гильберту колебания. Использование регулирования по огибающей должно позволить увеличить относительную среднюю мощность сигнала массового оповещения в 5 раз, повысить эффективность звукоизлучающих установок систем массового оповещения, повысить разборчивость, а, следовательно, и дальность передачи сообщения [1, 3-5]. Увеличение дальности пропорционально корню квадратному из отношения средней мощности первоначального сигнала и сигнала с увеличенной относительной средней мощностью (без учёта потерь энергии в атмосфере). Недостатком такой обработки остаются искажения сигналов звукового вещания инерционными устройствами автоматического регулирования.

К недостаткам существующих автоматических регуляторов уровня относятся:

- превышение номинального уровня, а, следовательно, искажения СЗВ на времени срабатывания АРУР, незаметные на слух, но приводящие к появлению помех в каналах при отсутствии пиковых срезов;
- изменение спектральных соотношений, тембра СЗВ за счёт подчеркивания, как правило, слабо уровневых высокочастотных звучаний в речи или музыке;
- ухудшение разборчивости при высоком соотношении сигнал/шум за счет времени изменения коэффициента передачи АРУР, времени срабатывания;
- искажения акустической обстановки, в которой формируется программа, за счёт изменения характера спада эхо-сигнала.

Кроме того, АРУР эффективны при тщательной установке входных уровней СЗВ, что не всегда выполняется в процессе эксплуатации.

Всё это даёт нам возможность определить исходные посылки к организации алгоритма:

1. регулирование не должно вносить изменений в сигнал, сформированный в тракте формирования программ, заметных слушателю;
2. контрасты громкости происходят, как правило, при максимальном уровне сигнала;
3. основное отличие сигналов определяется средней мощностью;
4. используемое регулирование не должно изменять динамику сигнала;
5. регулирование должно привести к повышению средней мощности сигнала, относительной сред-

ней мощности преобладающих в РВ передачах (особенно информационных), высокоуровневых сигналов, т.е. к повышению эффективности передатчиков.

Указанные требования могут быть реализованы при использовании безынерционного неискажающего регулятора гильбертовой огибающей СЗВ, позволяющей изменять пик-фактор сигнала и обеспечивать постоянство пикового выходного уровня практически без нелинейных искажений.

Исходя из требований на отсутствие искажений динамики для сигналов, не достигших пикового уровня, регулировка осуществляется только в случае достижения заданного порога, который может быть взят для работы системы массового оповещения порядка -20 дБ.

В этом случае определяется средняя мощность сигнала и, если пик фактор отличается от заданного, осуществляется путём изменения μ-характеристики, на которую умножается огибающая сигнала вещания. Регулировка осуществляется плавно, градациями не более 0,4 единицы на выборку с интерполяцией промежуточных значений на её длительности. Если в обработанном сигнале пик фактор не достиг заданной величины, μ увеличивается и так далее, спад μ осуществляется аналогичным образом [1].

Далее приводятся основные стадии выполнения алгоритма:

1. АЦП исходного вещательного сигнала в цифровой поток с частотой дискретизации 22050 Гц при 16-ти разрядном линейном преобразовании.
2. Предварительная буферизация сигнала для определения параметров сигнала: пикового значения, среднего значения, пик фактора – для формирования сигнала управления при регулировании и принятия решения согласная/гласная.
3. Формирование ортогонального входному сопряжённого по Гильберту сигнала.
4. Формирование огибающей и мгновенной фазы сигнала.
5. Деление огибающей на НЧ - и ВЧ - огибающие.
6. Регулирование НЧ-огибающей сигнала с использованием компрессирования по μ-характеристике.
7. Ограничение пиковых значений огибающей.
8. Регулирование ВЧ-огибающей пропорционально НЧ - огибающей.
9. Ограничение по суммарной аналитической огибающей;
10. Восстановление сигнала как произведение огибающей на косинус мгновенной фазы;
11. ЦАП выходного сигнала;

Графики, поясняющие работу алгоритма, представлены на рисунках 1-4.

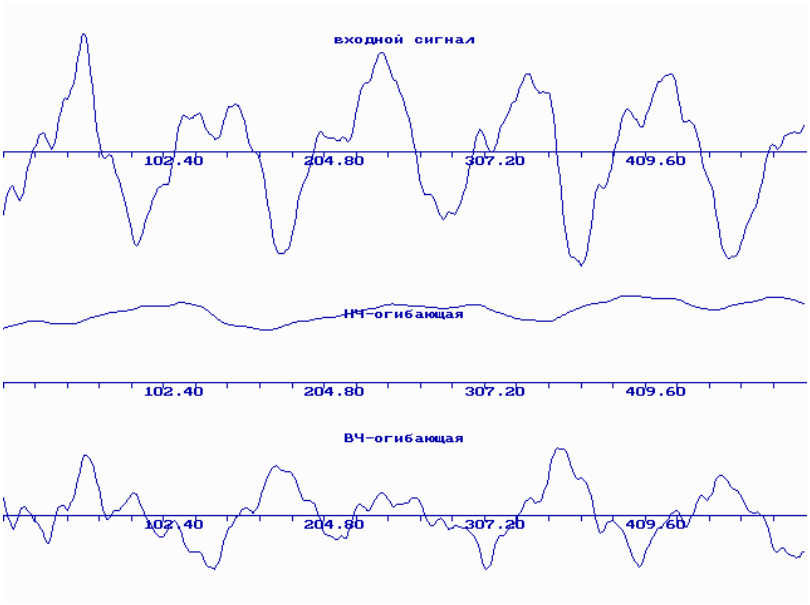


Рис. 1. Входной сигнал, выделенные НЧ и ВЧ-огибающие

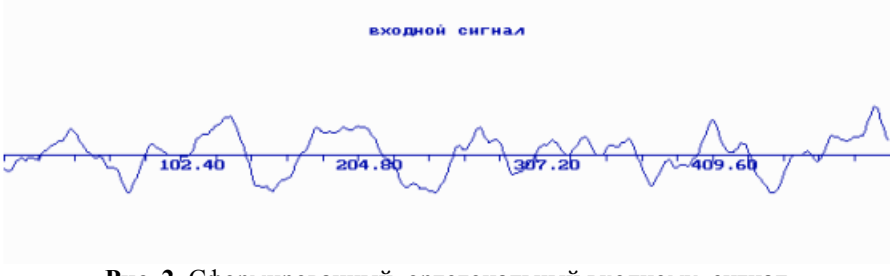


Рис. 2. Сформированный, ортогональный входному, сигнал

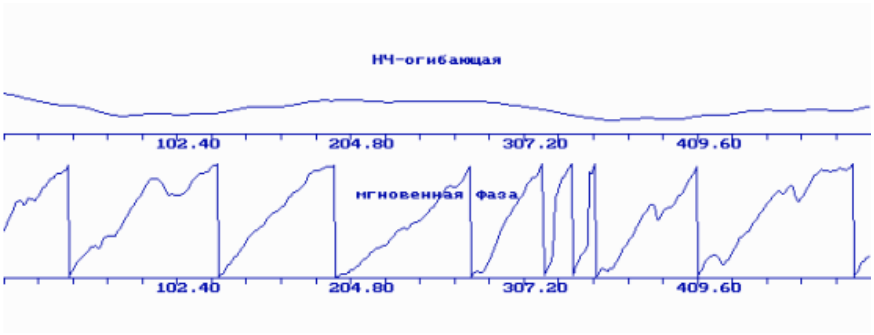


Рис. 3. Сформированная НЧ – огибающая и мгновенная фаза

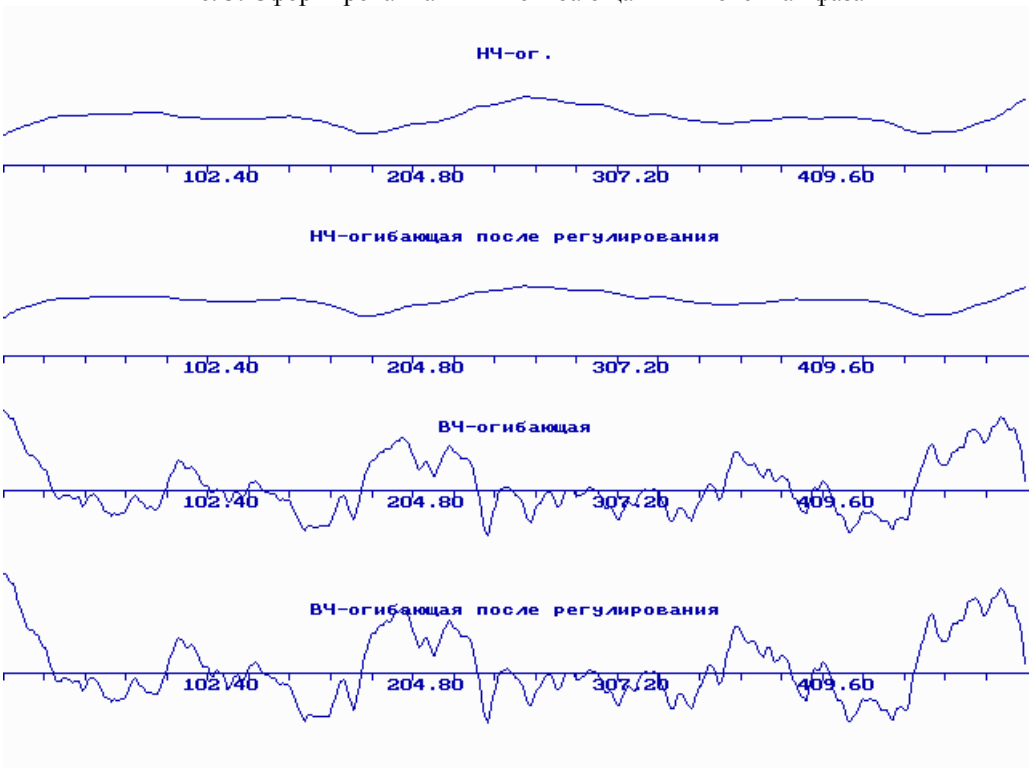


Рис. 4. Регулирование НЧ-огибающей и ВЧ-огибающей, пропорционально НЧ-огибающей

Из входной цифровой последовательности с помощью преобразования Фурье формируют ортогональный сигнал, из которого выделяют огибающую и мгновенную фазу сигнала, огибающую разделяют на низко и высокочастотную составляющие, которые раздельно регулируют под воздействием сигнала управления, после чего огибающую восстанавливают суммированием составляющих и получают выходную цифровую последовательность при перемножении восстановленной огибающей с мгновенной фазой, при этом сигнал управления формируют, с одной стороны, при сравнении цифровых пиковых уровней сигнала с цифровым пороговым уровнем, а, с другой стороны, при сравнении цифровых значений входного и выходного сигнала с другим пороговым цифровым уровнем [2].

Регулирование с использованием μ-характеристики происходит следующим образом. В случае превышения порога пиковым значением входного сигнала - μ изменяется с 0 до 0,2, в дальнейшем значение μ определяется выходным сигналом, в случае превышения пик фактором заданной величины, μ изменяется, например, с 0 до 0,3, а огибающая умножается на μ - характеристику с плавно изменяющимися и линейно-интерполированными промежуточными μ. На следующей выборке опять изменяется пик фактор выходного сигнала и так до тех пор, пока не будет получена заданная величина ОСМ с изменённым допуском, поддерживаемая адаптивно меняющимся в соответствии с параметрами сигнала во времени значениями μ. Спад осуществляется аналогично с учётом увеличения ступеней уменьшения μ для понижения заметности реверберации сигнала [2].

Ниже приведена осциллограмма речевого сигнала на входе и выходе АП (рис. 5) Арго, которая касается эффективности разработанного алгоритма, а также результаты испытаний системы звукоусиления объекта, проведенных с применением аудио процессора АРГО (табл. 1) при указанном (рис. 6) расположении точек.

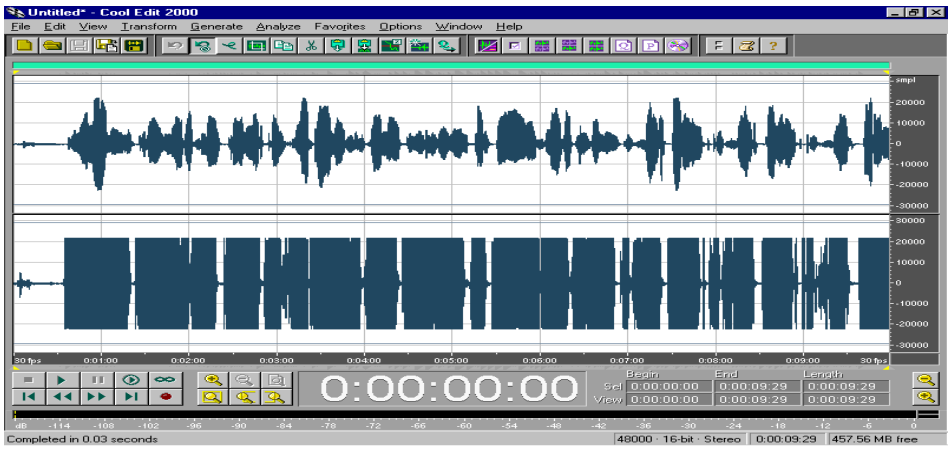


Рис. 5. Осциллограмма речевого сигнала на входе и выходе АП Арго

Таблица 1

Результаты испытаний системы звукоусиления объекта, проведенных с применением аудио процессора АРГО

Измерительная точка (по карте)	Уровень, измеренный шумомером «Брюль»			Энергетический выигрыш обработки, дБ	Оценка словесной разборчивости, %	
	Фоновый шум, дБ	Шум + сигнал необработанный, дБ	Шум + сигнал обработанный, дБ		Необработанный сигнал	Обработанный сигнал
1	70	70	70,2	3,25	25	50
2	70	70	70,3	1,71	~10	~15
3	-	-	-	5,85	30	80
4	55	60	65	4,84	25	75
5	-	-	-	6,23	20	65
6	48	51	55	2,04	~5	~10
7	-	-	-	3,87	15	30
8	46	48	52	2,48	~10	~20
9	68	68	68,5	1,67	~0	~5
10	68	68	68	~0	~0	~0

Рис. 6. Расположение исследуемых точек на местности

Выводы

1. Используя данный алгоритм регулирования уровня сигнала оповещения, мы обеспечиваем перераспределение его громкости при тех же пиковых уровнях сигнала, с общим повышением мощности, при этом сохраняем динамический диапазон по электрическому уровню, а также увеличиваем ОСМ сигнала оповещения в 2-4 раза, повышая таким образом КПД до 25%.
2. Программная реализация такого регулирования с применением новейших методов увеличения ОСМ (БПФ и преобразования Гильберта) не только разумна, но и является наиболее гибким подходом к решению этой проблемы.
3. Разборчивость сигнала оповещения, как и предполагалось, увеличилась при сохранении объективного качества звучания;
4. Регулирование сигнала совершенно незаметно для уха неподготовленного человека, однако сравнительно с необработанным сигналом уровень обработанного заметно выше для того же неподготовленного человека.

Литература

1. *Попов О.Б., Рихтер С.Г.* Цифровая обработка сигналов в трактах звукового вещания. М.: Горячая линия – Телеком», 2007.
2. *Мишенков С.Л., Копылов А.П., Ефимов А.П.* Системы звукового вещания и оповещения. Учебное пособие по курсу «Радиовещание и электроакустика». М.: МТУСИ, 2008.
3. *Абрамов В.А., Попов О.Б., Чернышева Т.В.* Измерение мощности звуковых сигналов вещания на коротких временных интервалах // T-Comm: Телекоммуникации и транспорт. 2012. Т. 6. № 10. С. 9-11.
4. *Абрамов В.А., Попов О.Б., Ождихин Г.М., Рихтер С.Г.* Оценка качества обработки звуковых сигналов в радиовещательных студиях // T-Comm: Телекоммуникации и транспорт. 2013. Т. 7. № 9. С. 6-8.
5. *Абрамов В.А., Попов О.Б., Ождихин Г.М., Черников К.В.* Повышение эффективности регулировании громкости сигналов телерадиовещания // T-Comm: Телекоммуникации и транспорт. 2013. Т. 7. № 9. С. 4-5.

ПРИМЕНЕНИЕ ТЕХНОЛОГИИ РАСПРЕДЕЛЕННОГО УСИЛЕНИЯ ПРИ ПОСТРОЕНИИ ШИРОКОДИАПАЗОННЫХ И СВЕРХШИРОКОДИАПАЗОННЫХ РАДИОЧАСТОТНЫХ УСИЛИТЕЛЕЙ МОЩНОСТИ

Шмаков Никита Дмитриевич
аспирант кафедры РОС МТУСИ
shmaki-shmak@yandex.ru

Иванюшкин Роман Юрьевич
МТУСИ ,доцент кафедры РОС
rivanyushkin@gmail.com

Обсуждаются перспективы и преимущества внедрения усилителей бегущей волны с распределенным усилением на полевых транзисторах, в качестве широкодиапазонных и сверхширокодиапазонных усилителей мощности в диапазонах ВЧ, ОВЧ и УВЧ. Приводятся результаты исследования энергетических и качественных показателей такого УРУ методом компьютерного схемотехнического моделирования.

Ключевые слова: *широкодиапазонный усилитель мощности, усилитель с распределенным усилением, усилитель бегущей волны, искусственная длинная линия,ФНЧ-звено.*

На сегодняшний день, при построении радиопередающих трактов, достаточно велика потребность в неперестраиваемых широкодиапазонных и сверхширокодиапазоных усилителях мощности радиочастоты. Это в первую очередь касается радиопередатчиков для систем, так называемого когнитивного радио, а также радиопередатчиков для наземного телевизионного вещания, работающих в нескольких поддиапазонах ОВЧ-УВЧ [7-12].

Бурное развитие полупроводниковых технологий еще в 1970-е – 1980-е годы, казалось бы, решило проблему построения неперестраиеваемых диапазонных и широкополосных радиочастотных усилителей мощности. Это стало возможным, благодаря тому, что невысокие входные и выходные сопротивления транзисторов на умеренно-высоких частотах слабо шунтируются паразитными емкостями, как самих усилительных приборов, так и монтажа. Разработка широкодиапазонных цепей связи в виде трансформаторов-линий (трансформаторов Рутрофа) позволила строить такие твердотельные усилители мощности на основе двухтактных схем [3]. Это, практически полностью решило проблему построения широкодиапазоных усилителей в диапазоне ВЧ, а также в начале ОВЧ диапазона. Однако, в области более высоких частот построение таких усилителей часто затруднено из-за сложностей реализации трансформаторов Рутрофа с широким перекрытием по частоте в диапазонах ОВЧ-УВЧ.

Решить проблему построения диапазонных усилителей мощности с перекрытием по частоте до декады и более позволяет технология распределенного усиления, ранее широко применявшаяся при построении широкодиапазонных усилителей мощности на электронных лампах [2-4]. Такой усилитель строится путем включения входных и выходных электродов усилительных приборов в звенья искусственных длинных линий, причем, входная и выходная емкости этих приборов всчитываются во входную и в выходную искусственные длинные линии. Если при этом в искусственных длинных линиях обеспечивается режим бегущей волны, то усилитель обладает примерно одинаковыми значениями коэффициента усиления и равномерной выходной мощностью во всем диапазоне частот, в котором выполняется данное условие. При этом, за счет включения входной и выходной емкостей усилительного прибора, а также емкостей монтажа в состав искусственных длинных линий, шунтирования усилительного прибора не происходит, вне зависимости от величин его входного и выходного сопротивлений. Такие усилители, получившие название усилителей с распределенным усилением (УРУ) и являющиеся разновидностью усилителей бегущей волны (УБВ), в свое время широко применялись в качестве широкодиапазонных предварительных и даже оконечных усилителей мощности радиопередатчиков диапазона ВЧ.

Однако, при переходе от ламп к биполярным транзисторам от УРУ быстро отказались. Одной из причин этого является появление технологии построения широкодиапазонных полупроводниковых усилителей мощности на основе двухтактных схем и трансформаторов Рутрофа (см. выше), которые при более простой

реализации полностью закрывают потребности в таких усилителях при построении радиопередающих трактов диапазона ВЧ. С другой стороны, такой хорошо известный недостаток УРУ [2, 3], как низкий коэффициент усиления, практически нивелирует целесообразность построения УРУ на биполярных транзисторах, поскольку их коэффициенты усиления по мощности существенно (в разы) ниже, чем у электронных ламп. Поскольку биполярные транзисторы принципиально не могут работать без токов базы, это существенно усложняет построение входной искусственной длинной линии УРУ.

Совершенно иная ситуация возникает при переходе к современным полевым транзисторам с изолированным затвором технологии MOSFET. Прежде всего, эти транзисторы имеют очень высокие коэффициенты усиления по мощности, не только заведомо большие, чем у биполярных, но и в разы, и даже на порядок, превышающие коэффициенты усиления электронных ламп. Это снова делает актуальной задачу построения УРУ, который сможет обеспечить коэффициент усиления по мощности, превышающий аналогичный показатель УРУ на электронных лампах. В силу своего принципа построения и работы, такие полевые транзисторы работают при нулевом постоянном токе затвора, а также (в следствие очень высокого коэффициента усиления) с достаточно малыми переменными токами во входной цепи. Это обстоятельство позволяет строить входную искусственную длинную линию УРУ примерно по тем же хорошо известным технологиям, которые раньше применялись при построении УРУ на электронных лампах. С другой стороны, входные и выходные емкости современных мощных радиочастотных полевых транзисторов существенно превышают аналогичные емкости биполярных транзисторов и даже электронных ламп. По этой причине, даже при малых входных и выходных сопротивлениях полевых транзисторов, их шунтирование паразитными емкостями по входу и выходу будет наступать при более низких рабочих частотах, по сравнению с биполярными транзисторами. Указанные особенности, а также то, что практическая реализация трансформаторов Рутрофа, обеспечивающих существенное перекрытие по частоте в диапазонах ОЧ и УВЧ, часто затруднена, делает перспективной разработку УРУ на полевых транзисторах в качестве предварительных и предоконечных усилителей мощности радиочастоты широкодиапазонных и сверхширокодиапазонных радиопередающих трактов на данных частотах. Учитывая еще один известный недостаток УРУ – достаточно низкий КПД, строить на основе УРУ оконечные каскады радиопередатчиков, вряд ли можно считать целесообразным, несмотря на имеющийся опыт построения мощных УРУ на электронных лампах.

Как показал опыт расчета и компьютерного схемотехнического моделирования УРУ на полевых транзисторах [5], применять при их проектировании существующие методики, предназначенные для расчета ламповых УРУ [2] (при их адаптации под транзисторное построение) нецелесообразно. Гораздо лучшие результаты дает построение входной и выходной искусственных длинных линий УРУ на основе Т-образных ФНЧ-звеньев 3-го порядка, где, в качестве емкостного элемента, включается входная или выходная емкость транзистора, соответственно. Частота среза такого ФНЧ-звена выбирается несколько большей, по сравнению с верхней частотой рабочего диапазона УРУ. При этом, расчет режимов работы самого транзистора ведется по традиционным методикам [3].

На основе вышеизложенного подхода проведен расчет УРУ на балансных полевых транзисторах типа BLF278 [1] для диапазона рабочих частот от 10 до 110 МГц. Частота среза Т-образных ФНЧ-звеньев была выбрана равной 120 МГц, что обеспечивает необходимый запас. Для получения на выходе УРУ полезной мощности равной 300 Вт, требуется 4 половинки полевых транзисторов BLF278. Таким образом, для получения заданной полезной мощности требуется два балансных полевых транзистора BLF278. По результатам расчета проводилось компьютерное схемотехническое моделирование УРУ в студенческой версии пакета Micro-Cap 10. Смоделированная схема представлена на рис. 1 [5, 6].

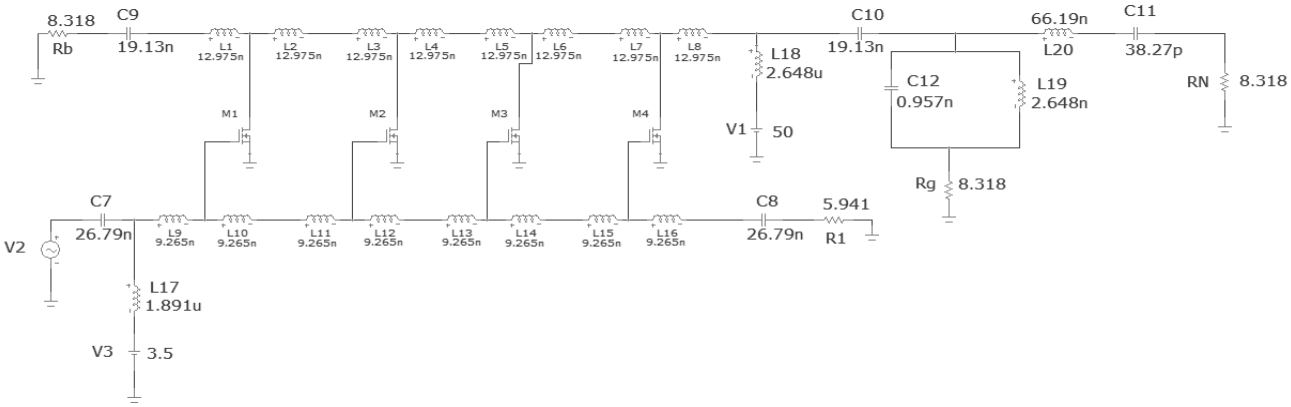


Рис. 1. Смоделированная схема УРУ на полевых транзисторах

В полученной схеме УРУ в качестве выходной цепи применяется диплексер, который (для упрощения компьютерной модели) образован последовательным и параллельным колебательными контурами. Колебательные контуры настроены в резонанс на первую (полезную) гармонику. Данные контуры нагружены на сопротивления, равные волновому сопротивлению выходной искусственной длинной линии. При расчете контуров, их нагруженная добротность принималась равной 5. При проведении исследований энергетических характеристик и диапазонных свойств УРУ осуществлялись перерасчет и перестройка колебательных контуров на заданный ряд рабочих частот в разных участках рабочего диапазона. Транзисторы в полученной схеме УРУ работают в режиме класса АВ.

При помощи созданной на основе проведенных расчетов компьютерной модели исследованы энергетические характеристики УРУ. На рисунке 2 представлена зависимость полезной мощности на выходе УРУ от частоты. Имеющаяся неравномерность выходной мощности по диапазону связана с неполным согласованием искусственных длинных линий, в том числе из-за не полного учета всех особенностей паразитных емкостей транзистора в компьютерной модели.

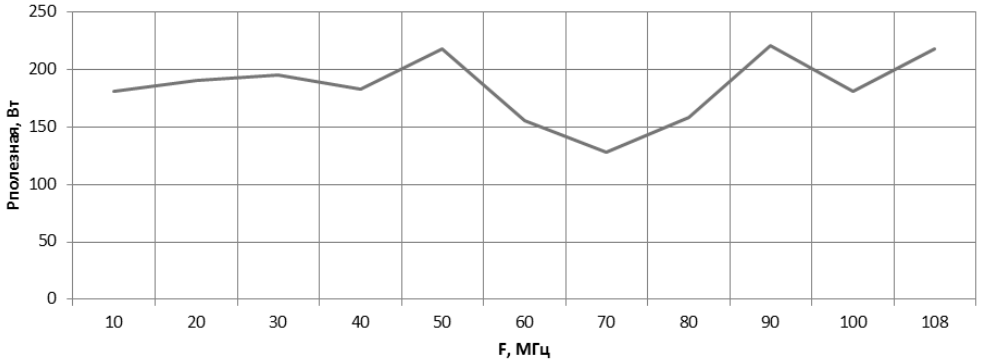


Рис. 2. Зависимость полезной мощности на выходе УРУ от частоты

При помощи созданной компьютерной модели также исследованы амплитудная и фазо-амплитудная характеристики полученного УРУ на рабочей частоте, соответствующей уровню максимальной полезной мощности (90 МГц). Полученные характеристики представлены на рис. 3 и 4, соответственно.

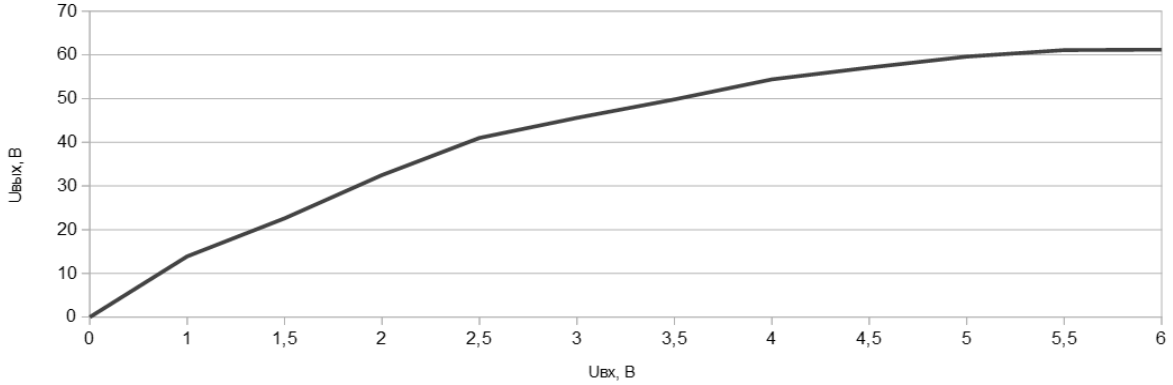


Рис. 3. Амплитудная характеристика УРУ

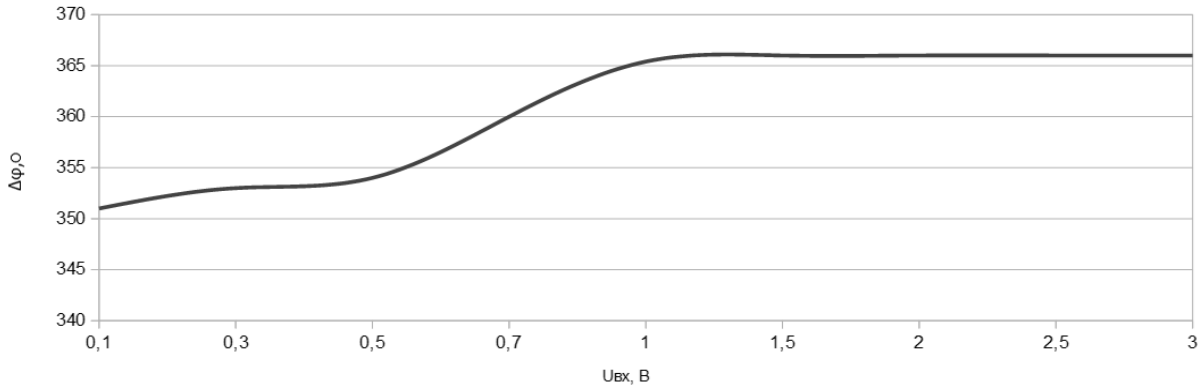


Рис. 4. Фазо-амплитудная характеристика УРУ

Нелинейность амплитудной характеристики обусловлена, как статическими характеристиками самого транзистора (основное назначение которого – усиление ЧМ-сигналов), так и существенной нелинейностью его вольт-фарадных характеристик, в следствие чего, при различных амплитудах усиливаемого радиосигнала, нарушается согласование искусственных длинных линий, которое происходит из-за существенных изменений величин входной и выходной емкостей транзистора. Наличие этой проблемы также подтверждает и приведенная выше фазо-амплитудная характеристика, вид которой свидетельствует о наличии амплитудно-фазовой конверсии (АФК). В случае необходимости использования такого УРУ в качестве усилителя радиосигналов с непостоянной огибающей, потребуется применение мер по его линеаризации, из которых, на сегодняшний день, наиболее эффективным методом является раздельная цифровая предкоррекция амплитудно-амплитудных и фазо-амплитудных искажений.

Искажения смоделированного УРУ подтверждаются также проведенными исследованиями уровней комбинационных составляющих на его выходе двухтоновым методом. Проводилось исследование комбинационных составляющих третьего и пятого порядков для нескольких точек, в пределах рабочего участка амплитудной характеристики. Для проведения двухтонового испытания в модель УРУ были введены два источника входного напряжения. Графики зависимостей уровней комбинационных искажений третьего и пятого порядка от пиковой амплитуды напряжения возбуждения на входе УРУ представлены на рис. 5. и 6.

Полученные уровни комбинационных искажений третьего и пятого порядка не соответствует требованиям, предъявляемым к линейным усилителям, что подтверждает изложенные выше соображения, относительно необходимости применения мер по линеаризации. При усилении большинства современных радиосигналов с цифровыми схемами и видами модуляции задача линеаризации усилителей мощности радиопередающего тракта является одной из наиважнейших, вследствие крайне жестких требований к внеполосным (внеканальным) составляющим излучаемого спектра, а также к искажениям, вызывающим ошибки модуляции передаваемых символов.

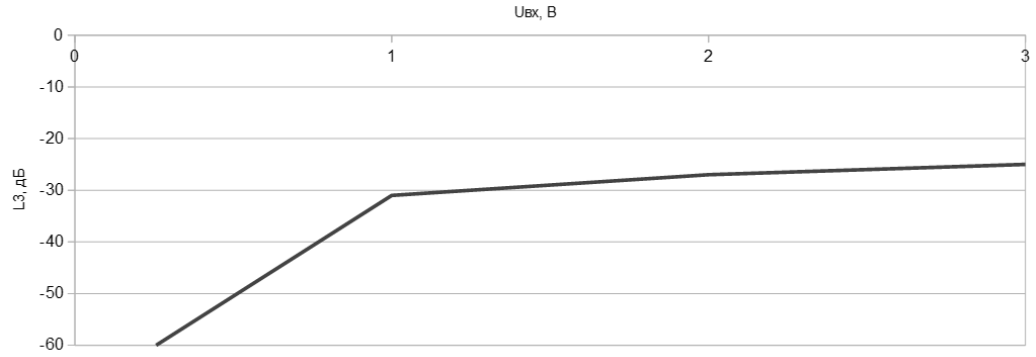


Рис. 5. Уровни комбинационных искажений третьего порядка от амплитуды напряжения возбуждения

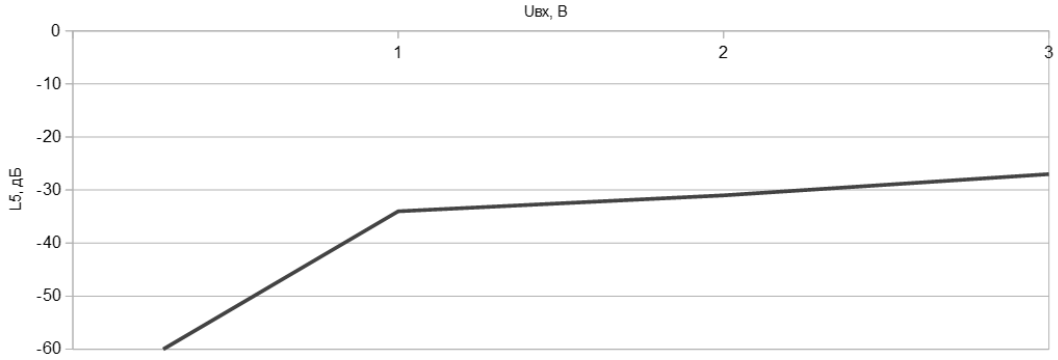


Рис. 6. Уровни комбинационных искажений пятого порядка от амплитуды напряжения возбуждения

Дальнейшие исследования УРУ планируется проводить с применением более совершенных программных пакетов схемотехнического моделирования. Это позволит точнее учитывать вольт-фарадные характеристики транзистора (в первую очередь за счет более корректной модели транзистора), а также решать задачи согласования транзисторов с элементами искусственных длинных линий, в том числе с учетом материалов реальных подложек и ряда других факторов.

Литература

1. Официальный сайт компании ASI. Datasheet BLF 278 [Электронный ресурс]. URL: http://www.advancedsemiconductor.com/transistors/BLF/BLF278.shtml (дата обращения 28.11.2017).

2. Алексеев О.В. Усилители мощности с распределенным усилением. // Ленинград: Издательство «Энергия». Ленинградское отделение, 1968. 244 с.

3. Шахгильдян В.В., Шумилин М.С., Козырев В.Б. и др. Проектирование радиопередатчиков // Учебное пособие для вузов. – М.: Радио и связь, 2000, 4-е изд., с. 177-186.

4. Арыков В.С., Дмитриев В.Д., Коротаев В.М., Шишкин Д.А. GaAs МИС усилителя распределенного усиления [Электронный ресурс]. URL: http://www.micran.ru/sites/micran_ru/data/UserFile/File/Publ/2012/GaAs_MMIC_of_distributed_amplifier.pdf (дата обращения 28.11.2017)

5. Nikita D. Shmakov, Roman Yu. Ivanyushkin. Research of solid-state amplifiers of traveling wave of the VHF band using two approaches of their calculation / IEEE Conference Publication: 2017 Systems of Signal Synchronization, Generating and Processing in Telecommunication (SYNCHROINFO).

6. *Шмаков Н.Д., Иванюшкин Р.Ю.* Компьютерное моделирование усилителя бегущей волны диапазона ОВЧ, построенного на ФНЧ звеньях 3-его порядка / Международная научно-техническая конференция "Перспективные технологии в средствах передачи информации – ПТСПИ-2017». Сборник трудов. Суздаль, 2017. Т. 3. С. 80-83.

7. *Иванюшкин Р.Ю., Юрьев О.А.* Способы построения передатчиков цифрового радиовещания диапазона ОВЧ // Системы синхронизации, формирования и обработки сигналов. 2016. Т. 7. № 1. С. 27-29.

8. *Иванюшкин Р.Ю., Юрьев О.А.* Перспективы применения ключевых усилителей мощности классов D и DE при построении радиовещательных передатчиков диапазона ОВЧ // T-Comm: Телекоммуникации и транспорт. 2016. Т. 10. № 5. С. 21-26.

9. *Дулов И.В., Иванюшкин Р.Ю.* Нелинейная АРР по питанию для усилителя мощности передатчика цифрового радиовещания // T-Comm: Телекоммуникации и транспорт. 2012. Т. 6. № 9. С. 59-63.

10. *Иванюшкин Р.Ю., Юрьев О.А.* Проблематика построения РЧ-тракта передатчиков цифрового радиовещания диапазона ОВЧ на основе метода Л. Кана // T-Comm: Телекоммуникации и транспорт. 2013. Т. 7. № 9. С. 91-93.

11. *Иванюшкин Р.Ю., Дулов И.В., Овчинникова М.В., Тришина Ю.А*. История и перспективы применения метода автоматической регулировки режима для повышения КПД радиопередатчиков // T-Comm: Телекоммуникации и транспорт. 2012. Т. 6. № 9. С. 66-67.

12. *Варламов О.В., Чугунов И.В.* Исследование энергетических характеристик цифрового усилителя мощности OFDM сигналов диапазона УВЧ с дельта-сигма модулятором // Наукоемкие технологии в космических исследованиях Земли. 2015. Т. 7. № 2. С. 30-33.

ОЦЕНКА ВРЕМЕННЫХ ХАРАКТЕРИСТИК ВЫПОЛНЕНИЯ ЗАДАЧ РЕАЛЬНОГО ВРЕМЕНИ НА ПЛАТЕ ARDUINO UNO

Безумнов Данил Николаевич
аспирант 1-го года МТУСИ
danbez@yandex.ru

Воронова Лилия Ивановна
МТУСИ, д.ф.-м.н., проф., зав. кафедрой ИСУиА
voronova.lilia@yandex.ru

Рассмотрены критерии, позволяющие отнести задачу к классам задач жёсткого, мягкого и твёрдого реального времени, а также временные характеристики задач реального времени. Описана методика измерения временных характеристик процесса выполнения задачи. Проведены измерения длительности выполнения наиболее часто используемых встроенных функций языка С для программирования плат Arduino. Проведены теоретическая оценка и экспериментальное измерение длительности процесса выполнения задачи для системы поддержания заданного уровня освещённости на рабочем месте. Определены критерии для минимального значения допустимого времени завершения программы.

Ключевые слова*: система реального времени, задача реального времени, временные характеристики, микроконтроллер, Arduino.*

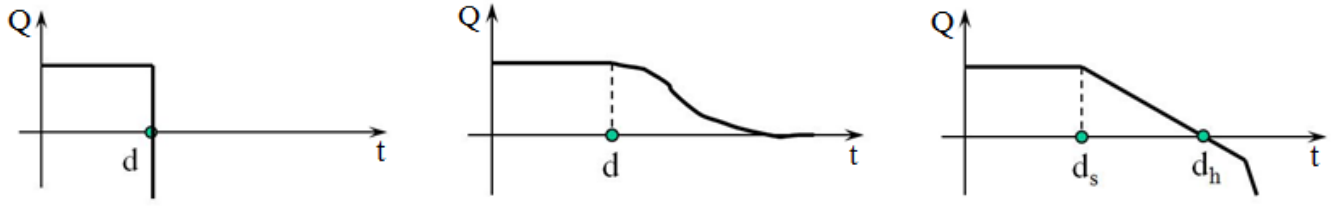
Требования к системам, решающим задачи реального времени

Системой реального времени (СРВ) называется система, которая способна обеспечить требуемый уровень сервиса в заданный промежуток времени. СРВ должна реагировать в предсказуемое время на непредсказуемое появление внешних событий. Под реальным временем понимается время, в течение которого протекает процесс.

Поскольку наряду с задачами реального времени система решает т.н. задачи «нереального времени» (*non-real-time*), задержка в выполнении которых допустима и некритична, то в дальнейшем будем считать, что система принадлежит к классу СРВ, если она способна выполнять задачи реального времени без отказов.

Отказом СРВ называется ситуация, при которой не выполняются требования по времени, т.е. не выполняется дедлайн – допустимое время завершения задачи.

По строгости требований на выполнения дедлайна задачи относят к классам жесткого, мягкого и твердого реального времени (см. рис. 1, [1]).



а) б) в)

Рис. 1. Требования к времени выполнения задач: а) жёсткого, б) мягкого, в) твёрдого реального времени, где: Q – «значимость» результатов выполнения задачи, t – время выполнения задачи, d – дедлайн

В задачах жёсткого реального времени (*hard real-time*) нарушения дедлайна приводит к аварийным последствиям и является недопустимым (см. рисунок 1а).

В задачах мягкого реального времени (*soft real-time*) нарушения дедлайна допускается, если такое событие происходит «не слишком часто» и дедлайн превышается на «небольшой промежуток времени» (см. рис. 1б).

В задачах твёрдого реального времени (*firm real-time*) определены два значения дедлайна: d_s – ограничение мягкого реального времени, d_h – ограничение жёсткого реального времени, при нарушении которых

задачу относят к задачам мягкого и жёсткого реального времени соответственно (см. рис. 1в).

Выполнение задач реального времени определяется набором временных характеристик:

- длительность выполнения t_p (*processing time*);
- период активизации *p* (*period*);
- допустимое время завершения *d* (*deadline*);
- время выполнения в наихудшем случае *wcet* (*worst case execution time*);
- приоритет *pr* (*priority*);
- время реакции t_r (*response time*);
- задержка выполнения процедуры обработки события, или просто задержка *l* (*latency*);

Измерение временных характеристик процесса выполнения задачи

В МТУСИ на кафедре «Интеллектуальные системы в управлении и автоматизации» [2, 3] для реализации лабораторных работ по дисциплинам «Микропроцессоры в системах управления», «Технические средства автоматизации», «Системы реального времени», «Киберфизические системы и интернет вещей» и др. используются платы Arduino Uno на основе микроконтроллера ATmega328. Платы позволяют создавать системы бытовой и промышленной автоматизации различной степени сложности, являясь для обучающихся наглядным синтезом электроники и программирования.

В процессе подготовки лабораторных практикумов по вышеперечисленным дисциплинам автору пришлось решить задачу оценки критериев для временных характеристик Arduino Uno при выполнении задач реального времени.

Для этого было необходимо измерить время отклика системы – время, в течение которого система соберёт и обработает данные с датчиков и сенсоров и выработает управляющие сигналы для исполнительных устройств.

Код программы на Arduino должен содержать две базовых функции: *setup()* – действия, которые выполняются единовременно при запуске программы; *loop()* – действия, которые выполняются циклично, пока на Arduino поступает электропитание. Для проверки способности Arduino решать задачи реального времени достаточно оценить время выполнения одной итерации цикличной функции *loop()* [4].

Время выполнения t_{loop} циклической функции *loop()*, содержащей последовательность из *n* встроенных функций, равно алгебраической сумме времени выполнения каждой из *n* этих функций:

$$t_{loop} = \sum_{i=1}^{n} t_{fi}, \quad (1)$$

Длительность выполнения была измерена для наиболее часто используемых встроенных функций языка C для программирования плат Arduino [3]: передача данных по последовательному соединению, чтение и запись аналогового и цифрового сигналов, арифметические операции.

Для измерения времени выполнения каждой функции была использована функция *micros()*, которая возвращает количество микросекунд, прошедших с начала запуска программы [5]. В переменные типа *unsigned long* записывались значения функции *micros()* до и после вызова каждой из исследуемых функций. Разность этих значений и составляет время, в течение которого выполнялась функция [5]. В таблице 1 приведено время выполнения вышеперечисленных функций.

С целью проверки способности платы Arduino Uno решать задачи реального времени были измерены временные характеристики упрощённой модели системы поддержания заданного уровня освещённости на рабочем месте (см. рисунок 2).

Таблица 1

Время выполнения наиболее часто используемых встроенных функций Arduino

№ п/п	Исследуемая функция	Описание функции	Время выполнения, мкс
1.	*Serial.println()*	Передача данных по последовательному соединению	72
2.	*digitalRead()*	Чтение цифрового сигнала	2084
3.	*digitalWrite()*	Запись цифрового сигнала	2048
4.	*analogRead()*	Чтение аналогового сигнала посредством встроенного АЦП	2192
5.	*analogWrite()*	Запись аналогового сигнала посредством широтно-импульсной модуляции	2086
6.	+ –	Сложение, вычитание	4
7.	* /	Умножение, деление	4

Система имеет следующий алгоритм работы:

1. Уровень освещённости на рабочем месте воздействует на фоторезистор *R*1, подключённый в схему делителя напряжения вместе с резистором *R*2. Напряжение с выхода фоторезистора подаётся на аналоговый вход *A*0 10-разрядного АЦП, на выходе которого формируются эквивалентные значения цифрового сигнала, принимающего значения от 0 до 1023.

2. В программе задаётся требуемое значение уровня освещённости. В данном случае задано значение 400. Как только уровень освещённости становится ниже заданного, программа подаёт на светодиод *LED*1 аналоговый сигнал, задающий такую яркость свечения светодиода, которая повышает уровень освещённости до заданного.

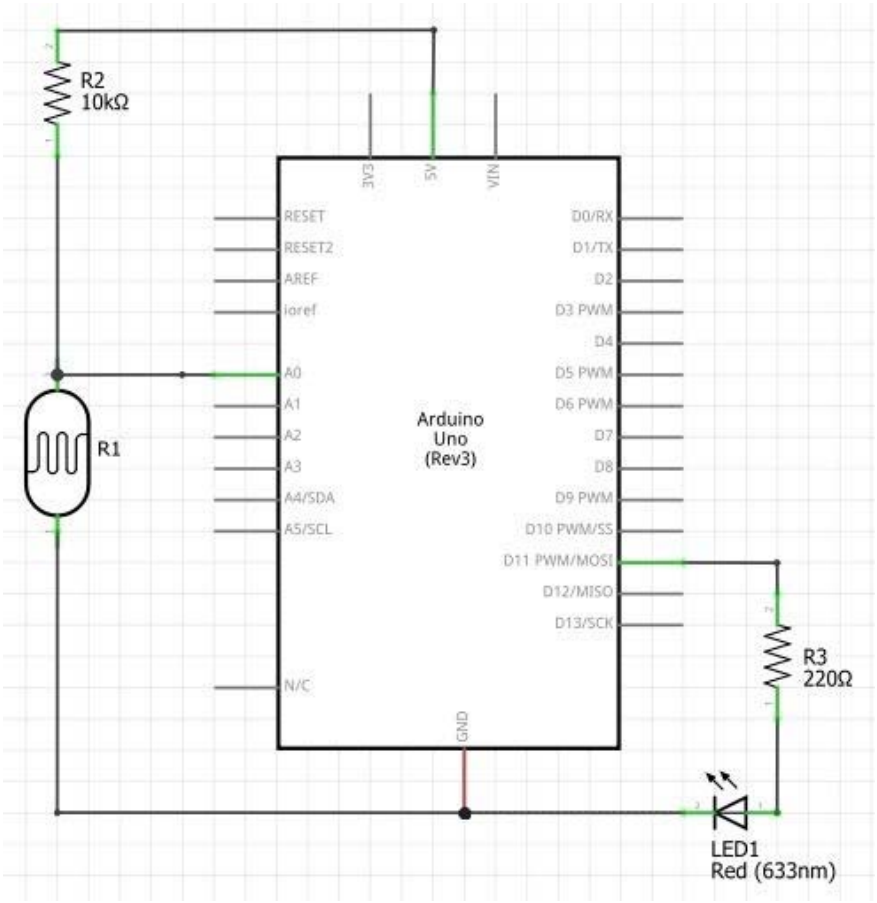


Рис. 2. Принципиальная схема системы поддержания заданного уровня освещённости на рабочем месте

Программа, реализующая поддержание заданного уровня освещённости на рабочем месте, содержит одну функцию чтения аналогового сигнала, одну функцию сложения, две функции умножения и одну функцию записи аналогового сигнала.

В соответствии с (1) время одного выполнения циклической функции *loop()* составит

$t_{loop} = 2192 + 4 + 2 * 4 + 2086 = 4290$ мкс.

Диаграмма выполнения задачи поддержания заданного уровня освещённости изображена на рис. 3.

В данной задаче время выполнения программы t_p вычисляется по формуле (1) и равно 4290 мкс. Задержка l=0, поскольку каждый новый цикл *loop()* выполняется сразу по окончании предыдущего цикла. Следовательно, период активизации p и время реакции t_r также равны 4290 мкс.

В результате измерения реального времени выполнения циклической функции *loop()* описанным выше способом, когда в начале и в конце функции записывалось значении функции *micros()*, а затем находилась их разность, среднее значение t_{loop} составило 4466 мкс. Отклонение между результатами оценки и эксперимента составило 4,1 %, что можно объяснить затратами процессорного времени на выполнение самой функции *micros()*.

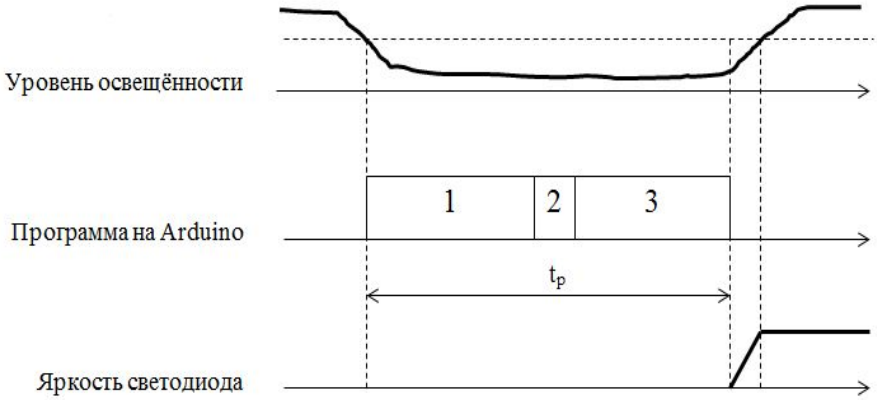


Рис. 3. Диаграмма выполнения задачи поддержания заданного уровня освещённости:
1 – процесс считывания аналогового сигнала; 2 – процесс обработки результатов;
3 – процесс записи аналогового сигнала

Минимальные критерии системы для решения задач реального времени

Задача поддержания заданного уровня освещённости может быть выполнена на Arduino в режиме жёсткого реального времени, если дедлайн *d*, при котором система будет успевать реагировать на мгновенное изменение уровня освещённости на рабочем месте, будет удовлетворять условию

$$d \geq t_R + t_p + t_{LED}, \quad (2)$$

где t_R – постоянная времени фоторезистора; t_p – длительность выполнения программы на Arduino; t_{LED} – время, за которое яркость фотодиода примет требуемый уровень.

Если система обрабатывает не одно, а *n* событий, происходящих с периодом p_n, то для функций, обрабатывающих эти события, также необходимо вычислить длительность выполнения t_{pn}, и тогда условие выполнения задач в реальном времени будет выглядеть:

$$\sum_{i=1}^{n} \frac{t_{pi}}{p_i} \leq 1 \quad (3)$$

При выполнении условия (3) система сможет обрабатывать все *n* задач в режиме жёсткого реального времени.

Выводы

Проведённый анализ позволяет рекомендовать использование плат Arduino Uno при изучении систем реального времени. В зависимости от предъявляемых к задаче требований на Arduino могут решаться задачи жёсткого, мягкого и твёрдого реального времени при некоторых ограничениях, предъявляемых к допустимому времени завершения (дедлайну).

Предложенный метод измерения длительности выполнения отдельных функций и всей программы в целом даёт обучающимся возможность более полно погружаться в процесс разработки и отладки систем реального времени.

Литература

1. *Линец Г.И.* Лекции по системам реального времени [Электронный ресурс]. – Режим доступа: url: http://www.studmed.ru/view/linec-gi-lekcii-po-sistemam-realnogo-vremeni_7f022715470.html, дата обращения: 28.11.2017.

2. *Воронов В.И., Воронова Л.И.* О повышении результативности магистерских программах в условиях инновационной экономики // Инновационные подходы в науке и образовании: теория, методология, практика. Монография. Под общей редакцией Г.Ю. Гуляева. Пенза: Наука и Просвещение, 2017. С. 35-44.

3. *Толмачев Р.В., Воронова Л.И.* Разработка приложения для контент-анализа интернет-публикаций // Телекоммуникации и информационные технологии. 2016. Том 3. № 3. С. 104-107.

4. *Соммер У.* Программирование микроконтроллерных плат Arduino/Freeduino. СПб.: БХВ-Петербург, 2012. 256 с. (Электроника).

5. Ускоряем работу Arduino [Электронный ресурс] – Режим доступа: url: http://cyber-place.ru/showthread.php?t=550, дата обращения: 16.10.2017.

РЕАЛИЗАЦИЯ ОПЕРАЦИИ УМНОЖЕНИЯ В ПОЛЕ MOD(X^N+K) В N-РАЗРЯДНОЙ СЕТКЕ

Мирошниченко Антон Валерьевич
студент группы БЗС1401, МТУСИ
Mirosh.A.V@yandex.ru

Шаврин Сергей Сергеевич
МТУСИ, д.т.н., и.о. декана ф-та СиСС
sss@mtuci.ru

Рассмотрен алгоритм вычисления различных математических операций в поле по модулю (X^n+k), в условиях вычисления в разрядной сетке, состоящей из n разрядов. Получены выражения для умножения по модулю (X^n+k) в разрядной сетке, состоящей из n разрядов, для разных систем счисления. Полученные выражения применены для разработки программы умножения алгоритма IDEA для 16-разрядных процессоров.

Ключевые слова: *алгоритм шифрования IDEA, разрядная сетка, переполнение разрядной сетки, умножение по модулю x^n+k, умножение по модулю x^n+1, умножение по модулю 65537.*

Введение

Использование «классических» арифметических (математических) операций при цифровой обработке сигналов, представленных в целочисленном формате, вызывает расширение разрядной сетки результата операции по отношению к размеру разрядной сетки операндов. Количественно расширение разрядной сетки может быть оценено в соответствии со следующими известными правилами.

При сложении двух чисел, каждое из которых требует для представления N разрядов (символов в данной системе счисления), результат для представления без потери точности требует $N+1$ – разрядной сетки.

Накопление k чисел, представленных в системе счисления по основанию m, каждое из которых занимает N разрядов, для представления без потери точности требует разрядной сетки длиной $N+\log_2 k$ разрядов.

Умножение двух чисел по N разрядов может быть выполнено без потери точности в пределах разрядной сетки длиной $2N$ разрядов, а возведение в степень V, эквивалентное V -кратному умножению, потребует VN разрядной сетки (при знаковом представлении операндов знак обрабатывается отдельно и представляет один дополнительный разряд).

На рисунке 1 представлена схема расположения разрядов в расширенной разрядной сетке при умножении двух беззнаковых чисел, где число BBBB – младшие разряды произведения, а AAAA – старшие.

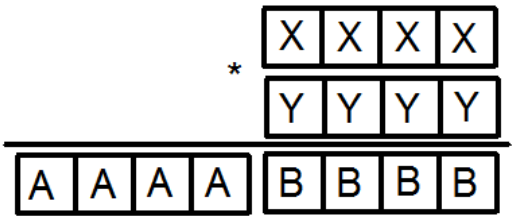


Рис. 1

Расширение разрядной сетки для некоторых приложений цифровой обработки может оказаться критическим. Одним из таких приложений, не допускающим потери «лишних» разрядов результата, является криптография.

Нетрудно убедиться, что при умножении по модулю, кратному степени основания системы счисления, расширения разрядной сетки не происходит, при переполнении разрядной сетки все вышедшие за ее пределы разряды “исчезают”. На рисунке 2 представлено вычисление $98 \cdot 51$ в разрядной сетке, состоящей из 2 разрядов в десятичной системе счисления.

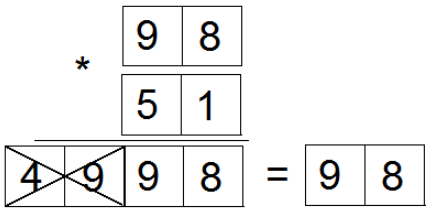


Рис. 2

Из рисунка 2 видно, что старшие разряды 4 и 9 “исчезли”. Математически данную особенность разрядной сетки можно представить, как:

$$(ab) \bmod (X^n), \quad (1)$$

где a, b – некие математические операнды (числа), X – основание системы счисления, n – номер старшего разряда, увеличенный на 1.

Пример на рисунке 2 можно представить через (1) как:

$$(98 \cdot 51) \bmod (10^2) = (98 \cdot 51) \bmod (100) = 98$$

Одним из наиболее широко используемых методов сохранения размера разрядной сетки в процессе цифровой обработки без потери информативности является реализация математических операций в полях Галуа [1, 2]. Механизм сохранения размера разрядной сетки реализуется в полях Галуа операцией mod n – вычислением остатка от деления на простое число (в простых полях) или неприводимый полином (в расширенных).

В технике цифровой обработки сигналов операция mod n не относится к простым; ее сложность обычно значительно превосходит сложность той математической операции, результат которой она корректирует. Стремление разработчиков тем или иным способом упростить реализацию операции mod n, таким образом, можно считать естественным.

Вычисление по модулю X^n+k

Стремлением упростить процедуру вычисления произведения и при этом остаться в исходной разрядной сетке и продиктовано, по-видимому, использование операции $\bmod(2^N+1)$ в алгоритме криптографической защиты IDEA [3, 4, 5]. Поле остатков mod 32769 содержит 32769 элементов от 0 до 32768, их размещение в пределах исходной 16-разрядной сетки не представляется возможным. Однако, поскольку основным назначением операции умножения в рамках алгоритма IDEA является перемешивание битов за счет межразрядных переносов, число 0 (ноль) может быть исключено, т.к. умножение на него ведет к потере информации. Таким образом, в рамках IDEA число 0 интерпретируется как 32768, а умножение на нулевой операнд реализуется специальным обходом.

Рассмотрим математическую операцию вычислению по модулю X^n+k

$$a \bmod (X^n + k) = a - h(X^n + k) = a - hX^n - hk, \quad (2)$$

где $h = \left\lfloor \frac{a}{X^n + k} \right\rfloor$, $\lfloor \ \rfloor$ – оператор округления вниз.

Рассмотрим два выражения $h_0 = \left\lfloor \frac{a}{X^n} \right\rfloor$ и $h = \left\lfloor \frac{a}{X^n + k} \right\rfloor$. Нетрудно убедиться, что h_0 равен старшим разрядам результата:

$$a = a_{n+l} X^{n+l} + \ldots + a_n X^n + \ldots + a_1 X^1 + a_1 X^0;$$

$$h_0 = \left\lfloor \frac{a}{X^n} \right\rfloor \Rightarrow a_{n+l} X^{n+l-n} + \ldots + a_n X^{n-n} = a_{n+l} X^l + \ldots + a_n$$

Вычисление h затруднительно, однако если наложить условие $a < \Theta$, тогда h может принять два значения: h_0 или $h_0 - 1$. В таблице 1 и 2 приведены значения Θ для различных параметров k.

Таблица 1

Значения параметра Θ в 10-ричной системе счисления

$X^n = 10$	Θ	$X^n = 100$	Θ	$X^n = 1000$	Θ
$k = 1$	120	$k = 1$	10200	$k = 1$	1002000
$k = 2$	70	$k = 2$	5200	$k = 2$	502000
$k = 5$	40	$k = 5$	2200	$k = 5$	202000
–	–	$k = 10$	1200	$k = 10$	102000
–	–	$k = 20$	700	$k = 20$	12000
–	–	$k = 50$	400	$k = 50$	22000
–	–	–	–	$k = 100$	12000

Таблица 2

Значения параметра Θ в 2-ичной системе счисления

$X^n = 2^4$	Θ	$X^n = 2^8$	Θ	$X^n = 2^{16}$	Θ
$k = 1$	$2^8 + 2^5$	$k = 1$	$2^{16} + 2^9$	$k = 1$	$2^{32} + 2^{17}$
$k = 2$	$2^7 + 2^5$	$k = 2$	$2^{15} + 2^9$	$k = 2$	$2^{31} + 2^{17}$
$k = 2^2$	$2^6 + 2^5$	$k = 2^2$	$2^{14} + 2^9$	$k = 2^2$	$2^{30} + 2^{17}$
$k = 2^3$	2^6	$k = 2^3$	$2^{13} + 2^9$	$k = 2^3$	$2^{29} + 2^{17}$
–	–	$k = 2^6$	$2^{10} + 2^9$	$k = 2^6$	$2^{26} + 2^{17}$
–	–	–	–	$k = 2^{10}$	$2^{22} + 2^{17}$
–	–	–	–	$k = 2^{15}$	2^{18}

Выбор значения происходит согласно правилу:

$$h = \begin{cases} h_0 - 1 = Z_{стар.} - 1 & , \textit{если } Z_{млад.} < k \cdot Z_{стар.} \\ h_0 = Z_{стар.} & , \textit{если } Z_{млад.} \geq k \cdot Z_{стар.} \end{cases}$$

где $Z_{млад.}$ и $Z_{стар.}$ – старшие и младшие разряды произведения чисел a и b.

Проверить верность представленного вывода можно с помощью эксперимента: пусть $k = 2$, $X^n = 10^2$, тогда $h = \lfloor a / 102 \rfloor$:

$a = 405 \Rightarrow h = \lfloor 405/102 \rfloor = 3$; $a = 406 \Rightarrow h = \lfloor 406/102 \rfloor = 3$;

$a = 407 \Rightarrow h = \lfloor 407/102 \rfloor = 3$; $a = 408 \Rightarrow h = \lfloor 408/102 \rfloor = 4$;

$a = 409 \Rightarrow h = \lfloor 409/102 \rfloor = 4$; $a = 410 \Rightarrow h = \lfloor 410/102 \rfloor = 4$;

Применим (1) к выражению (2):

$$(a - hX^n - h) \bmod (X^n) = (a \bmod (X^n) - (hX^n) \bmod (X^n) - h \bmod (X^n)) \bmod (X^n) =$$

$$= (Z_{млад.} - 0 - kh \bmod (X^n)) \bmod (X^n) = (Z_{млад.} - 0 - kh \bmod (X^n)) \bmod (X^n) =$$

$$= (Z_{млад.} - kh) \bmod (X^n) \qquad (3)$$

Введем переменную:

$$r = \begin{cases} 1, & \textit{если } Z_{млад.} < k \cdot Z_{стар.} \\ 0, & \textit{если } Z_{млад.} \geq k \cdot Z_{стар.} \end{cases}$$

Тогда (3) можно представить как:

$$(Z_{млад.} - kh) \bmod (X^n) = (Z_{млад.} - k(Z_{стар.} - r)) \bmod (X^n) =$$

$$= (Z_{млад.} - kZ_{стар.} + rk) \bmod (X^n) \qquad (4)$$

Замечание: если не вносить ограничение в виде Θ, тогда h и, как следствие, r будут принимать уже не 2, а большее число значений. В данной работе этот момент не будет рассмотрен, так как требует расширенного изучения.

Подведем итог: реализации операции умножения в поле по модулю $X^n + k$ на 16- разрядных процессорах сводится к вычислению: $Z_{млад.} - kZ_{стар.} + rk$ (все вычисления производятся без учета бита переноса и выключенном режиме переполнения)

Замечание: строго говоря, рассмотренный алгоритм не реализует математическую операцию $a \bmod(X^n + k)$, а реализует $(a \bmod(X^n + k)) \bmod(X^n)$ - которая позволяет уместить результат в разрядной сетке.

Примеры умножения по модулю X^n+k

Для более наглядного представления операции $ab \bmod(X^n + k)$ приведем несколько примеров.

В 10-ричной системе счисления:

$$11 \cdot 48 \bmod(10^2 + 1) = 528 \bmod(10^2 + 1) = \begin{cases} 28 - 1 \cdot 5 - 0 \cdot 1 = 23 \\ 528 - 101 \cdot 5 = 528 - 505 = 23 \end{cases}$$

$$11 \cdot 48 \bmod(10^2 + 5) = 528 \bmod(10^2 + 8) = \begin{cases} 28 - 8 \cdot 5 + 1 \cdot 8 = -4 = 96 \\ 528 - 108 \cdot 4 = 528 - 432 = 96 \end{cases}$$

В 2-ичной системе счисления:

$$1011 \cdot 1101 \bmod(2^4 + 1) = 10001111 \bmod(2^4 + 1) = \begin{cases} 1111 - 1000 + 0000 = 0111 \\ 10001111 - 10001 \cdot 1000 = \\ = 10001111 - 10001000 = 0111 \end{cases}$$

В 5-ричной системе счисления:

$$431 \cdot 121 \bmod(5^3 + 1) = 113201 \bmod(5^3 + 1) = \begin{cases} 201 - 113 + 000 = 33 \\ 113201 - 1001 \cdot 113 = 113201 - 113113 = 33 \end{cases}$$

Как можно видеть, данный алгоритм дает верный результат и допускает использование в любой системе счисления.

Умножение в алгоритме шифрования IDEA

В алгоритме IDEA операция умножения $(ab) \bmod(2^{16} + 1)$ обладает интересной особенностью. Перед выполнением операции умножения операнд, имеющий нулевое значение, заменяется на 2^{16}, а после умножения, если его результат равен 2^{16}, то он заменяется обратно на 0. Таким образом, расширение разрядной сетки происходит только на время умножения, а результат возвращается в исходную разрядную сетку.

Согласно (4), алгоритм умножения выполняет функцию возвращения в разрядную сетку. Однако он не предусматривает исходной замены 0 на 2^{16}. Что создает дополнительное исключение при умножении.

Рассчитаем произведение различных чисел с числом 2^{16} по модулю $2^{16} + 1$.

$(2^{16} \cdot 1) \bmod(2^{16} + 1) = 65536 \Rightarrow 0$

$(2^{16} \cdot 2) \bmod(2^{16} + 1) = 65535$

$(2^{16} \cdot 3) \bmod(2^{16} + 1) = 65534$

$(2^{16} \cdot 4) \bmod(2^{16} + 1) = 65533$

.....

$(2^{16} \cdot 2^{16}) \bmod(2^{16} + 1) = 1$

Из представленного расчета можно вывести закономерность:

$(2^{16} \cdot x) \bmod(2^{16} + 1) = (65535 - x + 2) \bmod(65536)$.

Выявленную закономерность можно упростить до:

$(2^{16} \cdot x) \bmod(2^{16} + 1) = (1 - x) \bmod(2^{16})$

Тогда, блок схему операции умножения $\bmod(2^{16}+1)$ алгоритма IDEA можно представить, как:

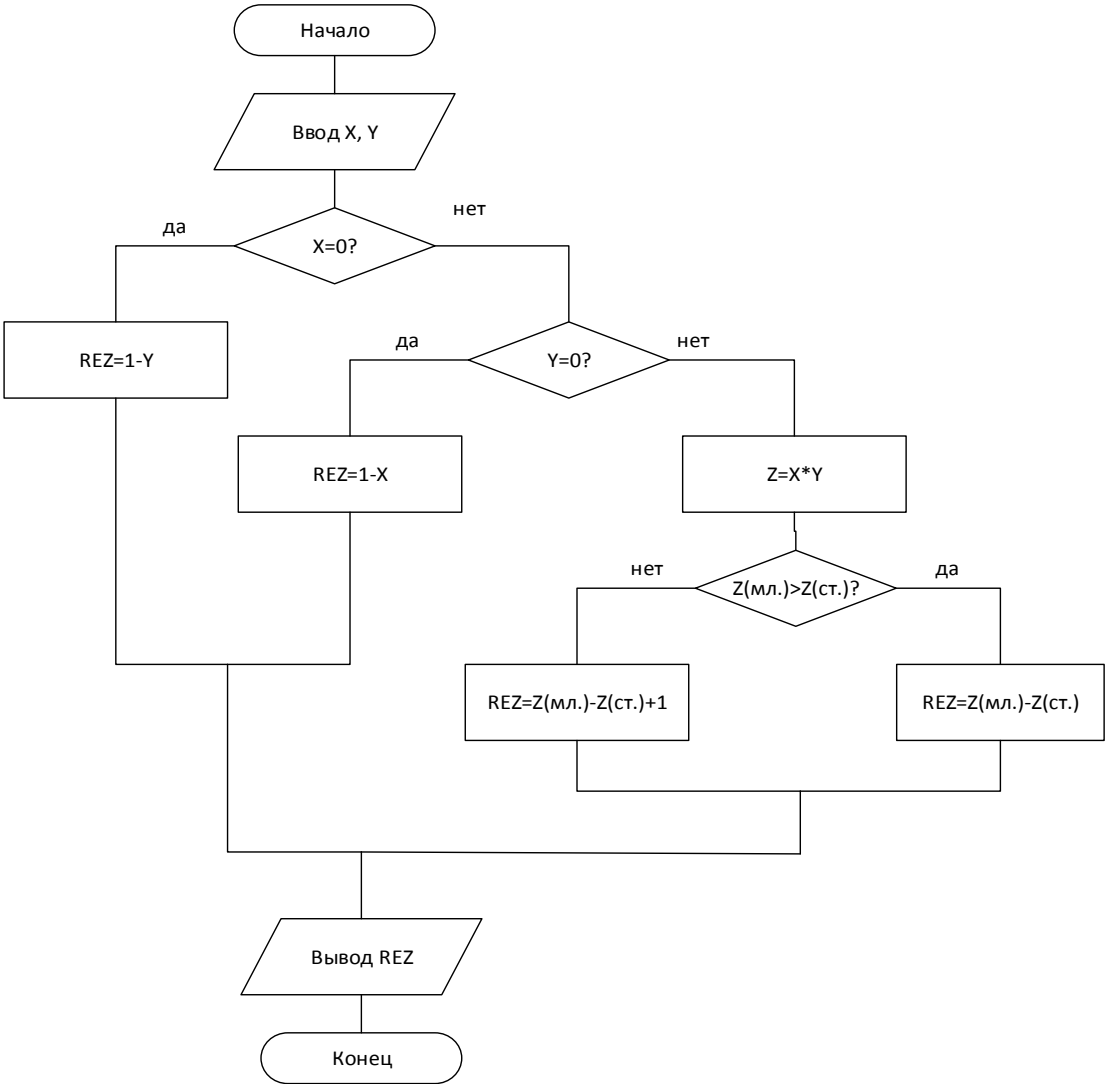


Рис. 3. Блок схема умножения IDEA

Выводы

1. Применение предложенного алгоритма позволит отказаться от сложно реализуемой операции нахождении остатка от деления при математических вычислениях в полях по модулю (X^n+k). Вместо нахождения остатка от деления предлагается применить одну операцию сравнения, две операции умножения и две операции сложения.

2. Данный алгоритм имеет ряд недостатков, главный из которых – это ограничение результата операции (до взятия модуля) числом Θ. Так, при реализации умножения, допустимо применение операндов всей разрядной сетки [0…X^n-1] только для k = 1. Этот порог возможно отодвинуть (или исключить совсем), однако это требует дальнейших исследований и разработки алгоритма, реализующего функцию $h = f(a,k)$.

3. Несмотря на порог по Θ, данный алгоритм можно успешно применять в некоторых алгоритмах шифрования, например в IDEA.

Литература

1. *Вейл. Г.* Алгебраическая теория чисел. М.: Государственное издательство иностранной литературы, 1947. 226 с.
2. *Окунев Л.Я.* Краткий курс теории чисел. М.: Госэнергоиздат, 1956. 240 с.
3. ITU-T. Recommendation H.223. Confidentiality system for audiovisual services. ITU, 2002.
4. *Шаврин. С.С, Зуйкова Т.Н, Мусатова О.Ю.* Реализация базовых алгоритмов шифрования на сигнальных процессорах. Часть 3. М.: МТУСИ, 2017. 35с.
5. *Шнайер Б.* Прикладная криптография. М.: Триумф, 2002. 816 с.

СПОСОБЫ ПОВЫШЕНИЯ ПОКАЗАТЕЛЕЙ КАЧЕСТВА РАБОТЫ ТРАНСПОРТНОЙ СЕТИ СВЯЗИ ТЕХНОЛОГИИ ПЦИ С ПОМОЩЬЮ АВТОМАТИЗАЦИИ ПРОЦЕССОВ СЕТЕВОГО ТЕХНОЛОГИЧЕСКОГО УПРАВЛЕНИЯ

Усков Вадим Дмитриевич
магистрант группы М61602 МТУСИ
uwd1000@gmail.com

Сызранцев Геннадий Валентинович
МТУСИ, д.в.н., доцент, зав. кафедры ССПиН
sysr9959@mail.ru

В работе описаны некоторые способы повышения показателей качества функционирования транспортной сети связи технологии ПЦИ с применением автоматизации процессов сетевого технологического управления. Представлен алгоритм взаимодействия телекоммуникационного оборудования, обладающий определённой новизной, внедрение которого обеспечит надёжное функционирование транспортной сети связи. Представлены результаты испытаний макетов мультиплексоров, реализующих предложенные технические решения, проведённые на кафедре ССПиН МТУСИ.

Ключевые слова: *ПЦИ, PDH, автоматизация, управление, сеть, контроль, модель.*

Технология ПЦИ и её недостатки

В 80-х годах минувшего века была разработана цифровая система передачи ПЦИ – плезиохронная цифровая иерархия (она же *PDH* в зарубежной литературе). К моменту начала своего использования данная система несла в себе огромный потенциал, но и технический прогресс не стоял на месте. С развитием технологий росли и объёмы передаваемой информации. Как следствие, усложнялись процессы обработки данных и предъявлялись новые требования к их передаче. На сегодняшний день применение этой технологии значительно сократилось, охватывая, в основном, местные сети и сети доступа.

Перечислим некоторые недостатки технологии:

- Наличие трёх различных иерархий;
- Затруднённый ввод/вывод цифровых потоков в промежуточных пунктах;
- Многоступенчатое восстановление синхронизма, занимающее достаточно длительное время;
- Отсутствие средств сетевого автоматического контроля и управления.

Именно отсутствие сетевого управления не позволяет строить высокодинамичные системы связи, которые требуются при решении экстренно возникающих задач, требующих немедленного реагирования.

Требования по оперативности управления сетью связи

Специализированные организации, части и подразделения, занимающиеся ликвидацией последствий техногенных катастроф, чрезвычайных ситуаций и других деструктивных воздействий, выполняют свои задачи, как правило, в районах, не оборудованных в отношении связи, или в условиях значительных нарушений телекоммуникационных структур. При таких или аналогичных условиях для обеспечения управления создаваемой группировкой развёртывается полевая система связи.

К развёртываемой полевой системе связи предъявляются высокие требования по управляемости, мобильности и пропускной способности.

Высокая мобильность отдельных средств связи и всей системы в целом обусловлена значительной неопределённостью на начальном этапе выполнения поставленных задач, затруднённостью ведения разведки местности по подготовке исходных данных для принятия решений.

В первую очередь, требования касаются технологического и оперативно-технического управления системой связи. Способность телекоммуникационного оборудования быстро реагировать на команды технологического управления во многом определяет численные значения показателей мобильности и, как следст-

вие, напрямую характеризуют оперативность управления группировкой и выполнение ею поставленных задач.

Именно от оперативности управления (принятие решения и доведение его до исполнителей), как правило, зависит исход той или иной ситуации, а, вероятно, и жизни задействованных в её разрешении людей.

Практикой эксплуатации полевых систем связи в условиях выполнения задач по ликвидации последствий ЧС определено, что требования, предъявляемые к полевой системе связи по мобильности и управляемости, могут быть выполнены только при автоматизации, как минимум, сетевых технологических процессов [1, 2]:

- по первоначальному конфигурированию структуры первичной сети связи;
- её переконфигурированию в ходе функционирования полевой системы связи;
- первоначальной прокладке трасс связей;
- перепрокладке их при изменении структуры первичной сети связи, отдельных информационных направлений, перемещении сетевых элементов или отдельных должностных лиц системы управления группировкой с персональными средствами связи;
- переводе связей на резервные трассы связи при выходе из строя основных, изменении структуры первичной сети связи или перемещении отдельных сетевых элементов, через которые проходили трассы связей;
- обеспечение полноценного функционирования отдельных фрагментов системы связи при её декомпозиции на два и более фрагмента.

Выполнение указанных функций технологического управления в автоматизированном режиме обусловливает построение автоматических сетей связи (автоматизированных систем связи) высокодинамичных систем управления специального назначения.

Информационные направления и связи в них, как правило, известны до начала работы создаваемой группировки специального назначения. По этой причине имеется возможность заранее, на этапе планирования или начальной стадии этапа организации связи, внести данные в базу данных автоматизированной системы связи по формируемым связям. На основе этих данных сетевая система управления технологическими процессами при регистрации в сети связи корреспондирующих узлов (подвижных корреспондентов) связи в автоматическом режиме запустит в действие механизм определения трассы связи и её загрузки в телекоммуникационном оборудовании (соответствующих блоках).

При построении системы связи на основе технических решений, предлагаемых в статье, требования по мобильности, управляемости и устойчивости к системе связи, со стороны системы управления, будут гарантированно выполнены, поскольку в автоматизированной (автоматической) системе связи всё технологическое управление средствами и комплексами связи осуществляется с тактовой частотой работы процессора по заранее записанной программе, разработанной на этапе принятия решения по связи в предстоящих действиях.

В качестве примера методов и показателей решения поставленной задачи представим последовательность действий центрального (шлюзового, сервера) сетевого элемента на этапе развёртывания автоматической сети связи (см. рис. 1).

На этапе развёртывания сети связи, при подключении сетевых элементов, функционирование телекоммуникационного оборудования направлено на автоматическую идентификацию, аутентификацию и регистрацию подключаемых сетевых элементов.

Данная блок-схема описывает последовательность действий работы центрального сетевого элемента при развёртывании автоматической сети связи. Шлюз опрашивает подключаемые к нему сетевые элементы на предмет их идентификационных данных и сверяет полученные данные с существующими в базе данных. При отсутствии – записывает новые и осуществляет коммутацию новых сетевых элементов.

При реализации в оборудовании описанных технических решений, испытания, проведённые на макетах модернизированных мультиплексоров, в лаборатории кафедры, показали следующие результаты (табл. 1).

Анализируя полученные значения, имеется возможность сделать вывод, что автоматизация сетевых технологических процессов при построении сети связи сокращает продолжительность первоначальной конфигурации, изменения конфигурации при функционировании сети связи, проключения трасс связей, резервирования трасс связей и их раскоммутации при необходимости. Сокращая продолжительность сетевых технологических процессов, гарантированно обеспечивается повышение показателей качества работы сети связи, построенной на оборудовании технологии ПЦИ.

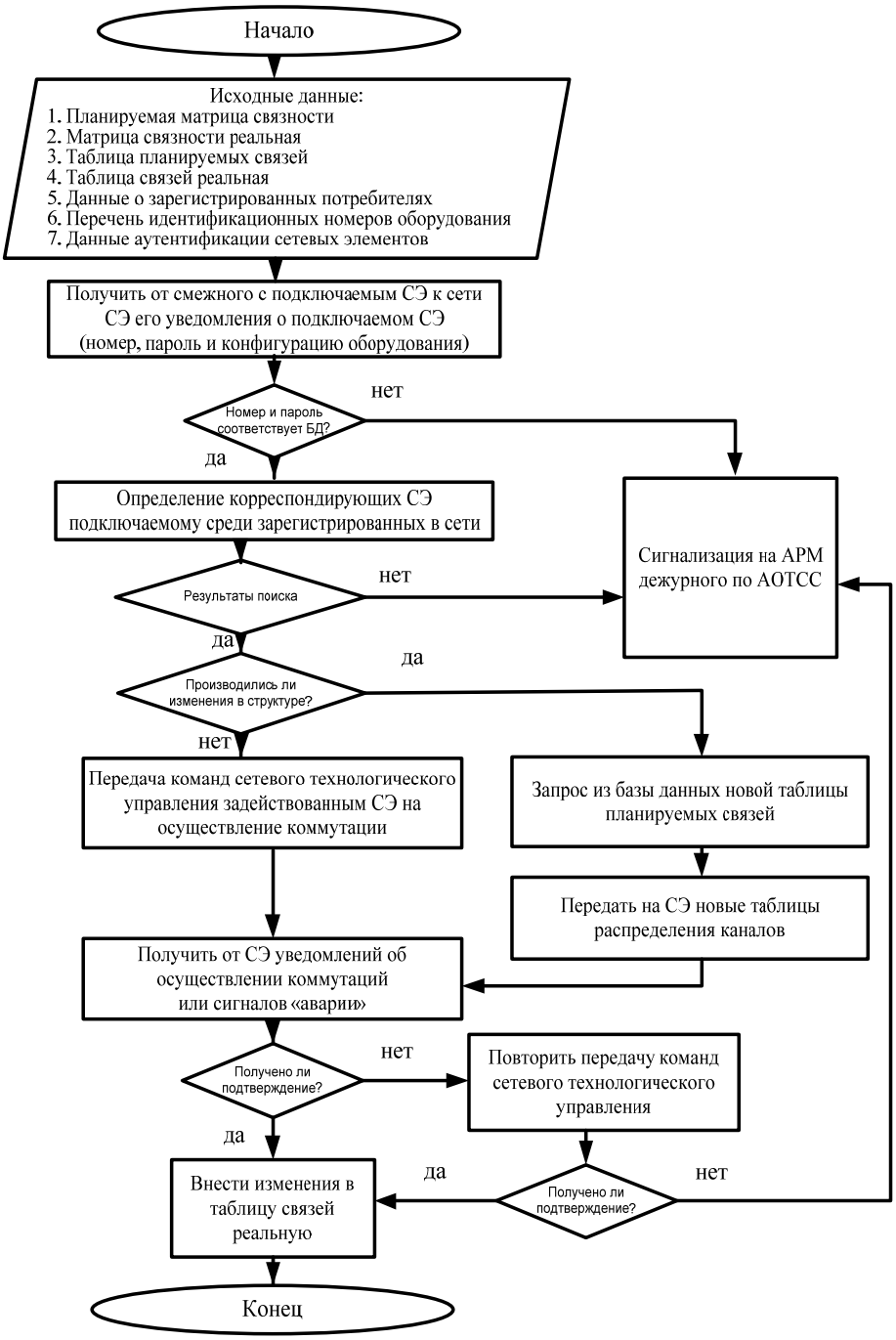


Рис. 1. Блок-схема функционирования центрального (шлюзового, сервера) сетевого элемента на этапе развёртывания автоматической сети связи

Таблица 1

Результаты расчётов оперативности функционирования сетевого технологического управления автоматической сетью связи на одном сетевом элементе

Виды управления	Ручной режим работы, мин	Автоматический режим работы (по битам нац. использования), сек	Автоматический режим работы по канальным интервалам потоков Е1 (с встроенным маршрутизатором сетевого управления), сек
Конфигурация	5,1	3,3	0,4
Изменение конфигурации	10,7	1,2	0,075
Проключение трасс	7,4	1,5	0,071
Резервная трасса	7,2	1,7	0,071
Расскомутация	6,1	0,24	0,062

Вывод

Предложенные технические решения обеспечивают увеличение численных значений показателей качества функционирования сети связи, построенной на оборудовании технологии ПЦИ, реализующих технические решения, представленные в данной статье.

Практическая реализация разработанных предложений позволит обеспечить и реализовать построение и функционирование в динамике ведения оперативных (тактических) действий и свёртывание динамически управляемой автоматической сети связи, реализующей автоматическую идентификацию, аутентификацию, первоначальную конфигурацию и переконфигурацию сети связи системы управления группировки специального назначения.

Литература

1. *Сызранцев Г. В., Мельников С. В., Лукин И. А*. Основные положения по построению автоматизированной полевой сети связи общего пользования. Труды 14-й Всероссийской НПК РАРАН «Актуальные проблемы защиты и безопасности» Изд. в 6-ти томах. Том 1. Вооружение и военная техника. СПб.: НПО СМ, 2011. С. 661-663.
2. *Сызранцева О.Г., Кириченко Р.Н., Иншин Г.В., Осарков В.Е.* Способ передачи служебной информации при автоматизации сетевых технологических процессов управления в высокодинамичных системах связи. Труды 16-й Всероссийской НПК РАРАН «Актуальные проблемы защиты и безопасности». Изд. в 5-ти томах. Том 1. СПб.: НПО СМ, 2013.

ИССЛЕДОВАНИЕ СИГНАЛЬНОЙ НАГРУЗКИ ПО ПРОТОКОЛУ SIP В ПОДСИСТЕМЕ IMS

Касапов Кирилл Валерьевич
магистрант группы М61601 МТУСИ
kirill23115@yandex.ru

Оханцев Сергей Сергеевич
магистрант группы М61603 МТУСИ
sergey-243@mail.ru

Маликова Елена Егоровна
МТУСИ, к.т.н., доцент кафедры ССиСК
emalikova@gmail.com

Рассмотрено использование платформы *SI*3000 компании *Iskratel* в учебных целях. Выполнено моделирование процесса установления соединения между двумя абонентами подсистемы *IMS*, показывающее циркуляцию заявок при установлении соединения между абонентами. Выполнен расчет таких показателей качества предоставления услуг, как интенсивность потока сигнальных сообщений, средняя длина сигнального сообщения и среднее время обслуживания сигнальной нагрузки на каждом узле. Во второй части работы показан расчет среднего времени пребывания заявки в системе массового обслуживания с помощью двух методов.

Ключевые слова: *сигнальная нагрузка, подсистема IMS, заявка, СМО, SI3000, компания Iskratel, протокол SIP, сеть NGN.*

В данной работе исследуется сигнальная нагрузка по протоколу *SIP* в подсистеме *IMS*.

Целью данной работы является изучение процесса установления соединения между двумя абонентами подсистемы *IMS* и изучение сигнальной нагрузки. Исследование сигнальной нагрузки по протоколу *SIP* имеет практическую значимость и может быть использовано при оценке среднего времени обслуживания заявки в подсистеме *IMS* при предоставлении основных и дополнительных услуг.

В данном исследовании были поставлены задачи: оценить нагрузку на различные элементы внутри ядра *IMS*, рассчитать среднее время пребывания заявки на каждом из узлов сети, а также среднее время пребывания заявки в системе массового обслуживания (СМО).

Для решения поставленных задач была составлена модель процесса установления соединения между двумя абонентами подсистемы *IMS*, на основе которой были рассчитаны интенсивность потока сигнальных сообщений, средняя длинна сигнального сообщения и среднее время обслуживания сигнальной нагрузки на каждом узле. Также в работе использованы два метода оценки среднего времени пребывания заявки в СМО.

В первой части работы представлено описание учебного стенда *SI*3000 от компании *Iskratel*, который используется для обучения студентов основам управления современным телекоммуникационным оборудованием. На рисунке 1 представлена структурная схема данного макета.

Макет серверного оборудования компании *Iskratel*, представленный в данной работе, расположен на кафедре ССиСК и включает в себя плату *ES* (*Ethernet*-коммутатор), эта плата используется для подключения двух телефонных *SIP* аппаратов, и применяется для коммутации всех узлов сети. На данном стенде организованы сеть *NGN* и сеть *IMS*. В качестве программного коммутатора для сети *NGN* применяется плата *CS-NGN,* плата *CS-IMS* выполняет роль *call*-сервера для сети *IMS* [1]. Два аналоговых телефонных аппарата подключаются к плате аналоговых абонентских линий *POTS* через кросс коннектор. Взаимодействие с другими коммутационными узлами осуществляется посредством платы транкингового шлюза *SMG* по протоколу *DSS*1. Для управления *call*-серверами применяются персональные компьютеры, подключенные к локальной сети. На сервере менеджмента *Lenovo* данного стенда размещена система управления *SI* 3000 *MNS* (*Management Network Services*) и сервер пользовательских данных *HSS* (*Home Subscriber Server*) [1].

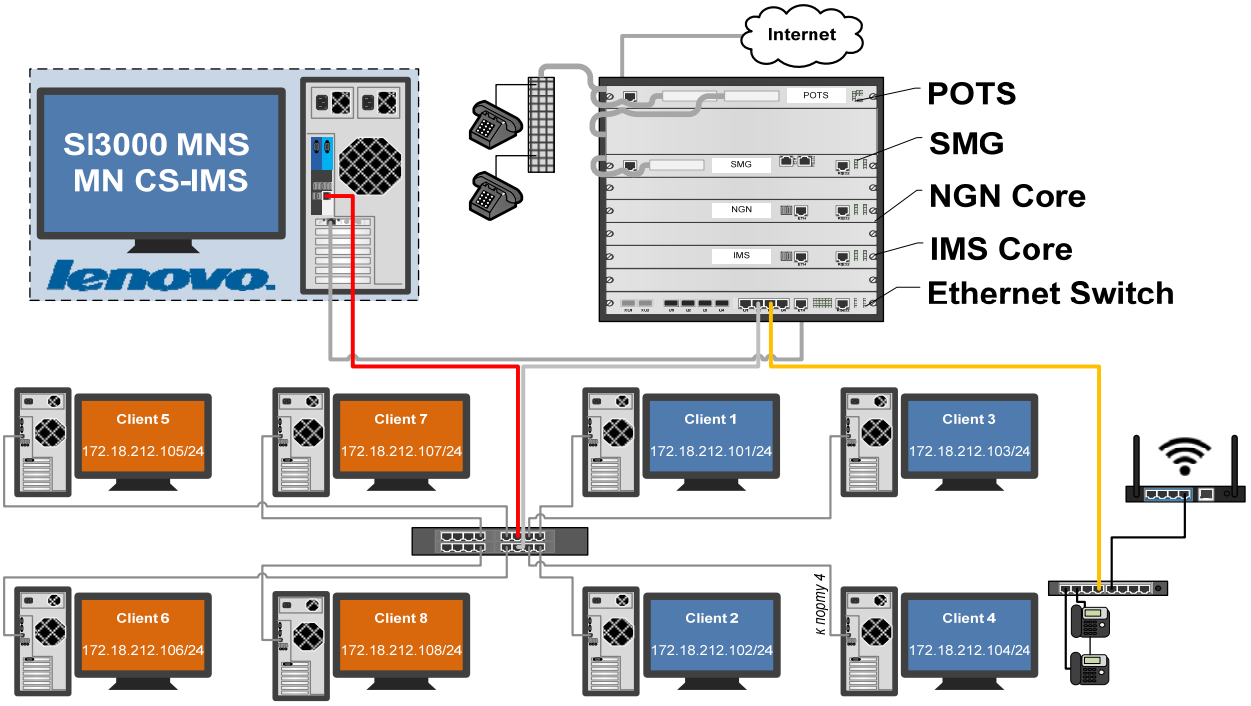


Рис. 1. Структурная схема макета *Iskratel SI*3000

Через точку *Wi-Fi* доступа осуществляется беспроводное подключение абонентов, использующих *SIP*-клиент для смартфона или планшетов.

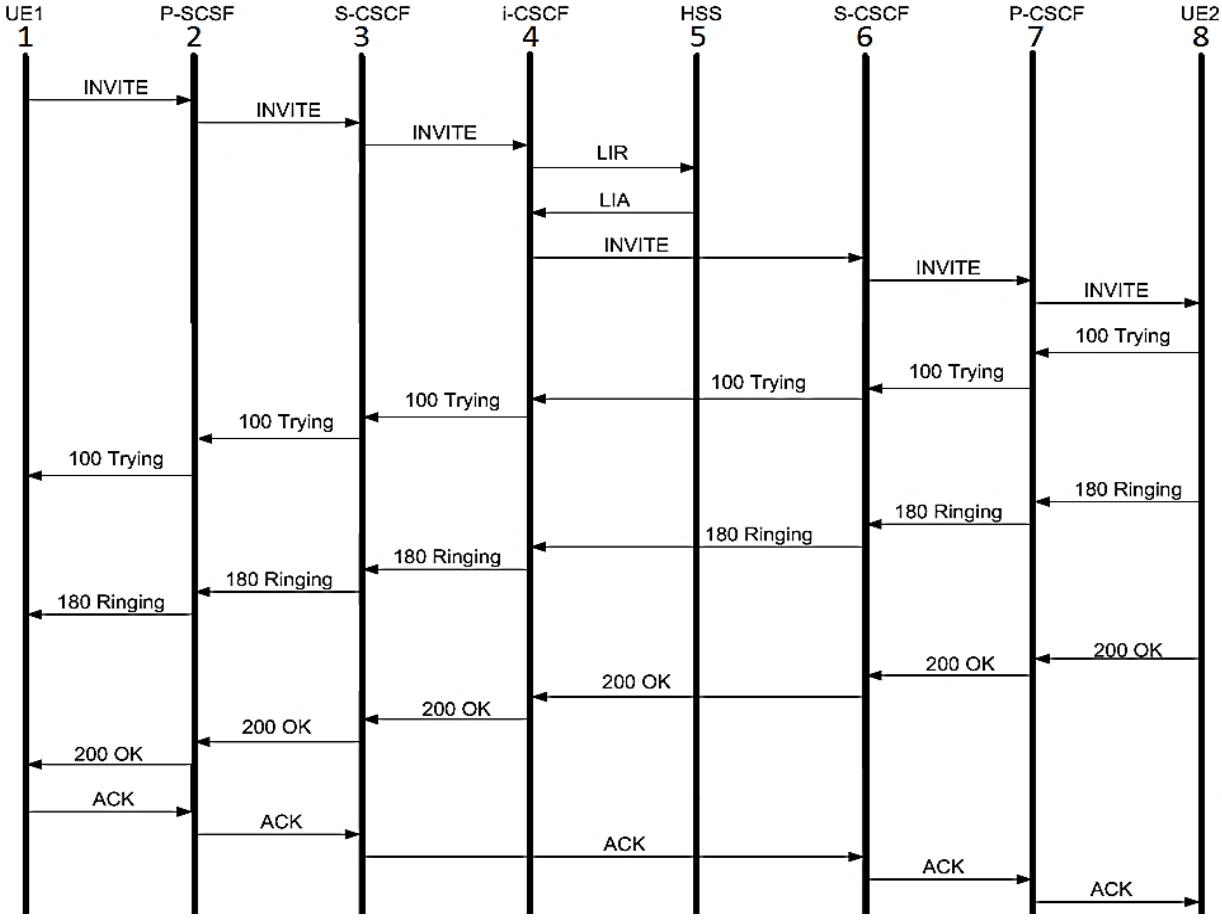


Рис. 2. Диаграмма установления соединения

С помощью программы-анализатора трафика *Wireshark* был изучен сигнальный обмен при установлении вызова между двумя абонентами, зарегистрированными в подсистеме *IMS*, на основе снятой трассировки. Далее построена диаграмма установления соединения, отображающая все сигнальные сообщения при установлении вызова. Диаграмма представлена на рис. 2.

На основе диаграммы установления соединения была составлена модель установления соединения, показанная на рис. 3 [2, 4]. Каждый элемент, отображенный на модели установления соединения, обозначен узлом (1-8), а для отображения входа и выхода заявок были добавлены узлы 0 и 9.

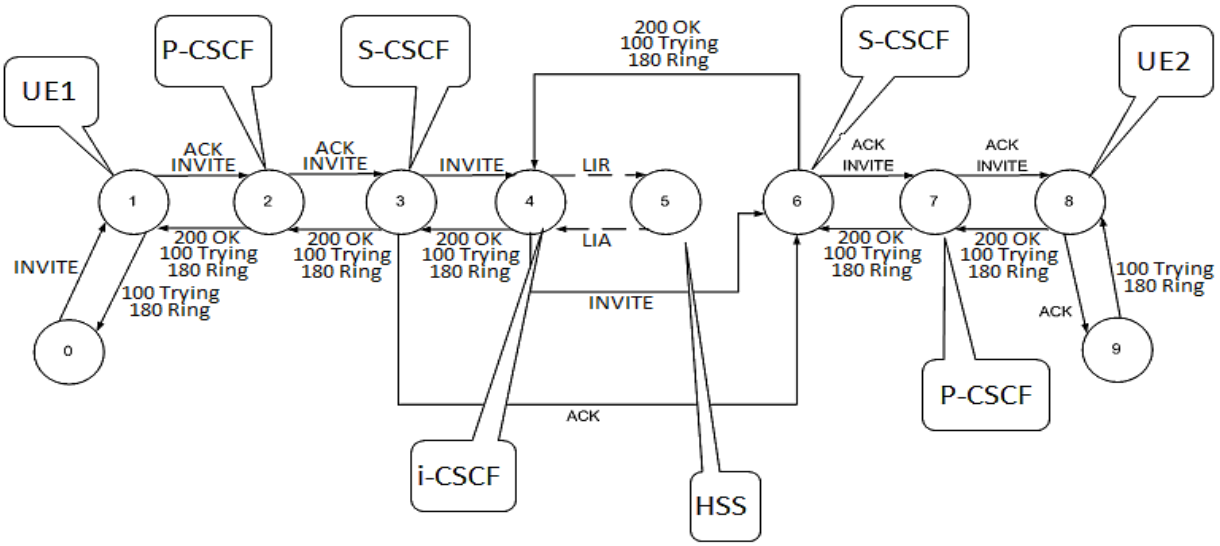


Рис. 3. Модель установления соединения

В качестве первого метода оценки времени пребывания заявки в СМО был использован метод, который разработали Самуйлов К.Е. и Гайдамака Ю.В. для расчета времени установления сессии с помощью системы сигнализации ОКС-7, позднее преобразованный для протокола *SIP*. Данный метод предусматривает разделение всех сообщений на классы заявок и дальнейшее составление матрицы вероятностей переходов.

Ниже представлен расчет таких показателей, как интенсивность потока сигнальных сообщений, средняя длина сигнального сообщения и среднее время обслуживания сигнальной нагрузки на каждом узле с помощью первого метода.

Учитывая то, что составленная схема является *BCMP* моделью [3] и поступающий поток заявок является пуассоновским, можно найти интенсивность потока сообщений по формуле (1):

$$\lambda_{i,r} = \sum_{(j,s)\in L} \lambda_{j,s} \cdot \theta_{js,ir} \qquad (1)$$

Решив систему уравнений, составленную на основе формулы (1), получим выражение:

$$\begin{cases} \lambda_i = q_i \cdot \lambda \\ q_i = \begin{cases} 4, & i = 1,4,8 \\ 5, & i = 2,3,6,7 \\ 1, & i = 5 \end{cases} \end{cases}.$$

Используя значения интенсивности поступающих сообщений (λ), пропускной способности (C) и длины сообщений (l_r), найдем среднюю длину сигнального сообщения, поступающего на i-й узел (l_i) по формуле (2) и среднее время обслуживания в узлах (b_i) по формуле (3):

$$l_i = \frac{\sum_{(i,r)\in L} \lambda_{i,r} \cdot l_r}{\sum_{(i,r)\in L} \lambda_{i,r}}, \qquad (2)$$

$$b_i = l_i + C, \qquad (3)$$

На основе выполненных расчетов составлены следующие диаграммы (рис. 4), из которых видно, что наибольшая нагрузка приходится на узлы 2 и 3, которые соответствуют элементам *P-CSCF* и *S-CSCF* подсистемы *IMS*.

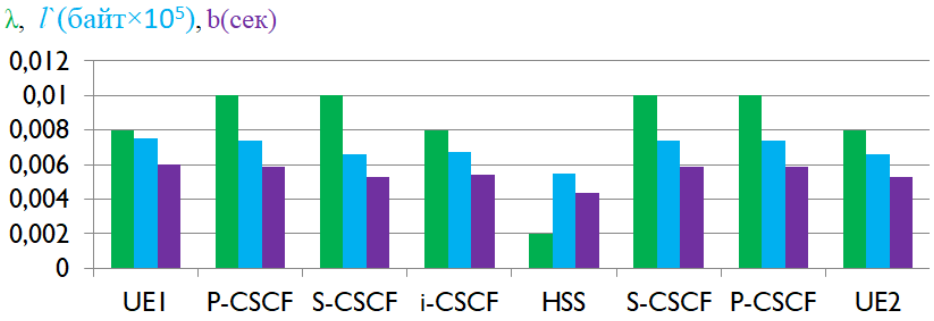


Рис. 4. Диаграммы для различных характеристик сигнальной нагрузки

Среднее время пребывания заявки в СМО, полученное на основе первого метода, составляет 0,0440 с.

В качестве второго метода, использованного в этой работе для оценки времени пребывания заявки в СМО, применен метод приближенной оценки, разработанный Геленбе Е. и Пюжолем Г. с использованием приближенной формулы Крамера и Лангенбах-Бельца. Позднее данный метод был доработан Башариным Г.П., Бочаровым П.П. и Наумовым В.А. и назван методом Университета Дружбы Народов.

Данный метод состоит из 4-х этапов [5].

На первом этапе выполняется расчет коэффициента вариации интервалов между поступлениями заявок на узел.

$$\lambda_i \cdot C_{A_i}^2(t) - \sum_{k=1}^{M} \lambda_{kt}\theta_{kt}\left(1-\rho(k)\right)C_{A_k}^2(k) == \lambda_{0t}C_{A_0}^2(0,t) + \sum_{k=1}^{M}\lambda_{kt}\left((1-\theta_{kt}) + \vartheta_{kt}\rho(k)C_{B_k}^2(k)\right), \quad (4)$$

где $\rho(k) = \begin{cases} \frac{\lambda_k}{\mu_k}, k \in M_{M/M/1} \\ 0, k \in M_{M/D/1}, \end{cases}$

λ_i – интенсивность потока заявок в *i*-узле, C_A – коэффициента вариации интервалов между поступлениями заявок на узел, θ – маршрутная матрица, C_B – коэффициента вариации интервалов между выходом заявки из узла, ρ – отношение интенсивности входящего потока на узел к интенсивности выходящего потока из узла.

На втором этапе определяется среднее время ожидания начала обслуживания заявки в *i*-узле с помощью приближенной формулы Крамера и Лангенбах-Бельца:

$$\omega_i \approx \frac{\rho_i b_i}{2(1-\rho_i)}\left(C_{A_i}^2(t) + C_{A_i}^2(t)\right) \cdot g\left(\rho_i, C_{A_i}(t), C_{B_i}(t)\right), \quad (5)$$

$$g\left(\rho_i, C_{A_i}(t), C_{B_i}(t)\right) = \begin{cases} \exp\left(\frac{-2(1-\rho_i)\left(1-C_{A_i}^2(t)\right)^2}{3\rho_i\left(C_{A_i}^2(t)+C_{B_i}^2(t)\right)}\right), & C_{A_i}(t) \le 1 \\ \exp\left(\frac{-(1-\rho_i)\left(C_{A_i}^2(t)-1\right)}{C_{A_i}^2(t)+4C_{B_i}^2(t)}\right), & C_{A_i}(t) > 1 \end{cases}, \quad (6)$$

На третьем этапе рассчитывается среднее время пребывания заявки в *i*-узле.

$$v_i = \omega_i + b_i, \quad (7)$$

b_i – время обслуживания в *i*-узле.

На четвертом этапе определяется общее время пребывания заявки в СМО как сумма времен пребывания заявки в каждом из узлов.

На рисунке 5 представлены результаты расчета среднего времени пребывания заявки на каждом узле. Среднее время пребывания заявки в СМО, полученное на основе второго метода, составляет 0,0557 с.

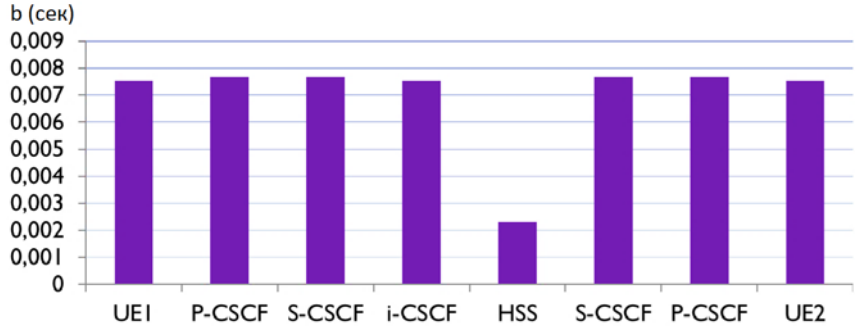


Рис. 5. Значения среднего времени пребывания заявки на каждом из узлов

Вывод

приведенная модель установления соединения позволяет оценить сигнальную нагрузку и задержку передачи сигнальных сообщений в подсистеме IMS. Как видно, оба метода дают схожие результаты. Отличие полученных результатов объясняется тем, что первый метод является более простым для расчетов и дает оценку времени установления сессии в явном виде, а второй метод является универсальным и позволяет приближенно оценить время пребывания заявки в СМО для любых типов систем. На основе приведенных выше расчетов можно сделать вывод, что наибольшая нагрузка приходится на узлы *P-CSCF* и *S-CSCF* подсистемы IMS.

Литература

1. *Вилков А.Р., Касапов К.В., Маликова Е.Е.* Постановка лабораторных работ на базе оборудования SI3000 компании Iskratel // Телекоммуникации и информационные технологии. №1. 2016. С. 84-87.
2. *Самуйлов К.Е., Сопин Э.С., Чукарин А.В.* Оценка характеристик сигнального трафика в сети связи на базе полсистемы IMS // T-Comm: Телекоммуникации и транспорт. №7. 2010. С. 8-13.
3. *Абаев П.О., Салтымакова К.Э., Ивашиненко Е.А.* Построение и анализ модели для оценки времени установления соединения в подсистеме IMS // T-Comm: Телекоммуникации и транспорт. №11. 2013. С. 11-15.
4. *Степанов С.Н.* Теория телетрафика: концепции, модели, приложения. М.: Горячая линия – Телеком, 2015. 868 с.
5. *Зарипова Э.Р.* Методы анализа показателей эффективности телекоммуникационной сети серверов протокола установления сессий. Диссертация на соискание ученой степени кандидата физико-математических наук. М.: 2014. С. 94-96.

ЗАЩИТА ИСХОДНОГО КОДА С ИСПОЛЬЗОВАНИЕМ МЕТОДА ОБФУСКАЦИИ

Креймер Андрей Викторович
студент группы М091601(72) МТУСИ
akreymer@yandex.ru

Беленькая Марина Наумовна
доцент кафедры МСиУС МТУСИ
mn.belenkaya@mail.ru

Защита исходного кода программного обеспечения – важная задача при разработке коммерческого программного продукта. При наличии конкуренции злоумышленники могут украсть разработанный исходный код или реализованные в нем алгоритмы, методы обработки данных, технологии. Необходимо затруднить процесс обратной инженерии, чтобы повысить защищенность кода программного продукта от использования конкурентами. В статье рассматриваются существующие виды и способы запутывания (обфускации) исходного кода, написанного на языках программирования разного уровня, предлагаются к рассмотрению наглядные примеры применения основных принципов обфускации, а также делаются выводы о том, в каком объеме и к каким участкам исходного кода следует применять запутывание.

Ключевые слова: *обфускация, защита исходного кода, запутывание исходного кода, методы обфускации, виды обфускации.*

Исходный код процессов операционных систем, исполняющих роль сервера в мобильных и интернет-приложениях, представляет интерес как для конкурентов компании-разработчика, так и для злоумышленников. Имея доступ к исходному коду, применить его можно в различных целях, например, использовать в качестве основы для собственной разработки или найти уязвимости для обхода систем безопасности программного продукта. Для предотвращения таких случаев компания-разработчик должна предусмотреть методы защиты своего исходного кода.

Основной процесс в изучении чужого исходного кода называется процессом реверсивной (обратной) инженерии. Он позволяет злоумышленнику понять принцип работы программного продукта изнутри. Так как реверсивная инженерия — задача, связанная с пониманием исходного кода человеком, то одним из методов защиты является намеренное усложнение или запутывание исходного кода программного продукта.

Обфускация — метод защиты программного продукта, при котором его исходный код искусственно «запутывается» (от англ. «*obfuscation*» – запутывание). То есть, написанный разработчиком код после применения специальных алгоритмов изменяется до состояния, которое гораздо труднее «читается» человеком при реверсивной инженерии. В процессе обфускации предпринимаются попытки максимально устранить логические связи в исходном коде.

Процесс изменения исходного кода можно считать полноценным процессом обфускации при выполнении следующих условий:

- измененный вариант кода сильно отличается внешне от исходного, но при этом остается работоспособным, выполняя те же функции с тем же результатом (за исключением скорости работы);
- процесс реверсивной инженерии измененного варианта кода требует большего количества времени и трудозатрат, чем исходного варианта;
- каждая попытка обфускации исходного кода программного продукта должна давать различные результаты (то есть, на выходе должен получаться разный обфусцированный код);
- написание программного продукта для выполнения процесса обратного преобразования обфусцированного кода к состоянию, близкому к исходному, не будет эффективным.

Так как при обфускации одного и того же исходного кода получаются разные результаты, это можно использовать, например, в качестве защиты от распространения нелицензионных копий программного продукта. Если каждая обфусцированная копия уникальна, то также уникальна и ее контрольная сумма, кото-

рая может быть использована для проверки подлинности.

Подвергать обфускации весь исходный код программы может быть неверным решением, так как это снижает быстродействие программного продукта. Поэтому необходимо проанализировать весь исходный код, выявить все важные участки кода и подвергнуть процессу обфускации только их. В случае применения обфускации для защиты исходного кода мультимедиа-сервера, разработанного в рамках бакалаврской работы одного из авторов статьи и функционирующего под управлением операционной реального времени *RTLinux*, наиболее критичными участками кода являются фрагменты обработки потоков данных (как входящих, так и исходящих), фрагмент агрегирования нескольких потоков в один, а также фрагмент обращения к операционной системе при инициализации сервера.

Процесс обфускации может быть применен как к исходному коду, представленному в виде последовательности инструкций какого-либо языка программирования, так и к коду, представленному в двоичном виде. На основе этой особенности выделяют два уровня этого процесса:

- обфускация низшего уровня, изменяющая исходный код, представленный на низкоуровневом языке программирования или в машинных кодах;
- обфускация высшего уровня, изменяющая исходный код, написанный на языках программирования высокого уровня (например, *C/C++*, *Java*, *Python*).

Таблица 1

Исходный код на языке C++

```
#include <stdio.h>

void Start(void);
void Task(void);
void Finish(void);

int main(void)
{
    // Вызов функции Start()
    Start();
    /* Некоторый комментарий
    в стиле языка C */
    Task();
    /* Ещё немного комментария */
    Finish(); // Вызов функции Finish()
    return 0;
}

void Start(void)
{
    printf("Start\n");
}

void Task(void)
{
    // Something task
    printf("\tDo smth...\n");
}

void Finish(void)
{
    printf("Finish\n");
}
```

Таблица 2

Лексически обфусцированный исходный код

```
#include <stdio.h>
void yq5uigojw6le(){printf("Finish\n");}void yq5uigojw6le();void ppotg6yogd98th(){/*wекийv
psej2й09wi1 lkjdsжid29.8c эf*/printf("\tDo smth...\n");}void Aui1IgbDw();int
main(){/*srfbvdtцyhdfsпhиgstyfhb ш rgeъrsdgsdёfgsrt*/Aui1IgbDw();goto
uohi8iokes;gdsfied:ppotg6yogd98th();/*Rock jdbopФws k2wrs9f jpef fбr
ji5*/yq5uigojw6le();return 0;uohi8iokes:for(int m9h8jh=819;m9h8jh>m9h8jh;m9h8jh+=5);/*fo9
Ымэdcifo switch kdцupк; d*/goto gdsfied;}void Aui1IgbDw(){printf("Start\n");};void
ppotg6yogd98th();
```

Несмотря на то, что обфускация низшего уровня представляется менее комплексной задачей, ее реализация гораздо сложнее. Так как современная вычислительная техника использует процессоры, основанные на большом количестве различных архитектур, при написании алгоритма программы-обфускатора необходимо учитывать особенности каждого набора инструкций. Сложность этого процесса также является одной из причин, почему данный процесс мало исследован.

Запутыванию исходного кода могут быть подвергнуты различные его элементы, поэтому обфускацию можно разделить на несколько видов:

1) Лексическая обфускация. Представляет собой самый простой способ организации запутывания исходного кода, затрудняющий чтение и понимание путем использования следующих приемов:

- все комментарии в файлах исходных кодов удаляются либо заменяются на бессмысленные или дезинформирующие;
- в исходный код добавляются различные ненужные элементы (не изменяющие результат работы программного продукта);
- отдельные блоки исходного кода (например, функции или процедуры) перемещаются в другое место, но таким образом, чтобы это не повлияло на работу программного продукта;
- короткие, простые и понятные имена переменных, структур, функций и прочих объектов в исходном коде изменяются на бессмысленные длинные наборы символов, чтобы усложнить понимание и восприятие человеком;
- символы, используемые для форматирования внешнего вида исходного кода (пробелы, табуляция, переносы строк) удаляются, что также добавляет трудности при чтении и анализе кода.

В таблицах 1 и 2 приводится пример использования лексической обфускации примитивного исходного кода на языке *C++* (обфускация проводилась автором вручную для демонстрации всех приемов).

Оба исходных кода, хранящихся в файлах *src_clear.cpp* («чистый» исходный код) и *src_obfus.cpp* (обфусцированный исходный код), были транслированы в исполняемые коды – файлы *clear.out* и *obfus.out* соответственно – с помощью компилятора *g++* версии 5.4.0. Компиляция и исполнение программ проводились под управлением операционной системы *Ubuntu* с ядром *Linux* 4.4.0-31-*generic* (рис. 1).

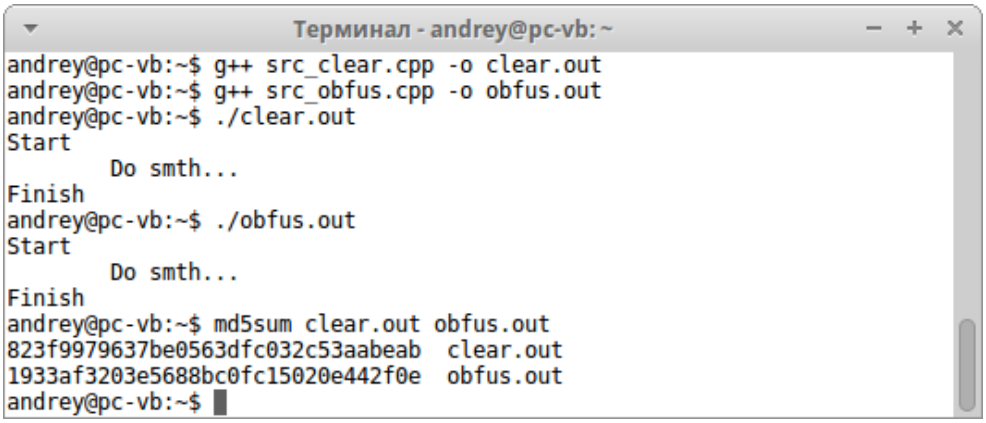


Рис. 1. Результаты компиляции и исполнения программы с исходным кодом и программы с обфусцированным исходных кодом

Как видно из рис. 1, результаты работы программ, полученных из исходного кода и офусцированного исходного кода, идентичны. Тем не менее исходные коды (табл. 1 и 2) существенно отличаются, как и исполняемые файлы, что подтверждается уникальными значениями контрольных сумм, подсчитанных по алгоритму *MD5* с помощью системной утилиты *md5sum*.

Рассмотрим приведенный пример согласно приемам лексической обфускации.

Первый прием – удаление или изменение комментариев. Например, комментарий «*// Вызов функции Start()*» заменен на «бессмысленную» фразу «*//*srfbvdtцyhdfsnhugstyfhb ш rgeъrsdgsdëfgsrt*/*», а комментарий «*// Вызов функции Finish()*» удален полностью.

Прием добавления ненужных элементов представлен циклом *for*, выполняющим бесполезную работу по проверке условия цикла, а также оператором *goto*, который сам по себе затрудняет понимание текста любой программы. Первая метка данного оператора ведет в конец функции *main* (после оператора *return*), вторая метка возвращает исполнение программы обратно.

В качестве иллюстрации третьего приема – перемещения блоков кода – функция *main* поднята выше, а прототипы и тела других функций перемешаны, но таким образом, что область их видимости не нарушена.

Прием переименования простых и понятных имен функций продемонстрирован на всех именах функций, кроме *main* (так как это точка входа в программу, ее имя должно остаться неизменным для корректной компиляции и исполнения). Например, функция «*Start()*» после переименования получила имя «*Aui1IgbDw()*», а функция «*Finish()*» – имя «*yq5uigojw6le()*». Данные имена при чтении кода не дают представления о назначении функции, а также их сложно запоминать и отличать от других подобных имен при дальнейшем анализе текста.

Последний прием – удаление форматирования исходного кода – наиболее нагляден. При его использовании текст программы сливается в одну или несколько строк, не имеющих каких-либо символов табуляции или пробелов, кроме необходимых.

Лексическая обфускация затрудняет чтение даже примитивного примера кода. При обфускации подобным образом больших и сложных частей исходного текста эффективность данного метода возрастает. Его преимуществом является возможность быстро преобразовать исходный код программного продукта в нечитаемый вид. Недостатком же является область применения – метод возможно использовать только для обфускации исходных текстов языков высокого уровня.

2) Обфускация данных. Этот вид запутывания исходного кода применяется для преобразования используемых в программном продукте структур данных. Сюда входят методы изменения исходного кода, разделенные на три группы.

Первая группа – обфускация хранения. Эта группа методов включает в себя следующие методы запутывания кода, изменяющие как сами типы данных, так и хранилища данных:

- запутываются (подменяются) способы интерпретации имеющихся в программном продукте данных определенного типа;
- смена области видимости хранилищ данных (например, локальное хранилище переносится в глобальную область);
- замена статических данных на их вычисление в процедурах (например, замена символов в строке их двоичным или шестнадцатеричным представлением, замена числовой константы сложным математическим выражением, «сборка» строковой константы внутри процедуры с использованием циклов, условий, вычислений, сложных для понимания операторов языка программирования);
- разделение используемых переменных, имеющих заранее известный заданный диапазон значений (например, переменная типа *boolean* может иметь значения только *true* и *false*, что известно заранее, и переменную такого типа можно разделить на две);
- подмена представления данных определенного типа (например, значение целочисленной переменной может быть представлено с использованием математического выражения).

Вторая группа – обфускация соединения. Задачей данной группы методов является, как следует из названия, запутывание с использованием соединения данных, независящих друг от друга, а также обратный процесс – разделение зависимых друг от друга данных. Ниже рассматриваются основные методы, относящиеся к обфускации соединения.

Метод соединения независимых друг от друга переменных. Две и более переменные можно соединить в одну переменную при условии, что их суммарный размер (в битах) не превысит размер новой переменной, используемой для соединения. Например, в языках *C/C++* для записи целочисленных значений можно использовать такие типы, как *short* (длина – 16 бит) и *int* (длина – 32 бита). Согласно вышеописанному правилу, в одной переменной типа *int* можно соединить две переменных типа *short*. Соединение проводится таким образом, что первые 16 бит переменной типа *int* отдаются для хранения значения одной переменной типа *short*, следующие 16 бит – для другой переменной.

Для получения значения объединяющей переменной используется формула, строящаяся по следующему правилу: значение первой соединяемой переменной прибавляется к объединяющей переменной без изменений, а значения последующих переменных предварительно умножаются на 2^N, где N – отступ (количество бит от начала), затем прибавляются к объединяющей переменной. Например, при объединении переменных A, B типа *short* в переменную C типа *int* может быть использована следующая формула: $C(A, B) = A + B * 2^{16}$. Не имеет значения, какая из соединяемых переменных будет храниться в начальных битах, а какая – в следующих. Программный обфускатор при использовании значений соединенных переменных проводит необходимые преобразования. Например, чтобы к значению переменной A, соединенной вышеуказанной формулой в переменную C, прибавить целое число n, необходимо выполнить обычное сложение, то есть $C = C + n$, а для прибавления числа n к переменной B будет использована формула $C = C + n * 2^{16}$.

Метод изменения структуры массивов. Данный метод соединяет либо разделяет одномерные и многомерные массивы. Ниже приводятся несколько примеров.

Одномерный массив преобразуется в двумерный. То есть, одномерный массив *A*, содержащий в себе 12 элементов, можно представить как двумерный массив с размерностью 2x6, 3x4 или 4x3.

Одномерный массив разделяется на несколько производных одномерных массивов по какому-либо правилу, например, разделение по признаку четности номеров элементов исходного массива (элементы с индексами 0, 2, 4 и т.д. помещаются в один массив, а с индексами 1, 3, 5 и т. д. – в другой).

Двумерный массив раскладывается на несколько одномерных, каждый из которых будет содержать в себе одну строку двумерного массива.

Строки и столбцы двумерного массива могут быть зеркально отражены, то есть строки станут столбцами, а столбцы – строками.

Метод запутывания иерархии наследования классов. Обфускация с использованием данного метода использует манипуляцию с наследованием классов таким образом, что логика наследования становится запутанной и не очевидной. Для этих целей программные обфускаторы могут, например, создавать дополнительные классы и встраивать их на какое-либо место в исходную иерархию классов, а также создавать ложные ветвления иерархии.

Третья группа – обфускация переупорядочивания. Эта группа методов включает в себя манипуляции с последовательностью расположения некоторых инструкций, изменением порядка элементов в структурах, запутыванием представления многомерных массивов и т. д. Например, переменная, объявленная перед блоком кода, в котором она используется, может быть перенесена в случайное более раннее место (это не повлияет на работоспособность кода и на результат).

3) Обфускация управления. Такой вид запутывает обычный и понятный ход выполнения программы путем применения так называемых непрозрачных предикатов. В данном случае непрозрачный предикат – это логическое выражение (как правило, очень сложное), которое имеет заранее известный для программиста результат, но все равно требует вычислений, так как не представлено в явном виде (единственным и явно выраженным значением «*true*» или «*false*»). С помощью непрозрачных предикатов выполняются преобразования структуры кода, изменяющие как внешний вид, так и логику выполнения программы.

Например, можно внутри какой-либо процедуры с помощью оператора «*if-else*», в котором выбор исполняемой ветви будет определяться непрозрачным предикатом, разместить два блока программы, используемые в разные моменты времени, и значение непрозрачного предиката будет строго зависеть от точки вызова процедуры. Таким образом, разрабочик программного продукта будет точно знать, какое значение в том или ином месте будет иметь предикат, и, следовательно, какой блок кода будет выполнен при том или ином вызове, а злоумышленник в процессе деобфускации будет вынужден предпринимать попытки разобраться в этом предикате.

Также, помимо «полезного» исходного кода, в подобные ветвления, зависящие от непрозрачных предикатов, можно помещать такие блоки кода, которые никогда не будут выполнены (в таком случае предикат никогда не будет принимать соответствующее этому блоку значение), либо блоки кода, выполняющие ложные операции. В таком случае, чем более разветвленный исходный код будет представлен с помощью непрозрачных предикатов, тем более сложным будет процесс реверсивной инженерии.

Для применения подобной обфускации возможно использование программных генераторов непрозрачных предикатов. К таким генераторам следует предъявить ряд требований, обеспечивающих эффективность использования:

- время генерации предиката должно быть линейным от длины генерируемого предиката;
- результат – сгенерированный непрозрачный предикат – должен быть похож на обычное выражение, используемое в исходных кодах (маскировка применения непрозрачного предиката);
- невозможность определения по сгенерированному предикату того, что применялся конкретный программный генератор;
- невозможность определения по сгенерированному предикату того, всегда ли его значение истинное (или ложное) или существует комбинация входных параметров, при которой результат будет противоположным.

Примеры качественных непрозрачных предикатов, которые возможно получить с использованием программных генераторов, приведены в табл. 3.

4) Превентивная обфускация. Этот вид нацелен на затруднение использования средств автоматизации при попытках проведения деобфускации. Например, применение статического анализатора исходного кода можно сделать менее эффективным и более трудоемким благодаря использованию в исходном коде большого количества структур указателей. Подобным образом следует изучать уязвимые места средств автоматизации, которые могут быть использованы для деобфускации конкретного исходного кода (набор таких средств может зависеть, например, от выбранного языка программирования) и применять в процессе обфускации соответствующие превентивные приемы.

Таблица 3

Примеры генерируемых непрозрачных предикатов

```
~x != x
(x + x & 1) == 0
~x != x * 4u >> 2
(-x & 1) == (x & 1)
x - 0x9d227fa9 != x - 0x699c945e
(x | 0xffffdbe8) - 0x1baa != x || (x & 0x10) == 0x10
((uint)x % 0x38 + 0xe4df62c8 & 0x6d755e00) == 0x64554200
(x & 0x8e3ef800) != 0x70641deb && (uint)x / 0x9388ea != 0x3ab6921c
```

Совокупное использование всех упомянутых выше методов запутывания исходного кода программных продуктов позволяет достичь максимальной эффективности защиты от реверсивной инженерии. В случае с системами обработки данных, такими как мультимедийные системы или системы анализа трафика, необходимо использовать все доступные способы защиты исходного кода от злоумышленников, в том числе и обфускацию. Поскольку применение подобных методов защиты влияет на производительность программного продукта, необходимо опытным путем выявлять максимально допустимый объем исходного кода критически важных блоков, который с применением обфускации сможет обеспечить требуемую скорость.

Литература

1. *Christian, Collberg*. A taxonomy of Obfuscating Transformations / Collberg. Christian, Thomborson. Clark, Low. Douglas. // Technical Report #148 Department of Computer Science The University of Auckland. C. 36.
2. *Gregory, Wroblewski*. General Method of Program Code Obfuscation. Wroclaw, 2002. 112 c.
3. *Michael, D. Ernst*. Static and dynamic analysis: synergy and duality // MIT Lab for Computer Science, 2003. C. 4.
4. *S. Prata*. C++ Primer Plus, 6th edition // Pearson Education, Inc. 2014. 1181 c.
5. *Беленькая М. Н., Малиновский С. Т., Яковенко Н. В*. Администрирование в информационных системах. Учебное пособие для вузов. М.: Горячая линия – Телеком, 2014. 400 с.

ПРИМЕНЕНИЕ МЕТОДА ЛОГИСТИЧЕСКОЙ РЕГРЕССИИ ДЛЯ ЗАДАЧИ КЛАССИФИКАЦИИ ТЕКСТОВ СУДЕБНЫХ РЕШЕНИЙ

Стрельников Владимир Геннадьевич
магистр группы М151701(70) МТУСИ
strel-prod@yandex.ru

Трунов Артем Сергеевич
МТУСИ, старший преподаватель кафедры ИСУиА,
Главный специалист управления информатизации
и связи Верховного Суда РФ
greek17@yandex.ru

Проводится исследование и реализация поэтапного решения задачи классификации документов на примере логистической регрессии. Описана реализация предобработки текстов, которая заключается в формировании векторной модели представления документов по приведенным методам Bag of N-grams и TF-IDF. Выполняется проверка работоспособности метода логистической регрессии на нескольких примерах по показателям точности и F-меры.

Ключевые слова: *машинное обучение, классификация, тексты, Bag of N-grams, TF-IDF, логистическая регрессия.*

В стремительно развивающееся информационное пространство активно внедряются методы машинного обучения для решения задач, связанных с обработкой естественного языка.

Обработка естественного языка – это область информатики, искусственного интеллекта и математической лингвистики, задача которой заключается в изучении проблем, связанных с компьютерным анализом естественного языка и его синтезом. К данному классу задач относятся: распознаванием речи; интеллектуальный анализ текстов для извлечения структурированной информации, построения вопросно-ответных систем; синтез речи и т.д.

Одним из направлений интеллектуального анализа текстов является классификация полнотекстовых документов, имеющая широкий спектр применения, например, в автоматизированной обработки заявок, фильтрации документов, персонализации новостей.

Существует актуальная задача, направленная на создание интеллектуальной системы анализа судебных решений региональных судов для выявления однотипных групп дел с целью определения общепринятой практики судов. Первоначальный этап – создание алгоритма, позволяющего классифицировать судебные решения.

В данной статье рассматривается поэтапное решение задачи классификация судебных решений на примере машинного обучения с учителем для метода логистической регрессий.

Решение задачи классификации полнотекстовых документов с применением машинного обучения подразделяется на две категории: обучение с учителем и обучение без учителя.

Таблица 1

Примеры алгоритмов классификации полнотекстовых документов [1]

Обучение с учителем			Обучение без учителя	
Алгоритм	Сложность		Алгоритм	Сложность
	Обучение	Тестирование		
K-ближайших соседей	$O(1)$	$O(N)$	K-средних	$O(N)$
Алгоритм опорных векторов	$O(CN^2)$	$O(C)$	Плотностный алгоритм *DBSCAN*	$O(n \log n)$
«Наивная» байесовская классификация	$O(N)$	$O(C)$	Нейросетевой алгоритм *SOM*	$O(TN^2)$
Логистическая регрессия	$O(N^3)$	$O(T)$	Нечеткий алгоритм c-средних	$O(N)$

C – число классов, N – число документов, T – число термов.

Обучение с учителем подразумевает наличие обучающей выборки, состоящей из коллекции документов, для которых заранее известна принадлежность к тем или иным классам [1]. Алгоритмы, решающие данную задачу, обучаются на тренировочном наборе размеченных документов и классифицируют новые документы в зависимости от степени близости к определенным классам.

Обучение без учителя в общем случае – это кластеризация документов (кластеры заранее неизвестны). Алгоритм определяет степень близости документов из коллекции и самостоятельно принимает решение об отнесении их к выявленным кластерам.

Решение задачи классификации начинается с предобработки коллекции документов, то есть приведения их к единому формату. Предобработка является одним из факторов, влияющих на точность классификатора, и заключается в применении следующих методов: удаление символов и цифр; удаление стоп-слов (предлоги, союзы, частицы, часто употребляемые слова); стемминг; лемматизация [1]. Для определения того, насколько эффективен тот или иной метод, или все сразу, необходимо экспериментировать с обучающей выборкой.

Большинство алгоритмов машинного обучения оперируют вещественно пространственными признаками документов или иными словами – векторной моделью. Векторная модель – это математическая модель представления текстов, в которой каждому документу $d \in D$, где $D = \{d_1, \dots, d_N\}$ – множество документов, сопоставлен вектор слов, состоящий из всех возможных слов коллекции. Таким образом, документ $d \in D$ представляет собой последовательность слов (термов) $T_d = \{t_1, \dots, t_{n_d}\}, n_d$ – длина документа d. Каждый терм t_i имеет вес w_{ij} по отношению к документу $d_j \in D$ [4]. Выбор модели представления данных является одним из факторов, влияющих на точность классификатора.

Bag of Words (от англ. – «мешок слов») – способ перевода текста в векторную модель, в которой порядок слов в документе не важен, а коллекцию документов можно представить в виде матрицы, строки которой являются отдельными документа, а столбцы – общий словарь слов коллекции. Значения пересечения строк и столбцов соответствует частотам вхождения слова (его вес w) в конкретном документе.

Bag of N-grams (от англ. – «*N*-грамма или группы по *N* элементов») – модель, в которой текст разделяется на последовательности из *N* слов. Если к тексту «Это текст для теста» применить биграммы ($N = 2$), то получим три группы: «это текст», «текст для», «для теста». Как и в случае с *Bag of Words*, считаем частоту вхождения *N*-грамм в документе.

Bag of Words & TF-IDF – наиболее популярный способ перевода текста в векторную модель. *TF-IDF* – это статистическая мера, используемая для оценки «важности» слова в контексте документа, который является частью коллекции документов, и вычисляется следующим образом:

$$TF - IDF(t, d, D) = TF(t, d) \times IDF(t, D) \quad (1)$$

TF – частота слова в документе, определяет его «важность» в пределах отдельного документа:

$$TF(t, d) = \frac{n_t}{\sum_k n_k}, \quad (2)$$

n_t – число вхождений слова t в документ, $\sum_k n_k$ – общее число слов в данном документе.

IDF – обратная частота документа, уменьшает вес широко употребляемых слов в коллекции документов:

$$IDF(t, D) = \log \frac{|D|}{|(d_i \supset t)|}, \quad (3)$$

$|D|$ – количество документов в коллекции, $|(d_i \supset t_i)|$ – количество документов, в которых встречается слово t_i.

Таким образом, каждый документ является вектором весов его термов $\vec{d_j} = \langle w_{1,j}, \dots, w_{|T|,j} \rangle$ где веса документов $0 \leq w_{ij} \leq 1$ для $\forall i, j : 0 \leq i \leq |T|, 0 \leq j \leq |D|$. То есть получаем матрицу весов слов в коллекции документов. Значение веса терма относительно мало, если он редко встречается в каком-то документе или встречается во многих документах, и относительно велико, если он часто встречается в небольшом числе документов, тем самым повышая степень их близости.

Bag of N-grams & TF-IDF – аналогичен методу *Bag of Words & TF-IDF*, лишь с той разницей, что вектор признаков содержит веса не только отдельных слов, а последовательностей из *N* слов.

Полученные наборы термов могут привести к ряду проблем из-за своего размера, связанных с низкой точностью классификатора из-за шумовых признаков или высокими вычислительными затратами. Шумовые признаки подразумевают единично встречающиеся термы во всей коллекции или неучтенные стоп-слова. Для устранения таких недостатков существуют методы отбора признаков для текстовых документов [1]. Одним из них, и вполне эффективным, является метод документной частоты (*DF*), суть которого заключается в определении порогового параметра τ, равного 1-5 документам, который используется для

сравнения числа документов $DF(t_i)$, в которых встречается терм t_i. Если $DF(t_i) \geq \tau$, тогда данный терм учитывается в дальнейшей работе алгоритма.

Существует множество алгоритмов для решения задачи классификации полнотекстовых документов с машинным обучением, часть из которых приведена в таблице 1. В данной статье рассматривается классический алгоритм классификации – много-классовая логистическая регрессия [2,3]. Логистическая регрессия – это статистический метод, который используется для прогнозирования вероятности исхода, и особенно популярен для задач классификации. Алгоритм прогнозирует вероятность возникновения события путем подгонки данных к логистической нелинейной функции сигмоиды.

Определение эффективности классификатора показывает насколько хорошо он справляется со своей задачей. Для этого используется два подхода – вычисление точности, как отношение числа правильно отнесенных документов к классам к общему числу документов в коллекции; оценочные метрики – точность, полнота и *F*-мера.

Первоначально, необходимо разделить обучающий набор в процентном соотношение, например, 70 на 30, где 70% – тренировочное множество, а 30% – тестовое множество [3]. После обучения алгоритма на тренировочном наборе, производится его оценка эффективности на тестовом множестве, для которого составляется матрица ошибок, пример которой приведен в табл. 2.

Таблица 2

Матрица ошибок

Оценка эксперта / Оценка алгоритма	$d_i \in c_j$	$d_i \notin c_j$
$d_i \in c_j$	TP	FP
$d_i \notin c_j$	FN	TN

Точность – доля документов истинно принадлежащих данному классу относительно всех документов, которые алгоритм отнес к классу:

$$P = \frac{TP}{TP + FP} \quad (4)$$

Для много-классовой классификации:

$$P_c = \frac{C_{c,c}}{\sum_{i=1}^{c} C_{c,i}} \quad (5)$$

где $C_{c,c}$ – значение TP для класса c_j, соответствующему оценке эксперта, а $\sum_{i=1}^{c} C_{c,i}$ – сумма значений TP и FP для класса c_j, соответствующему оценке алгоритма. Далее считаем среднее арифметическое для P_c.

Полнота – доля документов, отнесенных данному классу относительно всех документов:

$$R = \frac{TP}{TP + FN} \quad (6)$$

Для много-классовой классификации:

$$R_c = \frac{C_{c,c}}{\sum_{i=1}^{c} C_{i,c}} \quad (7)$$

где $C_{c,c}$ – значение TP для класса c_j, соответствующему оценке эксперта, а $\sum_{i=1}^{c} C_{c,i}$ – сумма значений TP и FN для класса c_j, соответствующему оценке алгоритма. Далее считаем среднее арифметическое для R_c.

F-мера – показатель, который находит баланс между полнотой и точностью, и сводит оценку классификатора к одной метрике [3]:

$$F = 2\frac{PR}{P + R} \quad (8)$$

F-мера дает понимание того, насколько хорошо обучен классификатор. На основе данной метрики можно судить о том, стоит ли использовать большую по объему выборку тренировочных примеров или изменить число параметров векторной модели, или существует необходимость оптимизации параметров алгоритма.

Постановка задачи. Пусть задано конечное множество классов $C = \{c_1, \ldots, c_n\}$, конечное множество документов $D = \{d_1, \ldots, d_n\}$ и неизвестная целевая функция Φ, которая для каждой пары <документ, класс> определяет, соответствуют ли они друг другу $\Phi : D \times C \rightarrow \{0, 1\}$. Для логистической регрессии мы рассчитываем вероятность принадлежности документа к классу, то есть целевая функция примет следующий вид: $\Phi : D \times C \rightarrow [0, 1]$. Задача состоит в том, чтобы найти максимально близкую к функции Φ функцию Φ', которую называют классификатором.

Из открытых источников [5] были скачаны судебные дела (табл. 3) для решения поставленной задачи. Размер коллекции составил – 600 документов, которые относятся к 3 классам, по 200 документов в каждом: уголовное, гражданское и административное производства первой инстанции; и 900 документов, которые относится к 9 классам, по 100 документов в каждом: уголовное, гражданское и административное производства для первой инстанции, апелляции и кассации.

После предварительной обработки, включающей удаление символов и цифр и исключение стоп-слов, были составлены векторные модели документов «мешка слов» и биграмм. Для каждого терма было рассчитано число вхождений в документы (по алгоритму *DF*), и отброшены те термы, частота которых составила менее 5. Таким образом получили обучающий набор, информация о котором приведена в табл. 3.

Таблица 3

Обучающий набор

Размер коллекции	Число классов	Векторная модель	Размер словаря до обработки (число слов)	Размер словаря после обработки (число слов)
600	3	*Bag of Words*	13129	5263
		2-grams	60117	11144
900	9	*Bag of Words*	13846	6734
		2-grams	65557	14481

Для «мешка слов» и биграмм веса термов определены, как отношение числа вхождений терма к размеру документа, а для этих же моделей, но с применением метода *TF-IDF*, веса определены по соответствующей формуле. Обработанные данные были случайным образом разделены на тренировочный и тестовый наборы с заранее размеченными классами. После чего был обучен алгоритм логистической регрессии на тренировочном наборе. Полученные веса функции сигмоиды использовались для проверки алгоритма на тестовом наборе.

Результаты классификации оценивались с помощью метрик точности и *F*-меры. Точность считалась, как доля верно классифицированных документов к размеру общему коллекции. Результаты работы алгоритма много-классовой логистической регрессии приведены в табл. 4.

Таблица 4

Результаты эксперимента

Размер коллекции →	600		900	
Векторная модель ↓	Точность	*F*-мера	Точность	*F*-мера
Bag of Words	1	1	0,9815	0,9823
Bag of Words & TF-IDF	0,9944	0,9942	0,9593	0,9634
2-grams	0,9944	0,9944	1	1
2-grams & TF-IDF	0,95	0,9526	0,9815	0,9821

Выводы: можно констатировать, что приведенные методы предобработки текстов, а также их последующее векторное представление позволяют эффективно производить классификацию с помощью алгоритма логистической регрессии, так как показатели точности и *F*-меры практически равны единице. Наилучшие показатели для коллекции с 600 документами были получены с моделью «мешок слов», а для коллекции с 900 документами – с моделью биграмм.

Рассмотренное поэтапное решение задачи классификации показывает стандартный подход к решению одной из задач обработки естественного языка. Произведено сравнение качества классификации на основе разных моделей представления данных. Показано, что логистическая регрессия справляется со своей задачей для классификации полнотекстовых документов. Дальнейшее решение задачи классификации судебных решений следуют протестировать на разных алгоритмах классификации с целью выявления лучшего из них или для составления «вероятностного классификатора» на основе нескольких алгоритмах.

Литература

1. *Большакова Е.И., Клышинский Э.С., Ландэ Д.В., Носков А.А., Пескова О.В., Ягунова Е.В.* Автоматическая обработка текстов на естественном языке и компьютерная лингвистика: учебное пособие. М.: МИЭМ, 2011. 272 с.
2. *Объедков Н.Ю., Турута Е.Н., Воронова Л.И.* Применение алгоритма логистической регрессии для классификации состояния структуры сенсорной сети – Студенческий научный форум, 2017 (https://scienceforum.ru/2017/2320/29613/).
3. *Воронова Л.И., Воронов В.И.* Machine Learning: Регрессионные методы интеллектуального анализа данных: учебное пособие. М.: МТУСИ, 2017. 81 с.
4. *Толмачев Р.В., Воронова Л.И.* Тематическая классификация статей новостного ресурса методами латентно-семантического анализа // Современные наукоемкие технологии, №3 (55-60), 2017. 134 с.
5. Открытая база данных судебных решений [Электронный ресурс]. Режим доступа – https://rospravosudie.com.

РАЗРАБОТКА АРХИТЕКТУРЫ СТАТИЧЕСКОГО АНАЛИЗАТОРА КОДА НА ЯЗЫКЕ PHP

Юсипов Евгений Александрович
студент группы М091601(72) МТУСИ
usupovzh@gmail.com

Кальфа Александр Алексеевич
МТУСИ, д.ф-м.н., профессор кафедры МСиУС
kalfa.alex@yandex.ru

Статический анализ исходного кода способен значительно облегчить и ускорить анализ безопасности приложения. Новые уязвимости постоянно обнаруживаются специалистами в области информационной безопасности. Поэтому архитектура статического анализатора должна быть легко расширяемой и не усложнять процесс добавления новых правил анализа. Для разработки такого сложного проекта, как статический анализатор необходимо предварительно спроектировать его архитектуру. В работе предложена простая, модульная, легко расширяемая архитектура статического анализатора кода на языке *PHP*. Для демонстрации предложенной архитектуры используется язык *UML*.

Ключевые слова: *статический анализ, архитектура ПО, уязвимости веб приложений, защита веб приложений, информационная безопасность, абстрактное синтаксическое дерево, язык PHP.*

За 2017 год в *PHP* было обнаружено 107 уязвимостей [1]. Большинство уязвимостей исправляется в последующих релизах *PHP*, однако по множеству причин не каждое Веб-приложение может быть обновлено. Вариантом решения проблемы является нахождение и исправление потенциально опасного кода в приложении. Подобный код может быть обнаружен либо путем ручного аудита кода приложения, либо автоматизированным статическим анализом кода. Статический анализ позволяет проанализировать код не выполняя его.

Важной частью разработки программного обеспечения является планирование его архитектуры. Часто в процессе разработки или эксплуатации программного обеспечения появляются новые требования или необходимость добавления функционала. Разработка программного обеспечения без предварительного планирования может привести либо к невозможности его изменения, либо цена таких изменений будет превышать цену его разработки с нуля. Программное обеспечение должно быть спроектировано так, чтобы оно было легко расширяемым без необходимости модифицировать уже существующий код [2].

Перед началом проектирования анализатора следует определить его входные данные, требуемые преобразования этих данных и вид предоставления результата. В качестве входных данных в данной работе приняты путь к анализируемому файлу и набор параметров. Выделение входных параметров в отдельный класс Input с методами *getFilePath* и *getParams* позволяет изолировать данные от метода их получения. Параметры хранятся в ассоциативном массиве с именем в качестве ключа. При этом не важно, получены данные через командную строку или через *HTTP* параметры, их всегда можно получить, используя метод *getParams*. В классе *Input* в дальнейшем можно будет реализовать проверку входных параметров и предзагрузку кода из файла.

Перед началом анализа может потребоваться ряд преобразований кода. Одним из них является преобразование кода в виде набора символов в абстрактное синтаксическое дерево (АСД). Для скрытия подробностей реализации его необходимо выделить в класс Parser, который будет использовать метод parse, принимающий текст программы. Метод должен возвращать список узлов АСД (рис. 1) с информацией об их содержимом. Реализация *Parser* может меняться в дальнейшем, в частности, он может быть сторонней библиотекой, интерфейс которой меняется со временем. Однако эти изменения не затронут код программы.

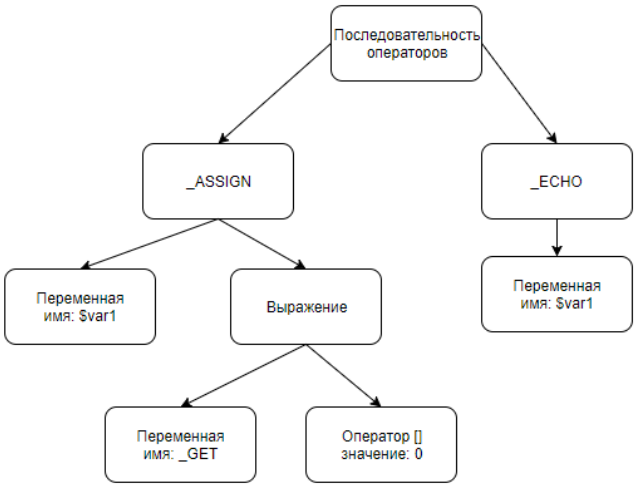


Рис. 1. Абстрактное синтаксическое дерево кода *$var1 = $_GET[0]; echo $var1;*

Для проведения анализа каждого узла АСД в него необходимо добавить информацию об окружении, в котором этот узел выполняется. Информацию об окружении узла хранит класс Environment. В программе в дальнейшем может быть несколько типов окружений, поэтому необходимо создать абстрактный класс, который будет содержать базовый набор полей и методов, от которого в дальнейшем будут унаследованы другие классы. Этот класс содержит ссылку на родительское окружение, массив переменных, которые в нем содержаться, методы создания дочернего окружения, получения и добавления переменных.

Для добавления информации об окружении в АСД необходимо создать отдельный класс *EnvironmentCreator*. Класс *EnvironmentCreator* реализует шаблон проектирования *Visitor* [3]. Главным методом класса является enterNode, который выполняется для каждого узла АСД. Если состояние окружения меняется (например, добавляется новая переменная), создается новое окружение, содержащее изменения (новая переменная). Предыдущее окружение добавляется в поле *$parent* нового окружения. Затем новое окружение записывается в текущий узел АСД. Если окружение не менялось, в текущий узел записывается старое окружение. Для экономии памяти, поле *$parent* должно быть ссылочного типа. Тогда в нем сохраняются не полные копии предыдущих окружений, а только адрес предыдущего окружения. На рисунке 3 показано окружение узла присваивания переменной *$var3* (окружение 3) в коде, состоящем из трех операций присваивания значений переменным *$var1*, *$var2*, *$var3*.

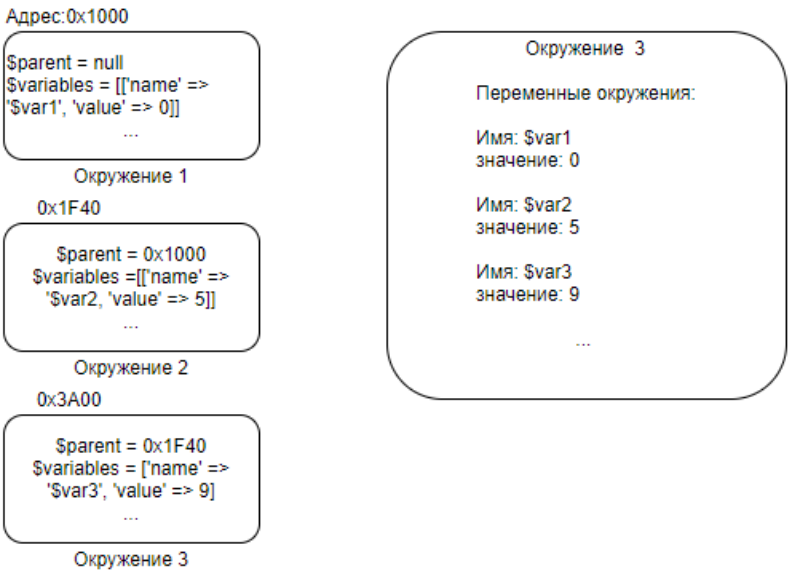


Рис. 2. Окружение 3 и его представление в памяти

В окружении так же можно хранить ссылку на узел АСД, чтобы не тратить вычислительные ресурсы на поиск узла при анализе. В этом случае узел, в котором переменная получила значение будет доступен из окружения узла, в котором используется переменная.

Использование потенциально уязвимой функции не всегда ведет к появлению уязвимости в коде. Преимуществом статического анализа перед поиском в коде списка потенциально уязвимых функций, таких как *print*, *echo*, *query*, является возможность анализа их выполнения в контексте программы. Для описания потенциально уязвимых частей программы необходимо создать отдельный класс Taint. В нем будет храниться информация о уровне уязвимости, функции, использованные для предотвращения уязвимости и другие данные.

Информацию о всех потенциально уязвимых местах в программе хранит класс *ProgramTaints*, который содержит ассоциативный массив, в качестве ключа в котором выступает имя переменной. Методы этого класса позволяют получать и устанавливать значение класса Taint для определенного окружения или переменной.

Информация о возможных источниках данных, таких как переменные или функции, хранится в массиве в классе *InputSources*. Любые данные, полученные от пользователя по умолчанию должны быть признаны небезопасными. Однако источники данных в приложении не ограничены пользовательским вводом. Информация может быть получена из базы данных или конфигурационных файлов. Эти источники часто уникальны для конкретного приложения и могут меняться в процессе разработки. Следовательно, их список должен быть легко расширяем. Класс *InputSources* создает единую единую точку приложения, где эти значения могут быть изменены.

Информация о функциях, которые предназначены для валидации входных данных, содержится в классе *SanitisingFunctions*. Кроме стандартных функций, преобразующих небезопасные данные, таких как *htmlspecialchars, striptags, mysql_real_escape_string,* разработчики создают свои функции, реализующие необходимую в каждом конкретном случае логику проверки. Как в случае с *InputSources,* этот класс позволяет создать единую точку, где могут быть добавлены или удалены такие функции. Класс содержит методы, возвращающие списки функций для разных типов уязвимостей.

Класс *Analyzer* содержит методы, осуществляющие проверку каждого узла АСД. Метод analyze анализирует узел в контексте его окружения. При обнаружении потенциально опасного кода он создает экземпляр класса Taint. Этот экземпляр добавляется к узлу АСД и в класс *ProgramTaints*. При обнаружении узла, для которого существуют уязвимости, вызывается метод *runChecks* класса *VulnerabilityScanner.*

Для того, чтобы разделить запуск проверок от их логики, создан класс *VulnerabilityScanner*. При необходимости, в конструкторе этого класса можно отключить инициализацию сканнеров некоторых типов уязвимостей, проверка которых не требуется. Например, SQL-инъекций, если приложение не работает с базами данных.

Логика проверок каждого типа уязвимостей содержится в классах, унаследованных от класса *VulnerabilityChecker.* Задавая этот класс как абстрактный, необходимо задать набор действий, которые обязательно должен выполнять дочерний класс проверки уязвимости. Любой дочерний класс, осуществляющий проверку уязвимости некоторого типа, должен осуществлять проверку наличия функций, защищающих данные от этого типа уязвимости, и проверку функций, которым передается переменная.

За получение результатов анализа отвечает метод *getReports* класса *VulnerabilityScanner.* Этот метод возвращает массив экземпляров класса *Report*. Не важно, выводятся ли результаты на экран, пишутся в лог или выдаются через *API* интерфейс. Для создания нового метода вывода достаточно создать класс, принимающий в качестве параметра в конструкторе экземпляр класса *Report*.

Класс *Report* содержит строковое представление информации об обнаруженной уязвимости и метод *getMessage*, позволяющий получить эту информацию.

В данной работе разработана модульная и расширяемая архитектура статического анализатора исходного кода для языка *PHP* (рисунок 3). Составлена диаграмма архитектуры на языке UML. Возможно добавление новых типов окружения путем наследования от абстрактного класса *Environment.* Одним из возможных дочерних классов является класс глобального окружения *GlobalEnvironment*. Он служит источником значений глобальных переменных, заданных до начала предполагаемого выполнения анализируемого кода.

Предусмотрено расширение класса *SanitisingFunctions*, если в процессе дальнейшей разработки программы его станет трудно поддерживать из-за увеличивающегося объема кода. В этом случае необходимо сделать его абстрактным и вынести код под конкретные типы уязвимостей в дочерние классы.

Предложенная архитектура допускает возможность добавления анализа новых типов уязвимостей путем наследования абстрактного класса *VulnerabilityChecker.* Путем добавления новых дочерних классов расширяется список типов уязвимостей, которые может находить программа.

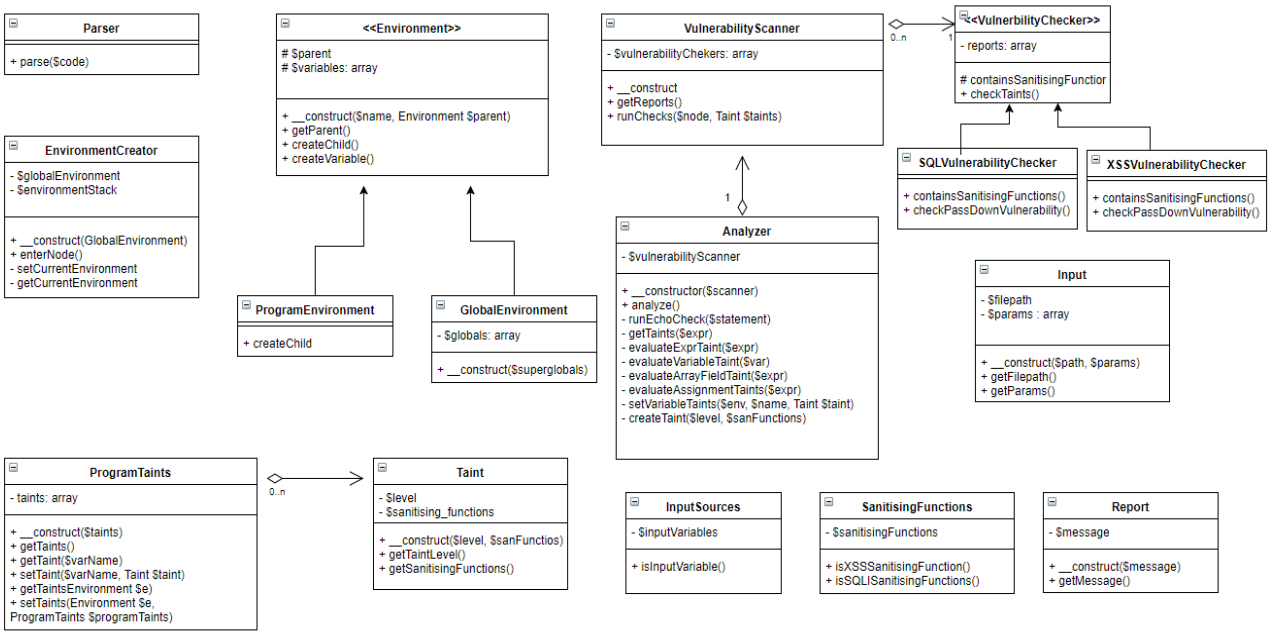


Рис. 3. Разработанная архитектура, описанная с помощью языка UML

Модульная архитектура программы позволяет менять конкретную реализацию модуля без влияния на остальной код, если не было изменений в интерфейсе. Реализация класса Parser может быть полностью изменена или заменена на другую стороннюю библиотеку, выполняющую функции создания АСД.

Возможность этих изменений доказывает правильное проектирование архитектуры программы.

Литература

1. Статистика *CVE* распределения уязвимостей языка *PHP* по годам (http://www.cvedetails.com/product/128/PHP-PHP.html?vendor_id=74).

2. *Meyer Bertrand.* Object-Oriented Software Construction. Prentice Hall, 1988.

3. *Э. Гамма, Р. Хелм, Р. Джонсон, Д. Влиссидес.* Приемы объектно-ориентированного программирования. Паттерны проектирования. Питер, 2013. 314 с.

ИССЛЕДОВАНИЕ ТЕХНОЛОГИИ ГЛУБОКОГО АНАЛИЗА ПАКЕТОВ DPI ДЛЯ ПРИМЕНЕНИЯ В КОРПОРАТИВНЫХ СЕТЯХ

Прохоров Даниил Олегович
студент группы М091601(72) МТУСИ
prokhorov.d.o@gmail.com

Беленькая Марина Наумовна
МТУСИ, доцент кафедры МСиУС
mn.belenkaya@mail.ru

Deep Packet Inspection — современная технология «глубокого анализа пакетов» по уровням модели OSI со второго по седьмой, предоставляющая, согласно стандарту ITU-T Y.2770, различные функции по работе с сетевым трафиком в инфокоммуникационных сетях. Существуют различные варианты архитектуры Deep Packet Inspection. Согласно стандарту, в типичную реализацию Deep Packet Inspection входят следующие функции: идентификации (классификации) потока и приложения, управления сигнатурами DPI, проверки трафика, представления отчетов, управления трафиком, идентификации сеанса, проверки шифрованного трафика, проверки сжатого трафика, обнаружения необычного трафика. Некоторые пункты стандарта, описывающие реализацию функций, являются рекомендациями. Технология DPI может использоваться как операторами связи, так и различными корпорациями. Для разных целей существуют решения Deep Packet Inspection разных сегментов производительности.

Ключевые слова: *DPI, Deep Packet Inspection, инфокоммуникации, Quality of Service, сетевой трафик, глубокий анализ, управление трафиком, идентификация трафика, фильтрация, Allot Communications, Procera Networks.*

DPI (Deep Packet Inspection, глубокий анализ пакетов) – технология накопления, проверки и фильтрации сетевого трафика. Термин «глубокий» означает, что анализ происходит со второго до седьмого уровня модели OSI.

При этом подразумевается возможность модификации, фильтрации или перенаправления трафика. Одним из важных отличий от предыдущих технологий анализа трафика является то, что системы на базе DPI могут принимать решения не только по содержимому пакета, но и по косвенным признакам, присущим отдельным сетевым программным продуктам и протоколам. Например, по частоте встречи определенных символов, длин пакетов, расстоянию между метками времени, то есть при помощи методов статистического анализа.

Технология DPI может применяться для управления угрозами и управления безопасностью [1], предотвращения утечки данных корпорации, защиты от отправки внутри корпорации защищенных от руководства файлов, целевого маркетинга на основе поведения пользователей, персонифицированной рекламы, исполнения законодательства в сфере информационных технологий.

6 декабря 2012 г. ITU-T был утвержден стандарт DPI Y.2770. На сегодняшний день DPI является стандартом де-факто для анализа трафика. При этом стандарт Y.2770 не разрешает доступ к личной информации пользователей.

Основной компонент DPI – классификация трафика в зависимости от целей применения. Она может осуществляться по типу протокола или приложения (Web, VoIP, PtP), конкретному протоколу уровня приложения (HTTP, BitTorent, SIP), приложению, использующему протокол (Google Chrome, Skype). Обычно DPI относят к средствам анализа отдельных пакетов без анализа нескольких пакетов одного потока (stateless). Но существует и DPI с хранением состояния и анализом содержимого (stateful).

В настоящее время в корпорациях применяется концепция “DPI как сервис”. То есть внедрение отдельных устройств, которые будут выполнять полный анализ сетевых данных и рассылать результаты анализа по всем устройствам в зависимости от их потребности.

Все системы DPI можно разделить на следующие группы:

1. **Standalone (автономные) системы DPI** – системы, работающие на специализированном оборудовании под управлением специализированных ОС. Это высокопроизводительные, сложные и дорогостоящие продукты.

2. Частный случай Standalone DPI - **программный DPI**. Это программные продукты, запускаемые под управлением универсальной ОС на выделенном сервере. Такие системы обычно малопроизводительны и подходят только для небольших кампусных сетей.

3. **Интегрированные системы DPI** – системы, поставляемые вместе с сетевым оборудованием. Такие решения обычно достаточно компромиссны и не могут предоставить весь спектр услуг.

Согласно стандарту Y.2770 [2], к функциональному объекту технологии DPI предъявляется ряд требований и рекомендаций, приведенных ниже (требования и рекомендации разделены по категориям выполняемых DPI функций).

Функция идентификации потока и приложения:

- Выполнение идентификации приложения.
- Поддержка различных видов правил политики DPI.
- Идентификация приложений путем проверки полезной нагрузки.
- Идентификация однонаправленных, а также двунаправленных приложений при условии, что одно направление трафика обеспечивает возможность однозначной идентификации.
- Идентификация приложений на основе двунаправленного трафика.

Функция управления сигнатурами DPI:

- Хранение сигнатур DPI в библиотеке сигнатур DPI (подобъект функционального объекта DPI).
- Ведение библиотеки сигнатур DPI с соблюдением мер безопасности, ее сокрытие от несанкционированных пользователей.
- Возможность добавления новых сигнатур в библиотеку сигнатур DPI.
- Возможность изменения (обновления), подключения и отключения, исключения (удаления) существующих сигнатур в библиотеке сигнатур DPI.
- Действия в рамках управления сигнатурами DPI должны осуществляться либо локально, либо дистанционно, либо обоими способами.
- Поддержка опросного режима в отношении операций над сигнатурами DPI в том случае, если операции инициируются функциональным объектом DPI локально.

Функция проверки трафика:

- Поддержка функциональным объектом DPI идентификации приложений без проверки на уровне потока (рекомендация).
- Первоначальная факультативная независимость сценария DPI от потока, возможность направления запроса о сборе информации о потоке.
- В таком запросе необходимо предоставление какого-либо ключа потока IPFIX.
- Поддержка идентификации приложений при наличии или отсутствии информации о стеке протоколов (рекомендации).
- Идентификация приложений при владении информацией о стеке протоколов IPv4 и IPv6, возможность дополнительной идентификации приложений при владении информацией о другом основном стеке протоколов.
- Идентификация приложений во вложенном трафике, например, инкапсулированном или туннелированном (рекомендация).
- После идентификации приложения - факультативное обеспечение возможности извлечения информации, относящейся к приложению.

Функция представления отчетов:

- Соответствие протокола экспорта спецификации IPFIX [IETF RFC 5101] [3] (рекомендация).
- При двунаправленных потоках – соответствие протокола экспорта спецификации [b-IETF RFC 5103] [3].
- Представление функциональным объектом DPI плоскости управления DPI информации о результатах проверки, а также информации, относящейся к потоку.
- Факультативное обеспечение возможности представления отчетов о новых, неизвестных или неправильных приложениях по результатам проверки трафика.
- Поддержка управляющих состояний, соответствующих IS (рабочее состояние) и OoS (нерабочее состояние) (рекомендация).

- Поддержка функции аварийного оповещения (рекомендация).
- Представление плоскости управления отчета об уровне загрузки ресурсных компонентов DPI (рекомендация).

Функция управления трафиком:

- Поддержка функциональным объектом DPI управления возможности участия в сетевых сценариях в целях управления трафиком, например, в функциях, определенных в [ITU-T Y.1221] [2] (рекомендация).
- Факультативная поддержка трафиком по умолчанию (рекомендация).
- Факультативная возможность взаимодействия с внешними функциями управления трафиком (рекомендация).

Функция идентификации сеанса:

- Возможность функционального объекта DPI проводить анализ режима сеанса (RTP, HTTP и т.д.).
- Возможность функционального объекта DPI отслеживать состояние сеанса.
- Факультативная возможность извлечения и создания данных измерений на уровне сеанса.

Функция проверки шифрованного трафика:

- Возможность факультативно проводить глубокий анализ пакетов в отношении нешифрованных элементов анализируемой информации (в зависимости от степени шифрования).
- В случае локального наличия используемого ключа шифрования DPI может факультативно применяться с предварительной операцией дешифрования локальной копии проверяемого пакета.
- Функциональный объект DPI факультативно имеет возможность идентификации как минимум потока шифрованного трафика IPSec.
- Функциональный объект DPI факультативно имеет возможность обнаружения трафика IPSec как в туннельном, так и в транспортном режимах.
- Факультативная возможность обнаружения трафика с защитой заголовком аутентификации (AH) или полезной нагрузки безопасности (ESP), основанная на соответствующем номере по протоколу IP.

Функция проверки сжатого трафика:

- Факультативное обеспечение глубокого анализа пакетов, подвергнутых сжатию, если локально имеется информация о примененном методе сжатия (проводится восстановление локальной копии пакета).
- Возможность получения информации из потока трафика о примененном методе сжатия (например, из элементов информации заголовка файла).

Функция обнаружения необычного трафика.

- Функциональный элемент DPI должен обеспечивать возможность обнаружения необычного трафика. То есть, сигнатуры DPI должны иметь возможность описывать как обычный, так и необычный трафик, анализируемый системой.

Все системы DPI осуществляют захват трафика либо при помощи стандартных сетевых адаптеров, либо при помощи адаптеров на базе специальных микросхемных решений ASIC (application-specific integrated circuit, интегральная схема специального назначения) или FPGA (field-programmable gate array, программируемая пользователем вентильная матрица), имеющих встроенные средства для проставления временных меток, аппаратной фильтрации, снятия заголовков низкоуровневых протоколов, выявления ошибочных или дублированных пакетов, балансирования нагрузки CPU с учетом IP-потоков.

Отдельной и очень обширной задачей является классификация трафика.

Она делается на основе:

- Вывода данных по двум параметрам. Используемые для вывода данные и используемые для их анализа алгоритмы. Данные анализируются или по характеристикам отдельных пакетов в рамках потока, или по характеристикам потока в целом. Алгоритмы анализа также делятся на два направления. Это сравнение с шаблоном и подход на основе машинного обучения и последующего распознавания. Ко второму относятся байесовские сети (Bayesian network, belief network), методы К-среднего, методы опорных векторов. Данные методы в свою очередь делятся на группы по методу обучения: классификация (обучение с учителем), кластеризация (обучение без учителя), ассоциирование, численное предсказание.
- На основе анализа сигнатур. При помощи либо поиска строк (прямой перебор, фильтры Блума), либо поиска регулярных выражений (описание при помощи регулярных языков в виде грамматик).
- Анализа данных в разных представлениях. Одни и те же данные могут быть по-разному закодированы в зависимости от протокола. Например, в коде ASCII и в коде Unicode. Возможны различные алгоритмы сжатия (gzip или deflate). Возможны различные алгоритмы шифрования (AES, RCU).

- Классификации угроз. Она осуществляется на основе статистического изучения аномалий. Вначале производится обучение системы на трафике, не содержащем атак, а затем изучается отклонение от нормальной картины реального трафика – статистическое детектирование аномалий.

Подключение систем DPI осуществляется либо как распределенной системы (анализаторы –probes и коллекторы), либо как беспроводной системы (перехват коммуникаций при достаточном уровне сигнала), либо как локальной системы (к конкретному кабелю или к точке единственного входа с методами зеркалирования, прокси или байпаса).

Решения компаний Procera и Allot представляют наиболее известные на рынке варианты DPI-систем. Их системы DPI проводят обмен только метаданными о трафике, что сводит к минимуму дополнительную нагрузку на сеть.

Procera Networks – американская компания, занимающаяся производством систем DPI и анализа данных, производит линейку автономных систем DPI PacketLogic. Архитектура их решения представлена на рис. 1.

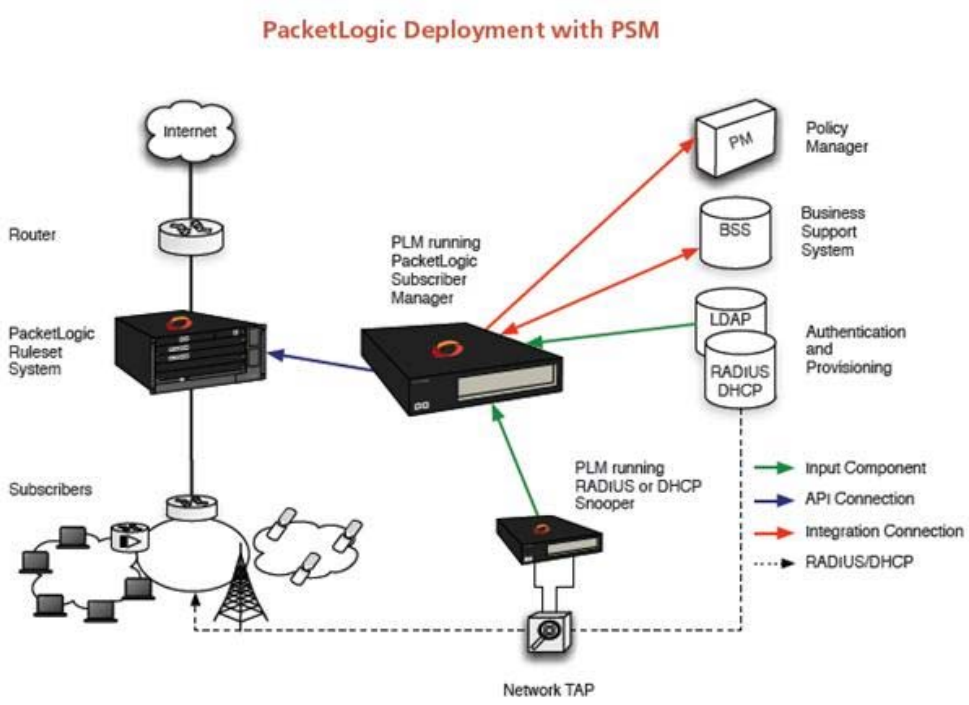


Рис. 1. Архитектура системы управления трафиком Procera Networks

Как видно на рис. 1, оборудование, выполняющее функции DPI (PacketLogic Ruleset System), устанавливается в разрыв канала для перехвата, обработки трафика и выполнения некоторых действий в соответствии с установленной политикой. Вспомогательные части, такие, как управление подписчиками, системы RADIUS или DHCP, соответствующие базы данных и менеджер применяемых политик, устанавливаются отдельно [4].

Allot Communications – израильская компания, разработчик решений по управлению трафиком для операторов связи и корпораций. Все платформы компании Allot используют собственную разработку в области DPI, называемую DART. Один из вариантов архитектуры их системы, осуществляющей управление трафиком, представлен на рис. 2.

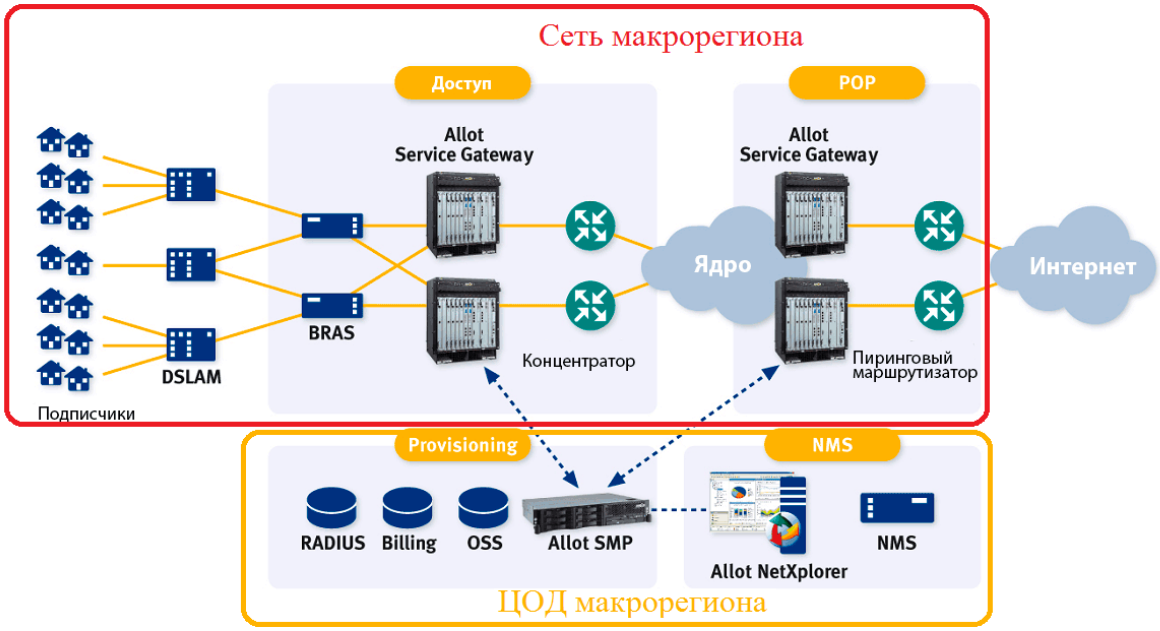


Рис. 2. Архитектура системы управления трафиком Allot Communications [5]

В данном примере архитектуры показан вариант установки DPI в инфокоммуникационной сети и центре обработки данных (ЦОД) макрорегиона. Как видно на рисунке 2, основные модули DPI Allot Service Gateway, осуществляющие отслеживание сетевого трафика и применение действий по его управлению, устанавливаются в разрывы всех имеющихся каналов связи. Один «ряд» устанавливается на уровне доступа, между маршрутизатором широкополосного удалённого доступа и маршрутизатором агрегации, второй «ряд» – непосредственно между выходом из «ядра» сети макрорегиона и пограничными маршрутизаторами, взаимодействующими с сетью Интернет. Модули одного «ряда» обмениваются между собой метаданными о трафике разных каналов. Модуль Allot SMP устанавливается в ЦОД и производит централизованное управление подписчиками, указывая модулям Service Gateway различные наборы правил для разных групп или конкретных подписчиков. Модуль Allot NetXplorer также устанавливается в ЦОД, с его помощью производится централизованное управление остальными модулями DPI (мониторинг, отчеты, уведомления, настройка политик и т.д.) [6].

У компаний-разработчиков DPI имеются решения, обладающие различными характеристиками, согласно которым можно условно разделить предлагаемые решения на младший, средний и старший сегменты. В таблице 1 приводится сравнение по некоторым основным характеристикам предлагаемых компаниями Allot Communications и Procera Networks популярных среди корпораций решений в младшем и среднем сегментах.

Таблица 1

Сравнительная таблица распространенных решений Allot и Procera по младшему и среднему сегментам

Характеристика	Младший сегмент		Средний сегмент	
	Allot NetEnforcer AC-500	Procera PacketLogic 7000	Allot Sigma E6	Procera PacketLogic 9000
Платформа	Собственная аппаратная платформа, Стандарт 1U 19" стоечным монтажом	Собственная аппаратная платформа, Стандарт 1U 19" стоечным монтажом	Собственная аппаратная платформа, Стандарт 6U 19" стоечным монтажом	Собственная аппаратная платформа, Стандарт 2U 19" стоечным монтажом
Пропускная способность	400 Мбит/с	1 Гбит/с	64 Гбит/с	120 Гбит/с
Максимальное количество сессий	400 тысяч	400 тысяч	20 миллионов	30 миллионов
Максимальное количество абонентов	32 тысячи	20 тысяч	3,2 миллиона	3 миллиона
Сетевые интерфейсы	2x1GbE	2x1GbE	8x10GbE	16x10GbE
Рабочая температура	От 0°C до 50°C	От 0°C до 40°C	От -5°C до 55°C	От 0°C до 40°C

Данные, приведенные в табл. 1, взяты из официальной документации к рассмотренным решениям и отражают значения характеристик в базовой поставке, без использования дополнительных модулей.

DPI-решения старшего сегмента нацелены, в первую очередь, на использование крупными операторами связи. Для использования различными компаниями внутри своих сетей подходят решения из среднего и младшего сегментов. Они имеют пропускную способность до 120 Гбит/с, до 30 миллионов сессий и до 3 миллионов абонентов [4, 6].

В качестве вывода можно отметить, что “DPI как сервис” является сегодня основной концепцией развития программно-аппаратных средств для анализа трафика. Возможен выбор существующего продукта или разработка своего продукта для конкретно поставленной задачи корпорации. В частных сетях компаний возможна реализация решений, способных обеспечить безопасность согласно принятым внутри этих компаний политикам безопасности.

Литература

1. *Беленькая М.Н., Малиновский С.Т., Яковенко Н.В*. Администрирование в информационных системах. Учебное пособие для вузов. М.: Горячая линия – Телеком, 2014. 400 с.

2. Recommendation ITU-T Y.1221. Traffic control and congestion control in IP-based networks / ITU-T. – 2010. – Geneva / Recommendation ITU-T Y.2770. Requirements for deep packet inspection in next generation networks / ITU-T. – 2013. – Geneva. – Режим доступа: http://www.itu.int/en/ITU-T/publications/Pages/recs.aspx.

3. IETF RFC 5101. Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information [Электронный ресурс] / B. Claise, Ed., Cisco Systems, Inc. – 2008 / IETF RFC 5103. Bidirectional Flow Export Using IP Flow Information Export (IPFIX) [Электронный ресурс] / B. Trammell, CERT/NetSA, E. Boschi, Hitachi Europe. – 2008. – Режим доступа: https://tools.ietf.org/, свободный.

4. Procera Networks, Inc. PacketLogic9000 Platform DataSheet [Электронный ресурс] / Procera Networks, Inc. – 2015. - Fremont, CA 94538.:Procera Networks, Inc. Corporate Office. – Режим доступа: https://www.proceranetworks.com/hubfs/Datasheets/Procera_DS_PacketLogic9000_Platform.pdf?hsCtaTracking=667e8b24-af1d-4106-a718-470d86501dfd%7C73fec71e-b686-4371-8188-0ae31403619c, свободный.

5. Глубокий анализ пакетов (DPI) как инструмент управления трафиком [Электронный ресурс] / ООО «ХОТЛАЙН». – Электрон. дан. – 2013. – Режим доступа: http://itc.ua/articles/glubokiy-analiz-paketov-dpi-kak-instrument-upravleniya-trafikom/, свободный.

6. Allot Communications. NetXplorer Data Sheet rev.10 [Электронный ресурс] / Allot Communications. – 2014 / Service Gateway Sigma E Data Sheet rev.5.1 10 [Электронный ресурс] / Allot Communications. – 2013. – Woburn, MA 01801.:Allot Communications North America Headquarters. – Режим доступа: http://www.allot.com/wp-content/uploads.

ПРОГРАММНАЯ РЕАЛИЗАЦИЯ КОДЕКА ХЭММИНГА НА ЯЗЫКЕ VISUAL BASIC

Липаткин Владислав Игоревич
Инженер 1-ой категории, НИЧ МТУСИ
lipatkin.24@gmail.com

Вакурин Илья Сергеевич
студент группы БСУ1501, МТУСИ
vort57@mail.ru

Мурашко Юрий Викторович
студент группы БПЗ 1501, МТУСИ
yurik.murashko@mail.ru

Приведено описание алгоритмов кодирования и декодирования кода Хэмминга. Проведен анализ данных алгоритмов. Раскрыты основные особенности приведенных алгоритмов. Произведен выбор языка программирования для реализации кодека. Кодек Хэмминга реализован на языке программирования Visual Basic. Приведены рекомендации по использованию разработанной утилиты.

Ключевые слова: *код Хэмминга, самокорректирующийся код, одиночная ошибка, проверочные символы, информационные символы, контрольный бит.*

В настоящее время не одна система связи не обходится без использования алгоритмов кодирования и декодирования информации. Кодирование повышает помехоустойчивость системы связи из-за внесения избыточности. К наиболее распространенным относят коды Рида- Соломона, коды основанные на длинных псевдослучайных последовательностях (Задова-Чу, М-последовательности и др.). В работе будет рассмотрен код Хэмминга, так он имеет наименьшее кодовое расстояние и способен исправлять однократные ошибки.

Код Хэмминга – это алгоритм, который позволяет закодировать какое-либо информационное сообщение определённым образом и после передачи (например, по сети) определить появилась ли ошибка в этом сообщении (к примеру, из-за помех) и, при возможности, восстановить это сообщение. Код Хэмминга относится к самокорректирующимся кодам. Самокорректирующиеся коды образуют большую группу из блочных, разделимых кодов (в которых все символы слова можно разделить на проверочные и информационные). Особенностью систематических кодов является то, что проверочные символы образуются в результате линейных операций над информационными символами. Кроме того, любая разрешенная кодовая комбинация может быть получена в результате линейных операций над набором линейно независимых кодовых комбинаций [1, 3].

Стоит отметить, что кодек Хэмминга состоит из двух частей. Первая часть кодирует исходное сообщение (кодер), вставляя в него в определённых местах контрольные биты (вычисленные специальным алгоритмом). Вторая часть (декодер) получает входящее сообщение и заново вычисляет контрольные биты (по тому же алгоритму, что и первая часть). Если все вновь вычисленные контрольные биты совпадают с полученными битами, то сообщение получено без ошибок. В противном случае, выводится сообщение об ошибке и при возможности ошибка исправляется [4].

Кодек Хэмминга работает по следующему алгоритму:

ASCII код каждого символа кодируется путем перевода в двоичную систему счисления (рис. 1)

Символ	ASCII код	Бинарное представление
м	236	11101100
т	242	11110010
у	243	11110011
с	241	11110001
и	232	11101000

Рис. 1. Кодирование символов

Далее каждый символ (информационное слово) кодируется независимо друг от друга. К закодированной букве добавляются контрольные биты (рис. 2) в зависимости от длинны информационного слова. Возьмем для примера букву «м» и допустим, что длинна этого информационного слова 8 бит. Теперь на строго определенные позиции равные степеням двойки вставляем контрольные биты, в данном случае это 1,2,4,8. Теперь сообщение увеличится на 4 бита, до вычисления контрольных бит и присваивается значение «0».

м
11101100
001011001100

Рис. 2. Вставка контрольных бит

Теперь необходимо вычислить значение каждого контрольного бита (рис.3). Значение каждого контрольного бита зависит от значений информационных бит, но не от всех, а только от тех, которые этот контрольных бит контролируют. Для того чтобы понять, за какие биты отвечает каждых контрольный бит необходимо понять очень простую закономерность: контрольный бит с номером N контролирует все последующие N бит через каждые N бит, начиная с позиции N.

1	2	3	4	5	6	7	8	9	10	11	12	
0	0	1	0	1	1	0	0	1	1	0	0	
												1
												2
												4
												8

Рис. 3. Вычисление контрольных бит

Коричневым цветом обозначены те биты, которые контролирует контрольный бит, номер которого справа. То есть, к примеру, бит номер 3 контролируется битами с номерами 1 и 2. Ясно, что чтобы узнать какими битами контролируется бит с номером N надо просто разложить N по степеням двойки. Далее вычисляем значение контрольных бит. Для этого нужно взять каждый контрольный бит и посмотреть сколько единиц он контролирует, если число единиц чётное, то ставится ноль, в противном случае единица. После вычисления контрольных бит получаем следующие (рис. 4):

м
001111011100

Рис. 4. Информационное слово после вычисления контрольных бит

На этом процесс кодирования завершается.

Допустим теперь, что переданное сообщение поступило к нам с ошибкой в одиннадцатом бите, заменив «м» на «о» (рис. 5).

о
001111011110

Рис. 5. Сообщение с ошибкой

Процесс декодирования заключается в том, что необходимо заново вычислить все контрольные биты (так же как и при кодировании) и сравнить их с контрольными битами, которые были получены в декодере (рис. 6).

о
111111001110

Рис. 6. Сообщение с несовпадающими контрольными битами

Контрольные биты под номерами: 1, 2, 8 не совпадают с такими же контрольными битами, которые были получены в декодере. Теперь просто сложив номера позиций неправильных контрольных бит (1 + 2 + 8 = 11) получаем позицию ошибочного бита. Теперь просто инвертировав его и отбросив контрольные биты, мы получим исходное сообщение.

Согласно вышеописанному алгоритму была провидена программная реализация кодера и декодера кода Хэмминга на языке программирования Visual Basic [2]. В настоящие время существует большое количество языков программирования. И все они в полной мере подходят для реализации поставленной задачи. Наиболее популярным является язык программирования C/C++. Его преимущество заключается в возможности работы на низком уровне с памятью, адресами, портами. Однако данное преимущество, также можно расценить как недостаток. Ведь неаккуратное использование языка может привести к появлению нежелательных ошибок с памятью. В свою очередь язык программирования Visual Basic первоначально разрабатывался для начинающих программистов, поэтому написание программ на Visual Basic максимально приближено к естественному языку (английскому). В связи с этим обучиться этому языку гораздо проще, чем обучиться языку C/C++. Visual Basic не имеет типа данных «указатель», таким образом, программа на VB защищена от ошибок, связанных с неправильным использованием указателей. Что в свою очередь, помогает избежать появления ошибок с памятью. Также в поставленной задаче необходим интерфейс пользователя для взаимодействия оператора с программой. В языке Visual Basic максимально упрощено создание интерфейса будущих приложений, так как в нем поддерживаются средства быстрой разработки. Таким образом, в качестве языка программирования для реализации кодека Хэмминга был выбран Visual Basic. На рисунке 7 приведен интерфейс кодека Хэмминга.

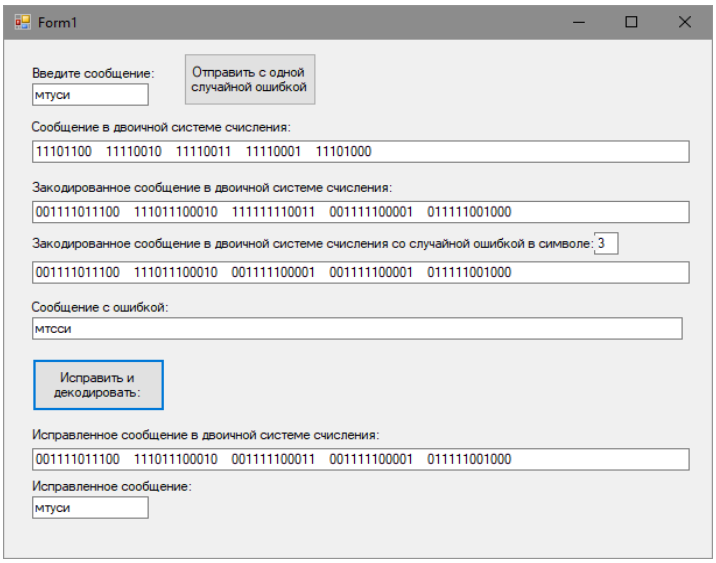


Рис. 7. Интерфейс программы

В программу вводится слово «мтуси», оно кодируется, а именно, добавляются контрольные биты, после чего в него вносится ошибка в случайный символ, в данном случае символ под номером 3. После этого сообщение декодируется, путем проверки контрольных бит. Как видно из рисунка 7, ошибка успешно исправлена декодером.

Из выше сказанного можно сделать вывод, что кодек Хэмминга наилучшем образом подходит для кодирования передаваемых данных в канале связи, в котором наблюдаются одиночные ошибки. Разработанное приложение можно использовать в учебном процессе, например, в лабораторных работах по изучению кода Хэмминга для проверки ручного расчета в рамках курсов по помехоустойчивому кодированию.

Литература

1. *Морелос-Сарагоса, Р.* Искусство помехоустойчивого кодирования. Методы, алгоритмы, применение. М.: Техносфера, 2006. 320 с.
2. *Брайан Сайлер, Джефф Спотс*. Использование Visual Basic 6.0. СПб: Москва – СПб – Киев, 2008.
3. *Пипп П.А., Фролов А.А.,* Исследование эффективности применения LDPC-кодека в системе телевизионного вещания DVB-T2 // Телекоммуникации и информационные технологии. №1. 2017. С. 19-23.
4. *Золотарев В.В., Овечкин Г.В.* Помехоустойчивое кодирование. Методы и алгоритмы. Справочник. М.: Горячая линия – Телеком, 2004.

КЛАССИФИКАЦИЯ СЕТЕВОГО ТРАФИКА С ПРИМЕНЕНИЕМ АЛГОРИТМОВ МАШИННОГО ОБУЧЕНИЯ ВМЕСТО DPI

Кротов Артем Витальевич
Студент группы М091701(70) МТУСИ
akrotov.work@gmail.com

Жуков Герман Васильевич
МТУСИ, к.т.н., доцент кафедры МКиИТ
zhukov@srd.mtuci.ru

Для оценки эффективности алгоритмов классификации в режиме обучения и тестирования с помощью разработанного программного комплекса была сформирована база данных трафика следующих приложений: «Skype», «VK», «Google Play», «Youtube», «Google Chrome», «Instagram», «Shazam», «Google Maps», «Twitter», «HearhStone». В данной работе рассматривались исключительно TCP сегменты. Полученная выборка пакетов была сгруппирована в потоки, которые в дальнейшем были разделены на обучающую и тестовую выборки в соотношение 2 к 1. Учитывая то, что многие атрибуты, используемые для классификации исследуемого сетевого трафика, не несут значимого информационного выигрыша и влияют на эффективность классификации, выбран уникальный алгоритм машинного обучения «Случайный лес».

Ключевые слова: *классификация сетевого трафика, исследование сетевого трафика, машинное обучение, случайный лес, сетевой пакет, модель OSI, обучение с учителем, атрибуты классификации, обучающая выборка, тестовая выборка.*

Постановка задачи

Система DPI выполняет глубокий анализ всего проходящего через неё трафика, используя для анализа каждых проходящий через нее пакет. Термин «глубокий» подразумевает анализ пакета на верхних уровнях модели OSI, а не только по стандартным номерам портов. Помимо изучения пакетов по неким стандартным паттернам, система DPI осуществляет поведенческий анализ трафика, который позволяет распознать приложения, не использующие для обмена данными заранее известные заголовки и структуры данных [1].

Система DPI, как правило, устанавливается на границе внутренней (локальной) и внешней сетей. Тем самым, весь трафик, который покидает или входит в сеть оператора, проходит через DPI, что даёт возможность его мониторинга и контроля.

Основная проблема всех существующих решений DPI заключается в том, что для того, чтобы однозначно определить принадлежность того или иного потока данных к одному из сетевых приложений, устройство, осуществляющее анализ трафика, должно «увидеть» оба направления сессии. Иными словами, входящий и исходящий трафик в пределах одного потока должны пройти через одно и то же устройство. Если оборудование «понимает», что «видит» только одно направление в рамках сессии, оно не имеет возможности соотнести данный поток с какой-либо известной категорией трафика. В связи с этим, когда речь заходит о контроле границе между внутренней (локальной) и внешней сетями, встаёт вопрос об асимметричном трафике. Одно из решений данной проблемы предложила компания «Sandvine». Весь трафик, являющийся асимметричным, пересылается на все устройства DPI, находящиеся в едином домене. В итоге данной пересылки устройства, «видевшие» до этого лишь одно направление в рамках сессии, «увидят» и второе, на основании чего можно будет осуществить полный комплекс мер по анализу и управлению трафиком. Недостаток данной схемы очевиден — при больших объёмах асимметричного трафика на сети предъявляются серьёзные требования к каналам связи, соединяющим устройства DPI на разных сайтах. [1]

При классификации сетевого трафика важнейшую часть составляет определение атрибутов классификации, для выделения которых следует сформировать весомую часть исходных данных.

Процедуру классификации можно разделить на две части: выделение атрибутов классификации и собственно дальнейшая классификации с выбором оптимального алгоритма машинного обучения (В дальнейшем МО). Методы МО можно разделить на две группы. Первая – это обучение «с учителем». Вторая – обучение «без учителя». [2] При обучении с учителем имеется множество ситуаций и множество возможных решений. Обучение классификатора производится с заранее подготовленными типовыми примерами с от-

ношением «ситуация» - «решение», в то время как «обучение» подразумевает исключительно набор «сырых» данных.

Исходные данные

Для получения исходного трафика использовалось мобильного устройство под управлением операционной системы «Android». В течении недели на исследуемом устройстве использовались следующие приложения: «Skype», «VK», «Google Play», «Youtube», «Google Chrome», «Instagram», «Shazam», «Google Maps», «Twitter», «HearhStone». Для журналирования трафика было использовано приложение «Shark» (аналог «Wireshark» для мобильных устройств под управлением операционной системы «Android»), с формированием записей в *.pcap формате) .

Для анализа было подготовлено два набора данных, содержащих сетевые пакеты, разбитые по потокам. Классифицированные потоки были разделены следующим образом: 66% от исходных данных использовались как обучающий набор, остальные 34% – для тестирования и его оценки. Полученные в результате измерений дампы трафика приведены в табл. 1.

Таблица 1

Полученные дампы трафика

Дамп трафика	Количество потоков сетевого трафика по группам										Общее кол. потоков
	Skype	VK	Google Play	YouTube	Google Chrome	Instagram	Shazam	Google Maps	Twitter	HearthStone	
Обучающий	6009	12098	7086	11416	8372	10654	3746	2249	3114	11416	76160
Общий тестовый	3004	6040	3542	5707	4185	5327	1872	1124	1557	5707	38074
Skype тестовый	3004	-	-	-	-	-	-	-	-	-	3004
VK тестовый	-	6040	-	-	-	-	П	-	-	-	6040
Google Play тестовый	-	-	3542	-	-	-	-	-	-	-	3542
YouTube тестовый	-	-	-	5707	-	-	-	-	-	-	5707
Google Chrome тестовый	-	-	-	-	4185	-	-	-	-	-	4185
Instagram тестовый	-	-	-	-	-	5327	-	-	-	-	5327
Shazam тестовый	-	-	-	-	-	-	1872	-	-	-	1872
Goggle Maps еустовый	-	-	-	-	-	-	-	1124	-	-	1124
Twitter тестовый	-	-	-	-	-	-	-	-	1557	-	1557
HearthStone тестовый	-	-	-	-	-	-	-	-	-	5707	5707

Выбор атрибутов классификации

Качество атрибутов является одним из наиболее важных этапов создания модели классификатора. Использование избыточного количества атрибутов или атрибутов плохого качества может привести к переобучению модели, снижению точности тестовых данных, чрезмерному увеличению времени обучения и созданию модели классификатора [3].

Экспериментальным путем были выделены следующие метрики потока: «IP клиента», «IP сервера», «Порт клиента», «Порт сервера», «Размер заголовка на транспортном уровне», «Размер данных на транспортном уровне», «Размер заголовка на сетевом уровне», «Размер данных на сетевом уровне» [4].

Данный ряд экспериментальных метрик очень хорошо характеризует поток данных, на его основе мож-

но точно предсказать искомое приложение. Исходя из этого, можно сформировать следующие статистические характеристики потока данных: «Адрес клиента (сервера)», «Порт клиента (сервера)», «Время жизни потока», «Средний размер пакета со стороны клиента (сервера)», «Стандартное отклонение размера пакета со стороны клиента (сервера)», «Среднее число пакетов на порцию данных со стороны клиента (сервера)», «Соотношение переданных байт между клиент-сервером», «Средний размер порции данных со стороны клиента (сервера)», «Стандартное отклонение размера данных со стороны клиента (сервера)», «Соотношение полезной нагрузки между клиент-сервером», «Соотношение пакетов между клиент-сервером», «Общее количество байт со стороны клиента (сервера)».

Выбор алгоритма классификации

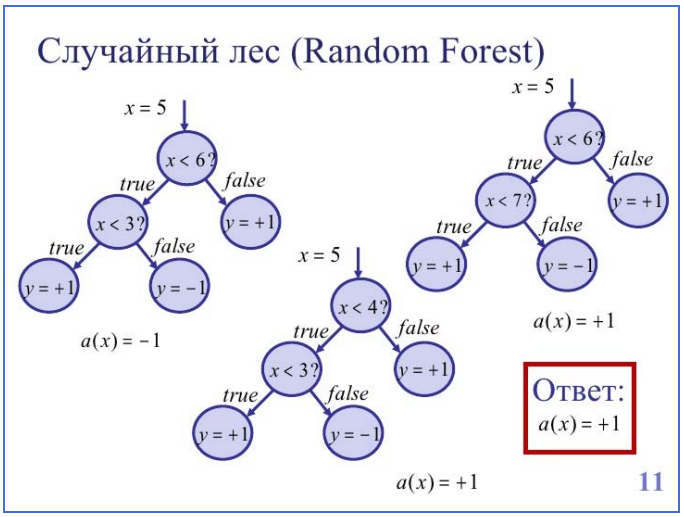


Рис. 2. Семейство деревьев алгоритма случайный лес

Исходя из количества сформированных атрибутов классификации, был выбран один из алгоритмов машинного обучения «Случайный Лес» (рис. 1). Он является семейством решающих деревьев, позволяющий увеличить процент точности по сравнению с классификацией по одному дереву, тем самым, решив проблему переобучения.

Процесс классификации представляет компоновку «решений» множества деревьев; важно заметить, что обучение каждого из «деревьев» происходит независимо от других «деревьев», тем самым решается вышеупомянутая проблема переобучения.

Результатом классификации будет тот класс, который «набрал» больше всего голосов (одно дерево – один голос). К примеру, если в поставленной задаче была сформирована модель, состоящая из пятисот деревьев, среди которых сто указывают на первый класс, а остальные четыреста на второй класс, то на выходе алгоритма (модели) будет представлен именно второй класс. Алгоритм «Случайный лес» потребляет весьма большое количество ресурсов системы, что может отразиться на ее производительности. Один из способов решения данной проблемы – ограничение глубины анализа (количество уровней дерева), но данное действие уменьшит точности оценки, так как для решения реальных задач, как правило, нужно строить достаточное количество именно «глубоких» деревьев. Также можно отметить, что время обучения прямо пропорционально их количеству (линейная зависимость). Данный алгоритм машинного обучения работает согласно принципу «обучение с учителем». Полученная выборка состоит из обучающей (66%) и тестовой (33%) выборок. [5]

Результаты классификации

Рассмотрим результаты классификации приложений на этапе тестирования алгоритма «Случайный Лес». В результате для десяти приложений с помощью этого алгоритма было создано 500 деревьев. Затем они сравнивались между собой и определялась их принадлежность к тому или иному классу. После обучения подготовленным набором и тестированием были получены результаты (табл. 2).

Таблица 2

Результаты тестирования алгоритма «Случайный лес»

Дамп тесто-вого трафика	Skype	VK	Google Play	YouTube	Google Chrome	Исходное	Ошибка, %
Skype	2913	52	-	-	-	3004	3,029
VK	-	5657	28	14	47	6040	6,341
Google Play	-	12	3344	16	129	3542	5,59
YouTube	-	5	29	5637	24	5707	1,227
Google Chrome	-	19	7	35	4091	4185	2,246
Instagram	-	243	-	14	22	5327	6,251
Shazam	19	13	-	6	-	1872	2,083
Google Maps	-	-	9	-	12	1124	1,868
Twitter	-	31	2	-	6	1557	5,78
HearthStone	-	-	16	-	2	5707	0,315

Дамп тестового трафика	Instagram	Shazam	Google Maps	Twitter	HearthStone	Исходное	Ошибка, %
Skype	-	39	-	-	-	3004	3,029
VK	129	-	-	165	-	6040	6,341
Google Play	8	-	12	7	-14	3542	5,59
YouTube	-	-	-	-	-	5707	1,227
Google Chrome	17	12	5	9	2	4185	2,246
Instagram	4994	-	-	54	-	5327	6,251
Shazam	-	1833	-	-	1	1872	2,083
Google Maps	-	-	1103	-	-	1124	1,868
Twitter	51	-	-	1467	-	1557	5,78
HearthStone	-	-	-	-	5689	5707	0,315

Наглядное представление результатов тестирования изображено на диаграмме зависимости ошибки тестирования от конкретного приложения (рис. 2), из которого видно, что наибольшая ошибка возникает при работе с приложениями «VK» – 6,341% , «Instagram» – 6,251%, «Twitter» – 5,78%.

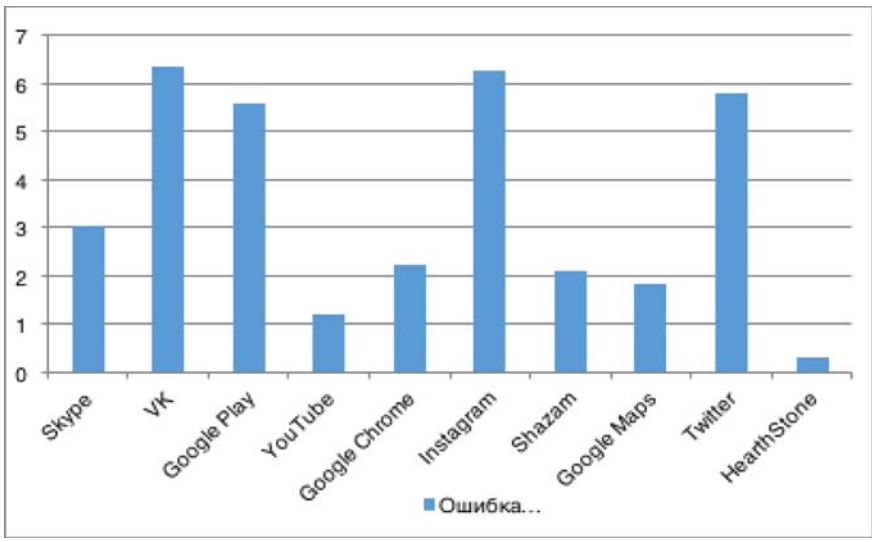


Рис. 3. Зависимость ошибки классификация от трафика приложения

Заключение

Для получения требуемых исходных данных использовалась сеть, состоящая из одного мобильного устройства с выходом в сеть интернет. Для исследования данных были сформированы обучающая и тестовая выборки трафика по таким приложениям, как: «Skype», «VK», «Google Play», «YouTube», «Google Chrome», «Instagram», «Shazam», «Google Maps», «Twitter», «HearthStone». На основе экспериментальных данных было сформировано 20 аттрибутов классификации. Учитывая их количество был выбран алгоритм машинного обучения – «Случайный лес» На этапе тестирования было показано, что данный алгоритм успешно справился со своей задачей.

Литература

1. *Эд Уилсон.* Мониторинг и анализ сетей. Методы выявления неисправностей. М.: Издательство «Лори», 2004.
2. *Петер Флах.* Машинное обучение. Наука и искусство построения алгоритмов, которые извлекают знания из данных. М.: Издательство «ДМК», 2015.
3. *Шелухин О.И., Калугин Ю.А*. Влияние прореживания пакетов на качество классификации потоков сетевого трафика методами машинного обучения. Нейро-компьютеры: разработка, применение, 2016.
4. *Щербакова Н.Г* Анализ IP-трафика методами Data Mining. Проблема классификации. Проблемы информатики, 2012.
5. *Ian H. Witten, Eibe Frank, Mark A. Hall, Christopher J. Pal* Data Mining, Fourth Edition: Practical Machine Learning Tools and Techniques 4th Edition. Burlington, Massachusetts, Morgan Kaufmann Publishers, 2015.

БАЗА ДАННЫХ ЭЛЕКТРОННЫХ КОМПОНЕНТОВ ДЛЯ АВТОМАТИЗАЦИИ СХЕМОТЕХНИЧЕСКОГО СИНТЕЗА РАДИОТЕХНИЧЕСКИХ УСТРОЙСТВ

Балашов Виталий Олегович
Группа: МИТ1702, каф.РОС
balash1996@list.ru

Долин Георгий Аркадьевич
МТУСИ, к.т.н., доцент кафедры РОС
dolin1974@gmail.com

Рассмотрены вопросы реализации распределенной базы данных компонентов РТУ на языках Embarcedero Delphi XE 10.2 и SQL. Проведенное обоснование показало целесообразность создания подобной программы на ЭВМ. Разработанная программа позволяет создавать новые базы данных компонентов, модифицировать компоненты, осуществлять поиск требуемых в ходе проектирования систем компонентов по задаваемым критериям, использовать сторонние базы данных. Показано, что разработанная распределенная реляционная динамическая БД параметров электронных компонентов систем может быть использована как в ходе автоматизированного проектирования устройств, так и как справочник при их ручном проектировании.

Ключевые слова: *САПР, проектирование, радиотехника, база знаний, экспертная система, анализ, синтез, схемотехника, БД, БЗ.*

Термин информационная система относится к классу программных продуктов, облегчающих или автоматизирующих производство. Систему называют информационной, если она поддерживает информационную поддержку производства. Соответствующая программа называется системой, если последовательно или параллельно выполняет более одной функции, например, позволяет осуществлять синтез и анализ РТУ и производить манипуляции с электронными компонентами, входящими в БД программы. В случае создания системы проектирования РТУ невозможно обойтись без использования БД.

В большинстве случаев для создания собственной информационной системы нельзя обойтись без использования БД, которой управляет специальная системная программа (СУБД) [2]. Например, в случае БД электронных компонентов, управляющая оболочка должна знать, что для всех компонентов, перечисленных в общей номенклатуре, должно быть представлены все их характеристики. Такого типа свойства называются целостностью БД. При создании БД информационной системы разработчик сообщает СУБД, какого рода ограничения целостности система должна поддерживать в БД, а далее ответственность берет на себя СУБД, без требования вмешательства прикладной программы. Обычно механизм обеспечения целостности БД интегрируется с механизмом управления транзакциями – последовательностью операций модификации БД, воспринимаемыми СУБД как одна атомарная операция.

Второй важной особенностью СУБД является обеспечение незапланированных (ad hoc) запросов к базе данных. Например, при проектировании информационной системы, предназначенной для автоматизации запросов в процессе синтеза и анализа РТУ, было запланировано выполнение запросов о наличии электронных компонентов в БД, операций модификации данных при вводе новых типов электронных компонентов или изменении параметров старых, а впоследствии возникла необходимость в информации о выборе электронных компонентов с определенными параметрами. При отсутствии СУБД понадобилась бы переделка всей информационной системы. Однако СУБД, обладая знаниями о предметной области (например, о структуре и смысле данных ИС электронных компонентов), может обеспечить универсальный язык запросов (обычно SQL), позволяющий сформулировать произвольный запрос на выборку информации из соответствующей БД. Такой запрос может быть в любой момент подан с терминала (без участия ИС) или встроен в одну из прикладных программ, входящих в информационную систему.

Не менее важно и то, что большинство СУБД способно обеспечить режим мультидоступа. Сегодня развитые компьютерные архитектуры обычно относятся к одной из двух категорий или к их комбинациям: информационно-вычислительный сервер с подключенными к нему терминалами или распределенная сеть серверов и клиентских рабочих станций, обеспечивающая совместное использование ресурсов. Соответст-

венно, ИС должна иметь возможность квазипараллельно (конечный пользователь не должен замечать задержки ответа) выполнить операции, задаваемые несколькими пользователями одновременно. При этом несколько параллельных транзакций выполняются последовательно. Подавляющее большинство современных СУБД поддерживают эту возможность, избавляя разработчиков информационных систем от необходимости заботиться об обеспечении режима мультидоступа.

Кроме того, довольно часто приходиться решать задачи, которые трудно реализовать, даже если система опирается на какую-либо современную СУБД, а именно: проектирование и разработка логической структуры самой информационной системы как набора программ, проектирование лежащей в основе общего проекта ИС БД, проектирование и разработка интерфейсных подсистем, как тех, которые относятся к взаимодействиям ИС с конечным пользователем, так и тех, что связывают прикладные программы с СУБД. И если проектируемая ИС, которой и является система синтеза и анализа РТУ, то решение подобных задач вручную, без привлечения программных ИС, как правило, превышает человеческие возможности.

Современные средства, служащие инструментом при проектировании и разработке ИС: системы класса CASE (Computer Aided Software Engineering), ориентированные на поддержку разработки ИС и программные средства (часто интегрированные с CASE-системами) языков четвертого поколения (4GL).

Применение автоматических методов проектирования РТУ требует введения в БД САПР РТУ набора параметров электронных компонентов. Это связано с тем, что ручные методы проектирования не накладывают особых ограничений на содержание и форму записи исходной информации. В то время как автоматические методы проектирования РТУ предъявляют более жесткие требования к форме записи информационной базы по составу, объему и степени формализации [3], что существенно увеличивает сложность и объем БД. Кроме того, ее формирование осложняется тем, что БД должна быть доступна при проведении проектирования РТУ как на функциональном (для структурных и функциональных схем), так и на схемотехническом (для принципиальных и эквивалентных схем) уровнях. А результаты проектирования РТУ в ЭС могут быть также сохранены в БД и использованы при синтезе радиотехнических систем (РТС).

Это вызывает необходимость обеспечить единство обобщенного описания узлов РТУ на функциональном и схемотехническом уровне, представленных в четкой структурной форме, не допускающей двоякого толкования представления разрабатываемого РТУ. Поэтому для проектирования и разработки БД САПР РТУ выбрана реляционная модель представления данных [1], отвечающая постулатам Кодда. Эта модель, контролируя ссылочную целостность, обеспечивает безошибочный ввод данных; экономит дисковое пространство, за счет исключения дублирования данных; логично и гибко организует данные за счет связей «один-ко-многим».

Содержимое БД компонентов РТУ представляет собой совокупность информации из различных источников. Организация быстрого доступа к ней и устранение ее несогласованности является сложной проблемой, которая решается при использовании двухуровневого построения системы, т.е. первичные исходные знания, заложенные в БД разработчиками при ее создании, хранятся отдельно от новой информации, добавляемой и хранимой во вторичных БД. Операции модификации первичных данных могут иметь ограниченную область действия или вообще могут быть запрещены, например, это может потребоваться при использовании подобных программ в учебных целях, когда с ними работают студенты [4].

Кроме того, в ходе проектирования РТУ БД обеспечивает независимость данных от процессов их обработки в САПР, что позволяет изменить, при необходимости, структуру и состав БД без внесения исправлений в работающие с ней программы и наоборот. Использование БД разными группами разработчиков, работающих с единым сервером данных, должно быть полностью отделено от процессов управления данными и их защиты. В БД состав элементов и связей между ними определяется до начала разработки прикладных задач и, следовательно, структура БД не испытывает навязываемых последними ограничений. Это достигается путем использования языка SQL [1] и универсальной модели доступа к данным через драйверы ODBC или ADO, с возможностью доступа к ним из локальных и распределенных сетей (например, в ходе проектирования РТУ можно использовать параметры электронных компонентов, размещенные компаниями – производителями в Интернет).

Электронный компонент РТУ, например, «транзистор» должен представляться в БД как отношение совокупности таких атрибутов, как тип, электрические и предельные эксплуатационные параметры, параметры математической модели, условное графическое отображение (УГО) и т. д. (см. рис. 1).

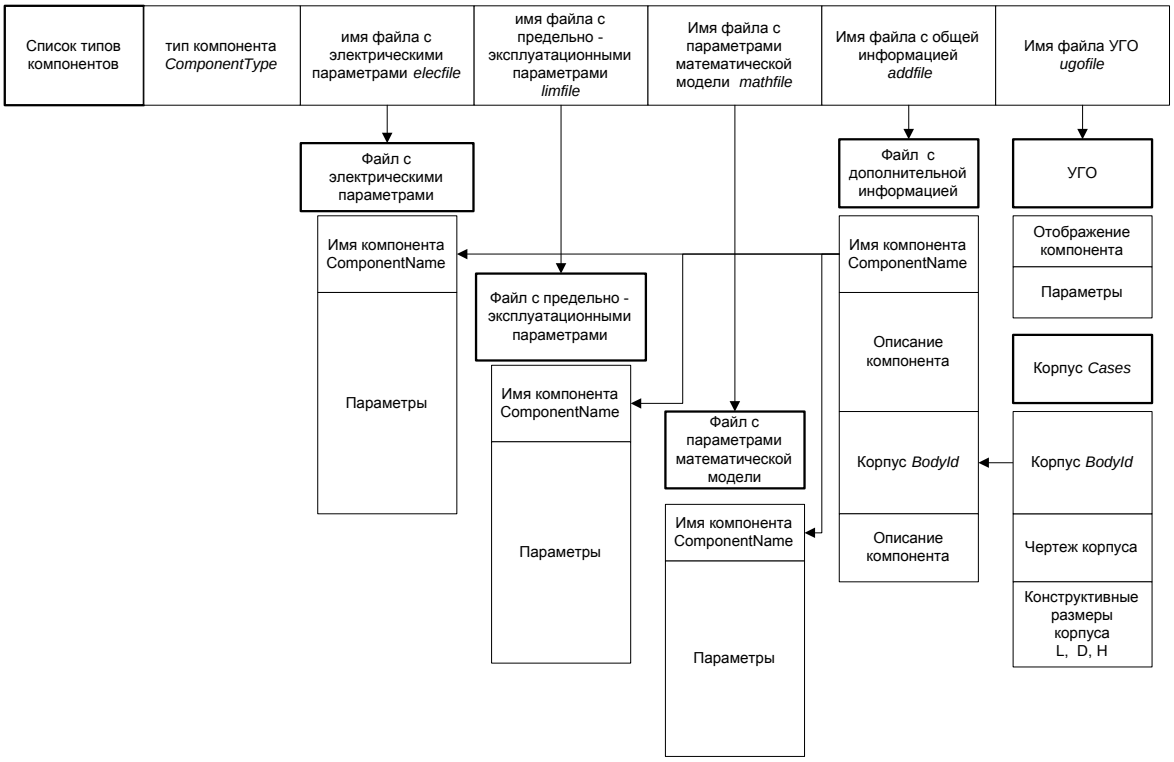


Рис. 1. Структурная схема построения базы данных электронных компонентов

Реляционная модель БД представляет любой объект, как совокупность отношений, которая может быть представлена в виде таблицы, в которой нет полностью совпадающих строк. Каждая строка таблицы соответствует определенному объекту – например, конкретному транзистору. Поскольку строки в таблице не дублируются, любой электронный компонент полностью определяется совокупностью значений своих атрибутов. Реляционная модель БД позволяет применять к таблицам систему операций, позволяющую получать (вычислять подобно арифметическим операциям) одни значения параметров по другим. Это дает возможность делить информацию на хранимую и не хранимую – вычисляемую части. Так можно вычислить требуемые для анализа параметры математической модели транзистора через его электрические параметры. При синтезе можно определить, подходит ли данный транзистор по предельно-эксплуатационным параметрам, оценив электрические параметры по параметрам математической модели.

Таким образом, БД разработанной САПР придерживается следующих условий и ограничений: не может быть одинаковых первичных ключей, т.е. все строки (записи) таблицы должны быть уникальны; все строки таблицы должны иметь одну и ту же структуру, т.е. одно и то же количество атрибутов с соответственно совпадающими именами; имена столбцов таблицы должны быть различны, а значения столбцов должны быть однородными (однотипными); значения атрибутов должны быть атомарными, следовательно, отношения не могут иметь в качестве компонента другие отношения.

Каждая из известных программ проектирования РТУ (например, DesignLab, Accel EDA, OrCAD, Micro-CAP и др.) работает, как правило, только со своей (внутренней) БД и нет практически никакой возможности изменить существующее положение (в рамках конкретной программы [5]). В разработанной же БД программы имеется возможность использования файлов БД, созданных другими разработчиками, за счет применения модуля динамического расширения номенклатуры связанных БД [2].

Для представления знаний о проектировании РТУ в ЭС целесообразно выделить иерархическую структуру РТУ, состоящую из их узлов и электронных компонентов [4], которая характеризуется уровнями: БЗ-I, БЗ-II и т. п. Нумерация уровней может быть произвольной.

Для формализации знаний о проектировании РТУ выделим шесть уровней, см. табл. 1.

Это позволяет сохранить предложенный механизм работы БД электронных компонентов РТУ и при возможном формировании БД РТС, хранящей параметры РТУ синтезированные в ЭС. При этом весь механизм работы БД остается неизменным, что дает возможность использовать его для формирования БД на всех уровнях иерархии РТУ и РТС.

Таблица 1

Уровни иерархии компонентов РТУ и РТС

Уровень	Объект проектирования	Алфавит
БЗ-I	Проектирование базовых электронных компонентов	индуктивности, емкости, резисторы, источники тока, напряжение и т. д.
БЗ-II	Проектирование каскадов РТУ (определение схемы включения УЭ, обратных связей и т.п.)	электронные компоненты, хранящиеся в БД электронных компонентов
БЗ – III	Проектирование узлов РТУ (формирование сигнала (смесь); формирование несущей; усилители + фильтры + АРУ; преобразователи частоты; ограничители; демодуляторы сигнала; НЧ фильтрация и усиление; суммирование сигналов; интегрирование сигналов; перемножение сигналов; задержка сигнала; фазовращение)	хранящиеся в объектно-ориентированных БЗ узлов РТУ
БЗ – IV	Проектирование РТУ из узлов (Радиоустройство: кодирующее; декодирующее; модулирующее и демодулирующее; генератор несущей; антенны (на входе и выходе); процесса распространения; вторичная обработка сигнала.)	хранящиеся в продукционных БЗ РТУ
БЗ – V	Радио канал: поиска сигнала; слежения за параметрами сигнала; измерения параметров сигнала; сопровождения цели; передачи информации; измерения координат цели; радиоразведки параметров сигнала; радио противодействия.	хранящиеся в продукционных БЗ синтеза радиоканалов
БЗ –VI	Радиосистемы: система передачи информации (СПИ); радиолокационная система (РЛС); радионавигационная система (РНС); система радиоуправления (СРУ); система радиоразведки (СРР); система радио противодействия (СРП).	хранящиеся в продукционных БЗ синтеза РТС

В БЗ ИС автоматизации должна храниться вся информация, касающаяся проектируемых устройств. В том числе:

– модели проектируемых объектов, например, модели электронных компонентов РТУ;

– информационные структуры, например, методы схемотехнического синтеза каскадов и функциональных узлов РТУ;

– данные о различных свойствах объектов;

– данные, описывающие текущее состояние процесса проектирования;

– конструкторские документы.

С помощью моделей данных РТУ можно представить как абстрактные информационные объекты, которые состоят из совокупности значений, каждое из которых описывает то или иное свойство моделируемого объекта. Как правило, для упрощения работы с моделями каждому значению, описывающему свойство моделируемого объекта, присваивается имя – атрибут. Таким образом, с тем или иным свойством моделируемого объекта в модели связаны атрибут и его значение и в базе данных хранятся значения, атрибуты, информационные объекты и модели.

Для выполнения работ по машинному проектированию РТУ необходимы инструменты для создания и манипулирования библиотеками математических моделей компонентов [3]. Данные в предметно-ориентированных базах данных хранятся в виде логических групп, чтобы упростить пользователям восприятие информации. Например, может быть создана группа схем усилителей, фильтров и т.п.

Содержимое подобных БЗ представляет собой совокупность информации из различных источников, организация быстрого доступа к ней и устранение ее несогласованности является сложной проблемой, которая может быть решена при использовании двухуровневого построения системы, когда исходные знания, заложенные в программу разработчиками при ее создании и хранящиеся отдельно от новой информации, добавляемой и хранимой в первичных базах знаний, а операции модификации данных могут иметь ограниченную область действия или вообще могут быть запрещены, что может потребоваться, например, при использовании подобных программ в учебных целях, когда с ними работают студенты.

В результате многолетней практики разработки РТУ сложились определенные способы описания структурных, функциональных, принципиальных схем и задач проектирования. Традиционные методы проектирования, в значительной степени использующие интуицию разработчика и не всегда подкрепленные чет-

ким математическим обоснованием, не накладывали особых ограничений на содержание и форму записи исходной информации. Современные методы проектирования с помощью ЭВМ предъявляют более жесткие требования к форме записи информационной базы по составу, объему и степени формализации.

Применение автоматизированных методов проектирования РТУ требует введения в исходную информацию ряда новых, не используемых в традиционном проектировании, параметров, что существенно увеличивает сложность и объем информационных баз знаний и данных, чье формирование осложняется тем, что схемотехническое проектирование РТУ проводится, по меньшей мере, на двух уровнях:

– функциональном (для структурных или функциональных схем);

– схемотехническом (для принципиальных или эквивалентных схем).

Уровень схемотехнического и функционального проектирования неодинаков. Для функционального проектирования преимущественно используют традиционные методы и экспертные системы, поскольку математический аппарат машинного моделирования разработан слабо. Найденные в результате функционального проектирования качественные показатели функциональных узлов РТУ следует использовать как образец, приближение к которому является задачей схемотехнического проектирования.

Следовательно, особенно важно обеспечить единство обобщенного описания функциональных узлов на функциональном и схемотехническом уровне, представленного в четкой математической форме, не допускающей двоякого толкования, а также целостность разрабатываемого проекта и высокую степень управляемости процесса проектирования.

Развитие методов адаптации в САПР РТУ непосредственно связано с процессом интеллектуализации САПР. Потребность в интеллектуализации САПР (особенно при проектировании СБИС) стимулирует исследования по созданию ЭВМ пятого поколения, которые сами являются частью программ по созданию искусственного интеллекта. Реальные перспективы для автоматизации проектирования связаны с ЭС, положившими начало новой области развития техники—инженерии знаний. Такие системы способны накапливать знания в данной области науки и техники и выступать в роли сопроектировщика при формулировке и решении сложных задач.

Литература

1. *Долин Г.А.* Разработка программного обеспечения для формирования распределенной динамической базы данных компонентов радиооборудования и телевидения // Сборник трудов Международной научно-технической конференции «Актуальные проблемы радио- и кинотехнологий», которая пройдёт 26-28 октября 2016 г. в Санкт-Петербургском государственном институте кино и телевидения. Спб.: Санкт-Петербургский государственный институт кино и телевидения, 2016. С. 146-151.
2. *Долин Г.А.* Проблема сквозного автоматического схемотехнического проектирования радиооборудования и пути ее решения // Образовательная среда сегодня и завтра: Сборник научных трудов XI Международной научно-практической конференции (Москва, 28-29 ноября 2016) / под общ. ред. Г.Г. Бубнова, Е.В. Плужника, В.И. Солдаткина. М.: МТИ, 2016. С. 191-193.
3. *Коробова И.Л.* Принятие решений в системах, основанных на знаниях [Электронный ресурс]: учебное пособие / И.Л. Коробова, Г.В. Артемов. Электрон. текстовые данные. Тамбов: Тамбовский государственный технический университет, ЭБС АСВ, 2012. 81 с.
4. *Лоскутов Е.Д.* Схемотехника аналоговых электронных устройств [Электронный ресурс]: учебное пособие / Е.Д. Лоскутов. Электрон. текстовые данные. Саратов: Вузовское образование, 2016. 264 с.
5. Основы компьютерного моделирования [Электронный ресурс]: учебно-методический комплекс. Электрон. текстовые данные. Алматы: Нур-Принт, 2015. 175 с.

ОСОБЕННОСТИ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ БОРТОВЫХ КОМПЛЕКСОВ УПРАВЛЕНИЯ

Борш Анастасия Дмитриевна
студент группы 2БШИ1403 ВШ БИТИС МТУСИ
3997799@mail.ru

Херсонский Антон Владимирович
студент группы БСУ1401 МТУСИ
xxx50099@yandex.ru

Иванова Ольга Валентиновна
МТУСИ, ст. преподаватель кафедры МКиИТ
ivolga07@gmail.com

Рассмотрены проблемы эффективности и качества функционирования бортовых комплексов управления обработкой данных. Сформулированы требования обеспечения надежности электронной компонентной базы для ракетно-космической техники с учетом радиационной стойкости и других факторов воздействия космического пространства. Приведена разработанная схема бизнес-процесса тестирования бортового комплекса управления и сбора научной информации.

Ключевые слова: *магистральный последовательный интерфейс, электронная компонентная база, модуль шифрации/дешифрации, проверка на целостность, блок данных, интерфейс электронных модулей.*

В настоящее время уделяется особое внимание разработке отечественного программного обеспечения для высокотехнологичных областей. Эффективность и качество функционирования бортовых комплексов управления определяется соответствующими характеристиками программного обеспечения, предназначенного для оптимизации всего комплекса задач обработки данных и управления.

Для достижения высоких требований к надежности аппаратуры ракетно-космической техники электронная компонентная база, предназначенная для ее комплектования, должна обладать повышенным уровнем качества и радиационной стойкостью, которая обеспечивает ее надежное функционирование в условиях различных воздействующих факторов космического пространства.

Для каждого космического аппарата разрабатывается совокупность испытаний, определенных комплексной программой экспериментальной разработки. Комплексные испытания проводятся для функционально связанного изделия в условиях, близких к реальным условиям эксплуатации.

Бортовой комплекс управления – это совокупность систем космического аппарата (КА), обеспечивающих управление функционированием КА как единого целого, бортовые системы которого связаны между собой информационными каналами. Одним из примеров применения подобных систем являлся проект “Луна-25” – российская автоматическая межпланетная станция с посадочным аппаратом для высадки лунохода на Южном полюсе Луны.

Бортовой комплекс управления и блок управления и сбора научной информации предназначены для организации работы исследовательской системы на орбитальных спутниках.

Система включает в себя независимые блоки научного оборудования, каждый из которых производит определенный объем работ. Вся полученная в ходе исследований информация собирается в блоке передачи данных, и, в момент выхода спутника на связь, передается принимающей стороне [1].

Главным бизнес-процессом является тестирование работы систем бортового комплекса управления и блока управления и сбора научной информации [2].

В связи с этим необходимо оптимизировать данный этап бизнес-процесса. Это можно осуществить, написанием программы, которая сможет не только прочитать журнал, но и выявить ошибки и составить статистику журнала за считанные секунды. Приложение должно одновременно отображать содержание двух документов – исходного протокола работы и дешифрированный протокол. Также в экранной форме должен отображаться результат работы анализатора в виде краткого обзора документа. Схема бизнес-процесса представлена на рис. 1.

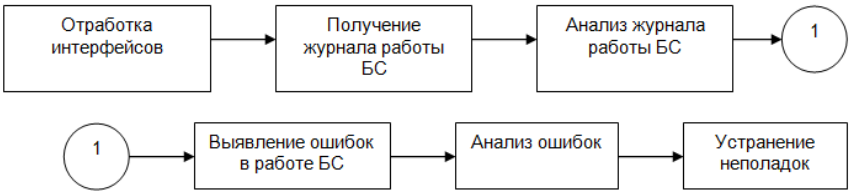


Рис. 1. Схема бизнес-процесса

Программно-аппаратная отработка интерфейсов реализована с помощью специально разработанной контрольно-измерительной аппаратуры (КИА БУНИ) на технологическом образце прибора, изготовленного в виде макета для экспериментальной отработки космического аппарата (натурно-габаритно-массовый макет и тепловой эквивалент). Тестовые проверки подтвердили использование заявленных характеристик прибора для всех типов интерфейсов.

Испытания проводятся для того, чтобы выявить ошибки в работе оборудования, которые впоследствии необходимо исправить, а так же для того, чтобы подготовить аппарат для запуска, который должен исправно работать в условиях, отличных от земных.

В результате исследования было выявлено, что самым продолжительным этапом является - анализ журнала бортовых систем.

Журнал работы системы содержит в себе тысячи строк, их анализ ручным способом очень трудоемок и занимает много времени. Данная программа во многом позволит сократить время разбора информации и повысить точность анализа. Также, на основе полученных данных, имеется возможность сформировать выходной документ с кратким обзором элементов, описывающих основные параметры системы [3, 6]. Это связано с тем, что в процессе испытаний оборудования создаётся большое количество журналов. Каждый журнал содержит в себе около 6000 строк. Чтение этих журналов также осложняется тем, что данные представлены в шестнадцатеричной системе счисления. Данная программа позволяет произвести проверку соответствия входных данных для выявления ошибок и несоответствия стандарту ГОСТ Р 52070-2003.

Настоящий стандарт распространяется на магистральный последовательный интерфейс с централизованным управлением, применяемый в системе электронных модулей и устанавливает требования к: организации обмена информацией; функциям устройств интерфейса и контролю передачи информации [4].

Программа предназначена для разработчиков и тестировщиков проекта “Луна-25”. Во время выхода космического аппарата на связь принимающая станция формирует на основе сигнала передачи файл журнала работы, который необходимо анализировать с помощью программы. Результаты анализа используются для изучения функционирования и обмена информацией бортового комплекса управления (блока управления и сбора научной информации) и отдельных научно-исследовательских модулей [5].

Передача информации организована через обмен сообщениями по магистральному последовательному интерфейсу шины МКО. Управление организуется контроллером шины. Команды от контроллера шины принимаются оконечными устройствами. Ими же, в ответ, посылается системная информация и данные.

Программа должна одновременно отображать содержание двух документов – исходного протокола работы и дешифрированный протокол. Также в экранной форме должен отображаться результат работы анализатора в виде краткого обзора документа.

Функции программы, доступные пользователю:

- загрузка текстового документа формата *.log;
- загрузка дополнительных файлов (правила анализа и таблица соответствия формата *.xml);
- анализ входного документа;
- анализ ошибок и сохранение результатов выполнения программы в текстовый документ.

Программа должна корректно передавать значения между отдельными модулями. Работа с внешними классами должна быть организована через отдельные переменные (свойства). Методы должны быть открыты для доступа извне, если требуется их функционал; остальные оставаться закрытыми. Данные должны сохраняться на жестком диске в выбранной пользователем директории. Ограничение на время выполнения анализа отсутствует, т.к. разные входные данные имеют разный объем.

В модуль дешифрации приходит структура данных, которая содержит в себе полную информацию о комплекте сообщений (запрос и ответ). При анализе такой структуры, формируется структура, которая уже непосредственно выводится в экранную форму. Схема работы программы представлена на рис. 2.

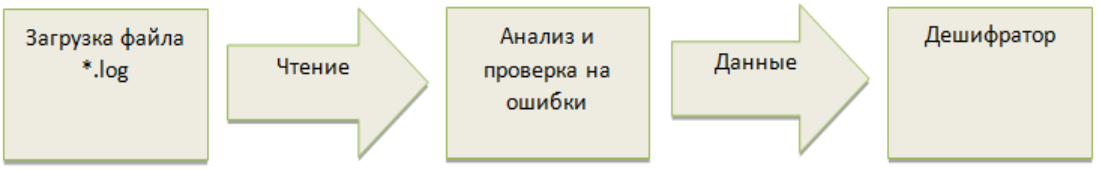


Рис. 2. Схема работы программы

Источником исходных данных является текстовый документ формата *.log, в котором построчно описаны сообщения, состоящие из блоков:

– времени;
– параметров;
– данных.

Сообщения бывают двух типов: запрос и ответ. Запрос формируется контроллером шины, ответ – оконечным устройством.

Все строки разбиваются на блоки по одному-два сообщения, в зависимости от параметров. Проверка на соответствие требованиям ГОСТ Р 52070-2003 производится в каждом блоке по отдельности. Блок сообщений вместе с отчетом об ошибках передается в дешифратор.

Строки проверяются на целостность. Они должны иметь четкую структуру и синтаксис. Требуется проверить сообщение на предмет искажений или потери части данных, т.к. информация с космического аппарата передается по радиоканалу.

Разница значений временных показателей между блоками времени в сообщениях строго регламентирована ГОСТ Р 52070-2003. Отклонения помечаются соответствующим сообщением об ошибке.

Блок параметров состоит из нескольких элементов, каждый из которых проверяется на корректность значения. ГОСТ Р 52070-2003 допускает различие в значениях отдельных параметров между сообщениями.

Блок данных состоит из командного слова/ответного слова (в зависимости от типа сообщения) и слов данных. Значения передаются в шестнадцатеричной системе счисления. При анализе, согласно ГОСТ Р 52070-2003, производится побитовое чтение слов с помощью масок. Данные передаются без изменений, т.к. имеют более сложную схему шифрования. Все встречаемые при анализе ошибки передаются дешифратору отдельно от блока сообщений.

Входная информация проверяется на корректность. Присутствует проверка на формат загружаемых файлов. В функциях программы включена обработка исключений, чтобы избежать аварийного завершения программы. Программа должна корректно работать как при наличии ошибок в работе бортовых комплексов, так и при ошибках в составлении файла исходного протокола работы.

Дешифрированный протокол должен содержать следующую информацию: время обращения; номер полукомплекта бортового комплекса; характер обращения (запрос или ответ); шина, по которой произведено сообщение (основная или резервная).

Данные, обрабатываемые при запросе:

– адрес оконечного устройства (ОУ);
– подадрес оконечного устройства или режим управления;
– число передаваемых слов данных (СД) или код команды;

Данные, обрабатываемые при ответе:

– адрес оконечного устройства (ОУ);
– признаки ответного слова (при их наличии);
– слова данных при обращении (при их наличии);
– ошибки (при их наличии).

Помимо выходного документа на форму необходимо вывести краткий обзор элементов, описывающих основные параметры системы (количество обращений, ошибок и т.д.). При выявлении ошибочного обращения необходимо описать характер ошибки и время данного обращения, чтобы облегчить навигацию по дешифрированному протоколу.

Взаимодействие с программой организовано с помощью оконного пользовательского интерфейса, т.к. он объединяет в себе все элементы и компоненты программы: навигация между блоками системы, визуальный дизайн и порядок использования программы, обратная связь с пользователем.

Процесс импортозамещения в России в космической отрасли реализуется по плану уже много лет, а не так давно введенные санкции еще сильнее его ускорили. В связи с этим необходимо осваивать новые тех-

нологии, которые в прошлом закупались у иностранных государств. Программа «Луна-25» является в России первой в лунной программе. Она является первым российским лунным заданием. «Луна-25» - это российская автоматическая межпланетная станция с посадочным аппаратом, который доставит в район Южного полюса Луны луноход. Запуск планируется осуществить в ноябре-декабре 2019 года с космодрома Восточный. В связи с этим, под каждое оборудование разрабатывается свое программное обеспечение, которое не имеет в своем роде аналогов. Все программы уникальны и разрабатываются с нуля. Данный программный продукт не является исключением. Отсутствие аналогов имеет свое преимущество, так как программиста не будут сдерживать рамки определенных функций, реализованных в аналогичных программах. Практическая значимость заключается в том, что разрабатываемое программное обеспечение может быть применено на любой реализации бортового комплекса управления и блока управления и сбора научной информации в бортовой авионике, работа которых удовлетворяет требования ГОСТ Р 52070-2003.

Литература

1. *Микрин Е.А.* Бортовые комплексы управления космических аппаратов. М.: Издательство МГТУ им. Н.Э. Баумана, 2014. 245 с.
2. *Виноградова Е., Лобанова А., Долганова О*. Моделирование бизнес-процессов. М.: Юрайт, 2017. 290 с.
3. *Иванова О.В., Иванов П.В*. Проектирование экспертных систем для диагностики неисправностей // T-Comm: Телекоммуникации и транспорт. 2013. №10. С. 51-52.
4. ГОСТ Р 52070-2003 Интерфейс магистральный последовательный системы электронных модулей, Госстандарт России, 2003. 24 с.
5. *Иванова О.В., Иванов П.В*., *Тишкин С.В*. Интегрированные решения в области компьютерного обеспечения для корпоративных пользователей // T-Comm: Телекоммуникации и транспорт. 2009. №S2. С. 154-155.
6. *Иванова О.В., Иванов П.В., Борисов К.А.* Использование интеллектуальных технологий обработки неструктурированных данных в НСИ // T-Comm: Телекоммуникации и транспорт. 2012. Т. 6. № 10. С. 56-57.

АКТУАЛЬНЫЕ ПРОБЛЕМЫ СОВЕРШЕНСТВОВАНИЯ СИСТЕМЫ ЭКОНОМИЧЕСКОЙ БЕЗОПАСНОСТИ РЕГИОНА С ИСПОЛЬЗОВАНИЕМ IT

Аношкина Е.С.,
Студентка группы ЗМФУ1601 МТУСИ
e.anoshkina@bk.ru

Система экономической безопасности региона невозможна без информационно-аналитической поддержки принятия решений на федеральном, региональном и муниципальном уровнях управления на основе комплексного мониторинга и оценки состояния объектов управления; анализа возникающих проблем, их причин и последствий; а также прогнозирования развития ситуации и информационной поддержки принятия решений. Поэтому основным направлением ее совершенствования является создание ситуационных центров глав субъектов Российской Федерации с использованием современных инфокоммуникационных технологий. Ситуационный центр представляет собой организационно-технический комплекс, включающий в себя специализированное помещение, оснащенное комплексом телекоммуникационного оборудования (видео-конференц-связь, конференц-связь, другие средства интерактивного представления информации) и специальным программным обеспечением.

Ключевые слова*: экономическая безопасность региона, информационные технологии, ситуационные центры, информационно-аналитические системы.*

Понятие «безопасность» многогранно, в связи с множеством источников возникновения угроз ее нарушения. В соответствии со Стратегией национальной безопасности, утвержденной Указом Президента Российской Федерации от 31 декабря 2015 г. № 683, обеспечение национальной безопасности включает в себя обеспечение военной, государственной, общественной, экологической безопасности; повышение качества жизни граждан; экономический рост и т.д. [1, с.3-5]. Таким образом, одним из ключевых факторов, влияющих на уровень национальной безопасности страны, является состояние социально-экономического потенциала страны. Развитие нормативно-правовой базы обеспечения национальной безопасности связано с утверждением Указом Президента Российской Федерации от 13 мая 2017 г. № 208 Стратегии экономической безопасности Российской Федерации на период до 2030 года. Стратегия экономической безопасности определила цели, основные направления и задачи государственной политики по реализации стратегических национальных приоритетов Российской Федерации в сфере обеспечения экономической безопасности. Основными направлениями государственной политики в сфере обеспечения экономической безопасности являются:

1) развитие системы государственного управления, прогнозирования и стратегического планирования в сфере экономики;
2) обеспечение устойчивого роста реального сектора экономики;
3) создание экономических условий для разработки и внедрения современных технологий, стимулирования инновационного развития, а также совершенствование нормативно-правовой базы в этой сфере;
4) устойчивое развитие национальной финансовой системы;
5) сбалансированное пространственное и региональное развитие Российской Федерации, укрепление единства ее экономического пространства;
6) повышение эффективности внешнеэкономического сотрудничества и реализация конкурентных преимуществ экспортно-ориентированных секторов экономики;
7) обеспечение безопасности экономической деятельности;
8) развитие человеческого потенциала [2, с.5].

Стратегия национальной безопасности страны предусматривает создание системы распределенных ситуационных центров, как важнейшего элемента системы стратегического планирования. Правовая основа создания ситуационных центров включает в себя ряд нормативно-правовых актов и рекомендательных документов, разработанных под влиянием необходимости информатизации процессов государственного управления. К ним относятся: Концепция создания системы распределенных ситуационных центров; Методические рекомендации по созданию и вводу в эксплуатацию ситуационных центров глав субъектов Российской Федерации; Указ Президента РФ от 25 июля 2013 г. № 648 «О формировании системы распределенных ситуационных центров, работающих по единому регламенту взаимодействия». В круг задач, решаемых ситуационными центрами, входят:

– проведение стратегического анализа социально-экономического развития Российской Федерации;

– мониторинг уровня развития Российской Федерации, необходимый для подготовки документов стратегического планирования на основе единых исходных данных;

– программно-целевое проектирование и программирование процессов устойчивого развития Российской Федерации и обеспечения национальной безопасности [3, с.12].

Таким образом, развитие ситуационных центров и их взаимодействие по единому регламенту позволит повысить эффективность информационной поддержки реализации государственной политики в сфере социально-экономического развития России и обеспечения национальной безопасности [3, с.13].

Ситуационные центры создаются в целях информационно-аналитической поддержки принятия решений на федеральном, региональном и муниципальном уровнях управления на основе комплексного мониторинга и оценки состояния объектов управления; анализа возникающих проблем, их причин и последствий; а также прогнозирования развития ситуации и информационной поддержки принятия решений.

Ситуационный центр представляет собой организационно-технический комплекс, включающий в себя специализированное помещение, оснащенное комплексом телекоммуникационного оборудования (видеоконференц-связь, конференц-связь, другие средства интерактивного представления информации) и специальным программным обеспечением. Совокупность специального программного обеспечения интегрируется в единую информационно-аналитическую систему, обеспечивающую всестороннюю поддержку функционирования Ситуационного центра. Информационно-аналитическая система ситуационных центров органов государственной власти - это программный комплекс с множеством подсистем, имеющих целевое назначение, предназначенных для мониторинга, анализа, прогнозирования показателей социально-экономического развития объекта управления, аналитической поддержки принятия решений на основе накапливаемой информационной базы. Информационно-аналитическая система позволяет повысить эффективность управления на основе использования IT-технологий. Среди задач, решение которых осуществляется посредством функционирования информационно-аналитической системы, следует выделить следующие:

– ведение единого информационного ресурса и обеспечение регулярного пополнения базы данных, посредством информационного взаимодействия со смежными информационно-аналитическими системами и ситуационными центрами;

– мониторинг индикаторов, характеризующих социально-экономическое, общественно-политическое состояние объектов управления, а так же состояние системы комплексной безопасности;

– выявление существующих проблем, анализ рисков и угроз; разработка на основе собранной информации сценарного развития ситуаций; выработка рекомендаций в части принятия управленческих решений на основе проведенного анализа;

– контроль исполнения указаний вышестоящих органов власти;

– обеспечение проведения дистанционных совещаний руководителей органов власти с использованием средств визуализации информации [3, с.20].

В соответствии с Единым регламентом взаимодействия распределенных ситуационных центров, утвержденным решением Межведомственной комиссии по координации деятельности федеральных органов исполнительной власти по созданию системы распределенных ситуационных центров, работающих по единому регламенту взаимодействия (Протокол № 2 от 7 мая 2015 г.), единая система распределенных ситуационных центров Российской Федерации включает в себя ситуационные центры федерального и регионального уровней. К федеральному уровню единой системы распределенных ситуационных центров Российской Федерации относятся ситуационные центры:

– Президента Российской Федерации;

– Правительства Российской Федерации;

– Контрольного управления Президента Российской Федерации;

– Совета Безопасности Российской Федерации;

– Федеральной службы охраны (ФСО);

– иных федеральных органов исполнительной власти (министерств и ведомств);

– полномочных представителей Президента Российской Федерации в федеральных округах [4, с.3].

Ситуационные центры федерального уровня взаимосвязаны между собой защищенными телекоммуникационными сетями и системами видео-конференц-связи, что позволяет руководителям в режиме реального времени принимать обоснованные и адекватные управленческие решения по вопросам социально-экономического развития страны.

В связи с тем, что социально-экономическое развитие страны напрямую зависит от социально-экономического состояния регионов, необходимо создание ситуационных центров на региональном уровне и их подключение к единой системе распределенных ситуационных центров в Российской Федера-

ции. Создание ситуационных центров глав субъектов Российской Федерации является основным инструментом повышения уровня экономической безопасности региона. Основными направлениями в структуре экономической безопасности региона являются:

1. Повышение конкурентоспособности региона, его экономической независимости от федерального центра и взаимосвязанности экономик субъектов Российской Федерации.
2. Обеспечение стабильности и устойчивости развития региональной экономики, предполагающее создание благоприятных условий для предпринимательской деятельности; развитие социальной экономики, ориентированной на повышение уровня благосостояния людей и социальной защиты.
3. Обеспечение условий для развития региона посредством создания благоприятных условий для инвестиционных вложений и инновационной деятельности; повышение профессионального, образовательного и культурного уровня образования населения [2, с.61].

Среди проблем, возникающих перед регионами при обеспечении экономической безопасности, можно выделить следующие:

1. дисбаланс в обеспеченности регионов ресурсами, в том числе финансовыми, природными и трудовыми;
2. дисбаланс в технологическом и инновационном потенциале регионов;
3. несогласованность в разделении полномочий в экономической сфере между федеральным центром и регионами;
4. длительность выполнения поручений вышестоящих органов управления в связи с усложненным процессом контроля исполнения поручений;
5. неэффективное прогнозирование социально-экономического развития региона [5].

Вышеуказанные проблемы могут привести к снижению уровня экономической безопасности региона. Интегральное рассмотрение государственной политики во всех сферах хозяйственной деятельности региона позволяет выделить специфические, присущие только данному региону задачи, эффективность решения которых напрямую связана с обеспечением экономической безопасности региона.

Важную роль в обеспечении экономической безопасности региона играют инфокоммуникации, обеспечивающие доступ потребителей к различным информационным и телекоммуникационным услугам, создающие условия для успешного функционирования экономических отраслей и сфер деятельности, способствующие появлению и развитию новых форм ведения бизнеса [7, 9, 10-12]. Внедрение новых инфокоммуникационных технологий и услуг, развитие фиксированной и подвижной связи способствует экономическому росту и развитию региона [6, 8].

На региональном уровне управления детальное рассмотрение задач, направленных на реализацию стратегических национальных приоритетов в целях обеспечения экономической безопасности, неизбежно приводит к необходимости корреляции социально-экономического, общественного политического развития и сферы обеспечения безопасности во всех документах стратегического развития региона. Прежде всего, это относится к стратегии социально-экономического развития субъекта на очередной плановый период, стратегии развития отдельных отраслей промышленности региона, стратегии территориального развития субъекта, стратегии обеспечения комплексной безопасности и обеспечения жизнедеятельности населения и другим планирующим документам. Интегрировать эти документы, обеспечить их взаимное согласование по целям, задачам, срокам, реализуемым мероприятиям и ресурсам, обеспечить постоянный и эффективный контроль реализации документов стратегического развития, а при необходимости и вносить коррективы в их осуществление, возможно только на основе совершенствования системы государственного управления регионом. С этой целью в Российской Федерации создаются ситуационные центры глав субъектов Российской Федерации, являющиеся, как уже было сказано ранее, составляющим звеном единой системы распределенных ситуационных центров.

Информационно-аналитические системы, развернутые на базе ситуационных центров глав субъектов Российской Федерации будут обеспечивать решение задач сбора, хранения, обработки, мониторинга, контроля и анализа данных, а также прогнозирования и моделирования социально-экономического, общественно-политического развития региона и обеспечения комплексной безопасности, на базе подсистем подготовки (поддержки) принятия решений.

Разработка соответствующих подсистем мониторинга и контроля, анализа и прогнозирования (моделирования), стратегического и территориального планирования, проектного управления, комплексной системы безопасности на мирное и военное время на базе IT-технологий позволит обеспечить согласованное и поступательное социально-экономическое и общественно – политическое развитие региона, обеспечение комплексной, в том числе и экономической безопасности субъекта Российской Федерации.

Информационно-аналитическая система региона – это интеллектуальный инструмент, обеспечивающий решение задач сбора, хранения, обработки, мониторинга, контроля и анализа данных, а также прогнозиро-

вания и моделирования социально-экономического, общественно – политического развития области и обеспечения комплексной безопасности, поддержки принятия решений. По словам Заместителя начальника Управления Спецсвязи ФСО России Н.И.Ильина: «система распределенных ситуационных центров – это инновационный инструмент для поддержки государственного управления. Стратегическая задача ИТ-специалистов в содружестве с управленцами состоит в его эффективном использовании».

Литература

1. Указ Президента РФ от 31.12.2015 № 683 «О Стратегии национальной безопасности Российской Федерации». // СПС КонсультантПлюс // Опубликован 31.12.2015 на официальном интернет-портале правовой информации http://www.consultant.ru.
2. Указ Президента РФ от 13 мая 2017 г. № 208 «О Стратегии экономической безопасности Российской Федерации на период до 2030 года». // СПС КонсультантПлюс // Опубликован 13.05.2017 на официальном интернет-портале правовой информации http://www.consultant.ru.
3. Ивашкевич В.Ситуационные центры: применение в государственном управлении на региональном и федеральном уровнях.– URL: http://www.prognoz.ru/sites/default/fi les/ vera_ivashkevich_situacionnye_centry_1.pdf (дата обращения 17.12.2017).
4. Единый регламент взаимодействия распределенных ситуационных центров, утвержденным решением Межведомственной комиссии по координации деятельности федеральных органов исполнительной власти по созданию системы распределенных ситуационных центров, работающих по единому регламенту взаимодействия (Протокол № 2 от 7 мая 2015 г.).
5. *Лаврут Н.С.* Экономическая безопасность регионов как основа безопасности страны // Экономика и современный менеджмент: теория и практика: сб. ст. по матер. XXII междунар. науч.-практ. конф.-Новосибирск: СибАК, 2013.
6. *Кухаренко Е.Г., Гасс Я.М., Серебряков Ю.Ю.* Механизм оценки перспектив развития операторов MVNO в регионах России // Электросвязь. 2015. №9. С. 44-46.
7. *Андреева О.Д., Абрамова А.В., Кухаренко Е.Г.* Развитие использования цифрового маркетинга в мировой экономике // Российский внешнеэкономический вестник. 2015. Т.2015. №4. С.24-41.
8. *Гасс Я.М., Кухаренко Е.Г.* Современный этап развития MVNO в России и в мире спутниковые системы связи и вещания // Труды научно-исследовательского института радио. 2015. №3. С. 26-32.
9. *Резникова Н.П., Кухаренко Е.Г.* Маркетинг в отрасли инфокоммуникаций. Учебное пособие для вузов. М.: Горячая линия-Телеком, 2013. 152 с.
10. *Кухаренко Е.Г.* Лояльность клиентов в инфокоммуникациях: значение и оценка // T-Comm: Телекоммуникации и транспорт. 2012. Т. 6. № 12. С. 62-63.
11. *Кухаренко Е.Г.* Исследование эволюции маркетинговых концепций в инфокоммуникационном бизнесе // T-Comm: Телекоммуникации и транспорт. 2015. Т. 9. № 9. С. 72-75.
12. *Кухаренко Е.Г., Салютина М.Е.* Применение методов стратегического анализа для оценки конкурентоспособности телекоммуникационных компаний // T-Comm: Телекоммуникации и транспорт. 2012. Т. 6. № 12. С. 64-65.

СРАВНИТЕЛЬНЫЙ АНАЛИЗ МОДЕЛЕЙ ВЗАИМОДЕЙСТВИЯ КОМПАНИЙ НА РЫНКЕ УСЛУГ ПОДВИЖНОЙ СВЯЗИ СОЦИАЛИСТИЧЕСКОЙ РЕСПУБЛИКИ ВЬЕТНАМ

Мак Ван Кыонг
студент группы МЭЭ1601 МТУСИ
maksimmoscow1984@gmail.com

Актуальной проблемой для операторов подвижной связи социалистической республики Вьетнам в условиях насыщения рынка традиционными голосовыми услугами является расширение предоставления услуги с добавленной стоимостью (VAS-услуги) как источника дополнительных доходов. По мере развития вьетнамского рынка VAS-услуг его структура усложнилась, появились новые типы компаний, поэтому важной задачей стало внедрение эффективных моделей взаимодействия между ними. Проведен анализ основных моделей взаимодействия операторов подвижной связи с контент-провайдерами, показаны преимущества агрегационной модели.

Ключевые слова: *услуги подвижной связи, мобильный контент, VAS-услуги, агрегационная модель взаимодействия операторов.*

Сотовая подвижная связь является крупнейшим сегментом инфокоммуникационного рынка социалистической республики Вьетнам (СРВ). Сегодня в стране функционируют 6 операторов мобильной связи: VinaPhone, MobiFone, Sfone, Viettel, GMobile, Vietnamobile; три оператора являются крупными государственными компаниями.

Развитие подвижной связи способствует росту национальной экономики, расширению форм и методов ведения бизнеса в различных отраслях, повышению качества жизни населения [1].

Не смотря на специфику рынка услуг подвижной связи СРВ, здесь проявляются общемировые тенденции развития подвижной связи. В настоящее время рынок находится в стадии зрелости, характеризующейся стабилизацией прибыли, снижением показателя ARPU вследствие снижения тарифов и появление тенденции к снижению рентабельности услуг. Поэтому для обеспечения конкурентоспособности операторы стремятся развивать услуги с добавленной стоимостью (VAS-услуги) как источник дополнительных доходов [3, 8, 13].

Мобильный контент – это текстовая, графическая, голосовая, звуковая или мультимедийная информация, которая сформатирована для передачи через сеть мобильной связи. Контент-услуга – это готовый продукт информационно-развлекательного содержания или совокупность действий по разработке, форматированию и предоставлению, а также технической поддержке контента [9]. Контентные услуги составляют основу вьетнамского рынка VAS-услуг.

На этапе становления рынка VAS-услуг операторы подвижной связи самостоятельно занимались разработкой дополнительных сервисов, в основном голосовых справочно-информационных услуг. В процессе реализации контент-услуг участвовали всего три стороны: операторы, транспортирующие услуги к абонентам и осуществляющие расчеты за эти услуги; контент-провайдеры, обеспечивающие поставку контента и владеющие правами на него, и потребители услуг. По мере развития рынка VAS-услуг его структура усложняется, появляются новые типы компаний, специализирующиеся на выполнении отдельных функций [11, 12].

Во Вьетнаме рынок услуг мобильного контента начал активно развиваться после 2005 года, появились многочисленные контент-провайдеры и сервис-провайдеры. Позднее появились виртуальные операторы или MVNO (Mobile Virtual Network Operator), то есть компании, продающие услуги под собственной торговой маркой, но при этом не имеющие сетевой инфраструктуры и лицензии на диапазон частот [6, 7]. Наряду с появлением новых типов компаний появились новые модели ведения бизнеса, новые организационные структуры компаний. Например, модели функционирования виртуальных операторов отличаются набором возлагаемых на компанию функций [2, 4, 5]. Проведенный анализ функционирования виртуальных операторов социалистической республики Вьетнам позволил выявить отсутствие бизнес-моделей, подразумевающие аренду или покупку элементов сети оператора подвижной связи.

Увеличение количества участников рынка VAS-услуг актуализировало проблемы поиска эффективных моделей взаимодействия между ними. Основные схемы взаимодействия операторов СРВ следующие:

1. Оператор и контент-провайдер относительно независимы; абонент оплачивает отдельно услуги оператора и контент-провайдера.

2. Оператор покупает контент и предоставляет его абонентам от своего имени, а абоненты расплачиваются за услуги непосредственно с оператором.

3. Контент-провайдер оплачивает услуги оператора, который обеспечивает абонентам доступ к контенту, и получает с абонентов плату за пользование услугами.

Эти модели являются типовыми и реализуются через следующие схемы взаимодействия операторов с другими участниками рынка контентных услуг. Схема взаимодействия (Walled garden) ограничивает доступ потребителя к сетевым ресурсам. Открытая схема взаимодействия (Open garden) предполагает передачу функций продвижения контент-услуг провайдерам, а оператор предоставляет бесплатную возможность обращения к этим услугам. Третий вариант – полуограничивающая схема взаимодействия (Semi-walled garden) [9].

Первоначально операторы подвижной связи использовали схему открытого взаимодействия, однако постепенно начали проявляться негативные тенденции, такие как невозможность контроля качества контента со стороны оператора, недобросовестная реклама или откровенное мошенничество со стороны провайдеров, когда со счета потребителя списывались деньги за непредоставленные услуги. Кроме этого, операторы столкнулись с нехваткой ресурсов для сопровождения всех контент-провайдеров, тем более что рост количества партнёров не приводил к аналогичному росту прибыли. Таким образом, стало очевидно, что работать с большим числом контент-провайдеров нецелесообразно, Поэтому сейчас вьетнамские операторы подвижной связи в основном работают рынке контента, используя агрегационную модель взаимодействия с контент-провайдерами, предполагающую активное участие в продвижении контентных услуг, допуская к агрегации контента на порталах только отдельные проверенные компании. Агрегатор – это компания-посредник между многочисленными контент-провайдерами и оператором, агрегатор непосредственно взаимодействует с оператором и организует процесс доставки контента его абонентам. Получив часть дохода от реализации контента от оператора, агрегатор расплачивается с многочисленными провайдерами.

На вьетнамском рынке определились две модели агрегирования. Первая группа агрегаторов выполняет те функции, которые схема агрегации подразумевала изначально: выполнение посреднических функций между операторами и более мелкими компаниями, желающими работать на рынке мобильного контента. Для многих компаний эта схема стала действительно удобной, они получили единый доступ к пулам коротких номеров всех операторов, их работу сопровождает один менеджер.

Вторая группа агрегаторов взяла на себя обслуживание эксклюзивных проектов операторов, в первую очередь порталов. С точки зрения контент-провайдеров, это положительная тенденция, которая ведет к четкому разделению функций, когда агрегатор получает определенную свободу действий, например, в оценке сервисов или выборе партнера.

Агрегаторам, аккумулирующим широкий спектр сервисов, проще договариваться с операторами мобильной связи и более активно пользоваться услугами рекламных площадок, а значит получать лучший отклик от абонентов. К недостаткам агрегационной модели взаимодействия можно отнести потерю оператором контроля над изменениями на рынке, так как появляются агрегаторы, которые самостоятельно принимают решения о запуске новых сервисов, что уменьшает участие оператора в этом сегменте бизнеса; доступ поставщиков контента к элементам инфраструктуры оператора; периодические задержки платежей рядовым участникам рынка, которые далеко не всегда являются следствием каких-то серьезных проблем той или иной крупной компании-агрегатора или ее необязательностью в отношениях с бизнес-партнерами.

Проведенное исследование показывает, что услуги агрегирования в последнее время становятся особенно актуальными. Однако пока рано говорить о том, что сегмент таких услуг уже сформирован. Лидерство здесь могут завоевать компании, которые безупречно подготовлены технически и обладают хорошо поставленной системой менеджмента. В целом развитие агрегаторства может положительно отразиться на расширении вьетнамского рынка VAS-услуг. Сотрудничество с агрегаторами позволит операторским компаниям быстрее и эффективнее реализовывать их сервисы.

Опыт европейских и японских операторов показывает, что современная, «расширенная» модель агрегации, где агрегатор не только суммирует трафик, но и отвечает за маркетинг услуг и работу по всему спектру сервисов, реализуемых на сети оператора, сделает деятельность этих компаний более эффективной, как с точки зрения предоставления качественного, востребованного рынком продукта, так и точки зрения достижения конечного результата [10]. Однако переход к такой модели агрегаторского бизнеса невозможен без разработки новой модели взаимодействия, которая оптимизирует бизнес-процессы как для оператора, так и для контент-провайдера (агрегатора). Очевидно также, что взаимодействие компаний в рамках бизнес-модели, как совокупности отдельных функциональных подразделений, недостаточно продуктивна, и что система, состоящая из группы скоординированных процессов, должна более эффективно

обеспечивать достижение стратегических целей компаний. Проведённый анализ показывает, что используемые схемы взаимодействия не позволяют операторам в полной мере сконцентрироваться на оказании услуг и управлении взаимоотношениями с абонентами, получая свою долю прибыли от услуг с добавленной ценностью. Бизнес-процессы по предоставлению контентных услуг остаются неоптимизированными, а финансовые потери участников взаимодействия – ощутимо серьезными.

Литература

1. *Андреева О.Д., Абрамова А.В., Кухаренко Е.Г.* Развитие использования цифрового маркетинга в мировой экономике // Российский внешнеэкономический вестник. 2015. Т.2015. №4. С. 24-41.
2. *Гасс Я.М., Кухаренко Е.Г.* Современный этап развития MVNO в России и в мире спутниковые системы связи и вещания // Труды научно-исследовательского института радио. 2015. №3. С. 26-32.
3. *Кухаренко Е.Г.* Жизненный цикл инфокоммуникационных услуг: особенности и тенденции // Экономика и качество систем связи. 2017. №3. С. 33-38.
4. *Кухаренко Е.Г., Гасс Я.М., Серебряков Ю.Ю.* Механизм оценки перспектив развития операторов MVNO в регионах России // Электросвязь. 2015. №9. С. 44-46.
5. *Кухаренко Е.Г., Бецков Г.А.* Исследование бизнес-стратегий мобильных операторов наложенных сетей в России / В сборнике: Труды Московского технического университета связи и информатики. М.: "ИД Медиа Паблишер", 2008. Т. 2. С. 231-239.
6. *Кухаренко Е.Г., Бецков Г.А.* Исследование факторов, влияющих на деятельность мобильных операторов наложенных сетей в России // T-Comm: Телекоммуникации и транспорт. 2009. №S3. С. 21-22.
7. *Кухаренко Е.Г., Бецков Г.А.* Проблемы и перспективы развития виртуальных операторов сотовой подвижной связи в России и в мире//Труды Московского технического университета связи и информатики. М.: "ИД Медиа Паблишер", 2007. С. 302-306.
8. *Кухаренко Е.Г., Гасс Я.М.* Совместное использование инфраструктуры электросвязи и радиочастотного ресурса как механизм управления инвестициями при создании MVNO / В сборнике: Технологии информационного общества. X международная отраслевая научно-техническая конференция: сборник трудов. 2016. С. 316-317.
9. *Кухаренко Е.Г., Гервер В.А.* Разработка модели кросс-функционального взаимодействия операторов на рынке услуг мобильного контента / В сборнике: Труды Московского технического университета связи и информатики. М.: «ИД Медиа Паблишер», 2008. Т.2. С. 240-243.
10. *Кухаренко Е.Г., Токмачев С.С.* Сравнительный анализ методических подходов к управлению проектами и их применение в инфокоммуникациях // T-Comm: Телекоммуникации и транспорт. 2014. Т.8. №7. С. 57-59.
11. *Кухаренко Е.Г., Гасс Я.М.* Преимущества инвестирования развития сетей сотовой подвижной связи при совместном использовании ресурсов/ В книге: Мобильный бизнес: Перспективы развития и реализации систем радиосвязи в России и за рубежом. Сборник материалов (тезисов) XXXVII Международной конференции РАЕН. Москва. 2016. С. 15-16.
12. *Кухаренко Е.Г., Иванченко П.А.* Развитие методов управления производственной деятельностью компании на рынке услуг подвижной связи на основе управления жизненным циклом новых услуг. М.: Компания Спутник +, 2005. 52 с.
13. *Кухаренко Е.Г., Салютина М.Е.* Применение методов стратегического анализа для оценки конкурентоспособности телекоммуникационных компаний // T-Comm: Телекоммуникации и транспорт. 2012. Т.6. №12. С. 64-65.