

НАУЧНЫЙ ЖУРНАЛ

**ТЕЛЕКОММУНИКАЦИИ
И ИНФОРМАЦИОННЫЕ
ТЕХНОЛОГИИ**

№2-2019

(Дата издания: 26.11.2019)

РЕДАКЦИОННАЯ КОЛЛЕГИЯ:

Орлов Владимир Георгиевич *(Главный редактор)*

к.т.н., начальник отдела организации научно-исследовательской работы студентов Московского технического университета связи и информатики (МТУСИ), начальник Центра научной работы и технического творчества молодежи МТУСИ, Москва, Россия

Андреев Владимир Александрович

д.т.н., профессор, Поволжский государственный университет телекоммуникаций и информатики, Самара, Россия

Бачевский Сергей Викторович

д.т.н., профессор, ректор Санкт-Петербургского государственного университета телекоммуникаций им. проф. Бонч-Бруевича, Санкт-Петербург, Россия

Зимин Игорь Викторович

Кыргызский государственный технический университет имени И.Раззакова.
Институт электроники и телекоммуникаций, Бишкек, Кыргызстан

Ланчиков Павел Николаевич

НП Учебный центр Huawei (Москва), Шеньчжень, Китай

Маркосян Мгер Вардкесович

к.т.н., доцент, Ереванский НИИ средств связи, Ереван, Армения

Прохода Александр Николаевич

к.воен.н., доцент, Балтийский военно-морской институт им. Ф.Ф. Ушакова, Калининград, Россия

Рябко Борис Яковлевич

д.т.н., профессор, ректор Сибирского государственного университета телекоммуникаций и информатики, Новосибирск, Россия

Самойлов Александр Георгиевич

д.т.н., профессор, заместитель директора института информационных технологий и радиоэлектроники Владимирского государственного университета имени Александра Григорьевича и Николая Григорьевича Столетовых (ВлГУ), Владимир, Россия

Рогачев Александр Александрович

д.т.н., в.н.с., Гомельский государственный университет имени Франциска Скорины, Гомель, Республика Беларусь

Суржиков Анатолий Петрович

д.ф.-м.н., профессор, Национальный исследовательский Томский политехнический университет, Томск, Россия

Титов Евгений Вадимович

к.т.н., профессор, Московский технический университет связи и информатики, Москва, Россия

УЧРЕДИТЕЛЬ:

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

РЕДАКЦИОННАЯ ПОДГОТОВКА:

**Отдел организации научно-исследовательской работы студентов
(ОНИРС МТУСИ)**

Содержание №2-2019

«Цифровые технологии радиосвязи и телерадиовещания»

Алешин В.С.

Пять мифов о спутниковой связи 5

Разин К.О., Иванюшкин Р.Ю.

Энергетический выигрыш от введения автоматической регулировки режима по питанию в линейный усилитель мощности по схеме У Догерти 12

Санников В.Г., Волчков В.П.

Повышение помехоустойчивости модема с оптимальными финитными сигналами, не вызывающими межсимвольную интерференцию в линейном канале связи 19

Шубина М.В., Косичкина Т.П.

Исследование вопросов точности формирования последовательности Задова-Чу 29

Гут Т.М., Горгадзе С.Ф.

Характеристики ковариационных функций и оценка параметров шумоподобного сигнала..... 35

Мирошниченко А.В., Волчков В.П.

Экстраполяция нестационарных зашумленных сигналов с помощью фреймов Вейля-Гейзенберга..... 42

Валицкая Н.С., Власюк И.В.

Методы синхронизации потоков в видеоинформационных системах 51

Офицера К.Б., Лобов Е.М.

Разработка и проектирование устройства предварительной фильтрации с использованием гибридных банков фильтров 58

Шмаков Н.Д., Иванюшкин Р.Ю.

Исследование параметрических нелинейных искажений в усилителях с распределенным усилением 63

Овчинников В.И., Логвинов В.В.

Многостанционный доступ в сетях системы 5G 69

«Сетевые технологии и системы телекоммуникаций»

Басараб М.А., Бельфер Р.А., Кравцов А.В., Орлов В.Г.

Алгоритмы передачи данных в имитаторе объединенной сети ПД специального назначения 76

Канищева М.Г., Железов Д.Б., Маликова Е.Е.

Разработка лабораторных работ по изучению и исследованию мультисервисных сетей связи..... 82

Кузнецов П.Г., Данилов А.Н.

Анализ спецификации open flow концепции SDN для сетей будущего поколения..... 89

| | |
|---|------------|
| Шишкин М.О., Степанов М.С. Разработка и анализ математической модели обслуживания вызовов в перспективных контакт-центрах | 96 |
| «Информационные технологии и автоматизация процессов в системах связи» | |
| Акопян В.А., Беленькая М.Н. Обзор микросхем ASIC и FPGA и их роль в системе мониторинга компьютерных сетей и анализе пакетов | 103 |
| Кошечаров Д.Я., Беленькая М.Н. Обзор функциональных возможностей программного продукта RIVERBED® STEELCENTRAL™ ETPROFILE | 109 |
| Ефименков Д.Д., Депутатов Е.А., Воронцов Ю.А. Оптимизация процесса оплаты в ресторанном бизнесе при помощи современных распределенных IT-решений..... | 113 |
| Груздев С.П., Шелухин О.И. Бинарная классификация компьютерных атак на информационные ресурсы при помощи нечёткой логики | 116 |
| Летунов А.С., Докучаев В.А. W-code платформы, как инструмент разработки программного обеспечения | 124 |
| Нуралин Д.Г., Шевелев С.В. Сравнительный анализ современных методов сжатия изображений без потерь | 130 |
| Векшин Р.В., Буянов Б.Я. Методы обнаружения аномалий в потоковых видеоданных | 136 |
| Петухов Д.А., Докучаев В.А. Исследование подходов к управлению рисками при создании облачной инфраструктуры территориально-распределённой компании | 141 |
| Абросимов Д. Д., Кононов Р.В., Беленькая М.Н. Машинное обучение в игровых проектах | 147 |
| Келимов Э. Р., Гузеев А. В. Построение каскадной схемы эффективного кодирования геномных данных с удалением разных типов избыточности данных | 151 |

ПЯТЬ МИФОВ О СПУТНИКОВОЙ СВЯЗИ

Алешин Виктор Сергеевич
МТУСИ, к.т.н., с.н.с., вед. научн. сотр.
abc082012@yandex.ru

Ключевые слова: спутниковая связь, сети спутниковой связи, земная станция, спутниковая радиолиния, частотный спектр, орбитальная группировка, задержка передачи сообщений, космический Интернет.

Показано: что спутниковые сети связи не могут стать альтернативой наземным магистральным сетям связи, а должны лишь дополнять последние, как средство доступа к ним из малонаселённых и труднодоступных регионов земного шара; что средние значения задержек передачи сообщений в высокоорбитальных системах спутниковой связи соизмеримы с задержками в сетях спутниковой связи, построенных на базе низкоорбитальных группировок; что низкоорбитальные спутниковые системы связи не могут обеспечить высокоскоростной доступ в сеть Интернет непосредственно с портативных абонентских терминалов (типа смартфонов и планшетов). Опровергаются некоторые другие распространённые заблуждения о возможностях и перспективах развития спутниковой связи.

Введение

В последние годы резко возрос интерес к спутниковой связи, особенно, к системам спутниковой связи (ССС) на базе «низколётов» – *искусственных спутников Земли* (ИСЗ) на низких круговых околоземных орбитах высотой от 600 до 1500 км. Этот интерес, в значительной мере, был вызван намерением британской компании *OneWeb LLC* построить глобальную спутниковую сеть доступа в Интернет на базе около 700 миниспутников, вращающихся вокруг Земли на высотах 800...1200 км [3, 4], а также последующими многочисленными заявлениями руководителя компании *SpaceX* Илона Маска о планируемом создании с той же целью грандиозной сети *Starlink* [16] на базе ещё более мощной группировки из 12-ти тысяч ИСЗ на разных высотах – от 336 до 1325 км.

Сейчас около 40 подобных проектов активно разрабатывается во всём мире, начиная от США и Китая и заканчивая Испанией, Швейцарией и Лихтенштейном [5, 7, 8, 10]. Россия также приняла участие в этой «гонке», предложив, в частности, свой проект «Эфир», входящий в многоцелевой комплексный проект «Сфера» [6], и призванный конкурировать с *OneWeb* и *Starlink*.

В связи с этим в Интернете и даже в специальных, профильных изданиях распространяется много необоснованных утверждений и попросту мифов о перспективах и возможностях СССР, некоторые из них и будут рассмотрены ниже.

Миф №1: «Спутниковая связь - альтернативный путь развития телекоммуникаций в мире»

В настоящее время действующими СССР, такими как *Iridium-Next*, *Globalstar* [1,18], *03b* [10] и другими передаётся совершенно ничтожная доля объёма мирового телекоммуникационного трафика. Предполагается, что новые ИСЗ с высокой пропускной способностью – класса HTS (*High Throughput System*) с многолучевым «покрытием» поверхности Земли и повторным использованием частот [10, 11], на которые и ориентированы вышеупомянутые проекты, кардинально изменят это положение [7], вплоть до создания сети «*космического Интернета*».

Однако это не так, поскольку спутниковая связь использует принципиально ограниченный природный ресурс – *частотный спектр*, острый дефицит которого наблюдается уже сейчас, и в дальнейшем будет ограничивать потенциальную пропускную способность СССР.

Освоение новых более высокочастотных диапазонов спектра (70 ГГц и выше) затрудняется быстрым ростом затухания радиосигналов в атмосфере с повышением частоты, что, несмотря на воз-

возможность создания при этом спутниковых бортовых антенн с меньшими углами раскрытия диаграмм направленности (и большими коэффициентами усиления), требует наращивания суммарной мощности бортовых передатчиков ИСЗ. Но эта мощность ограничивается энергетическим потенциалом спутника, получающим энергию только от солнечных батарей.

Таким образом, спутниковые сети связи, по-видимому, никогда не смогут стать альтернативой наземным магистральным сетям, уже построенным на базе *волоконно-оптических линий связи* (ВОЛС) с огромной пропускной способностью (терабайты в секунду и более) (рис. 1). Они могут лишь дополнять наземные сети как для обеспечения доступа к ним, в первую очередь, из малонаселённых, труднодоступных и океанических регионов земного шара, так и для создания резервных магистральных линий связи.



Рис. 1. Глобальная сеть ВОЛС

Миф №2: «Низкоорбитальные многоспутниковые системы связи обеспечивают намного меньшее время задержки передачи сообщений, чем высокоорбитальные»

Действительно, широко распространено мнение, что задержки передачи сообщений существенны только для *высоких орбит* (ВО): *геостационарной* (ГСО) и *высокоэллиптической* (ВЭО), в меньшей степени для *средних* по высоте, но не для *низких орбит* (НО). На самом деле это утверждение справедливо только для «односкачковых» радиолиний спутниковой связи (рис. 2) – 6-10 мс для ИСЗ на НО против 260 мс для ИСЗ на ВО.

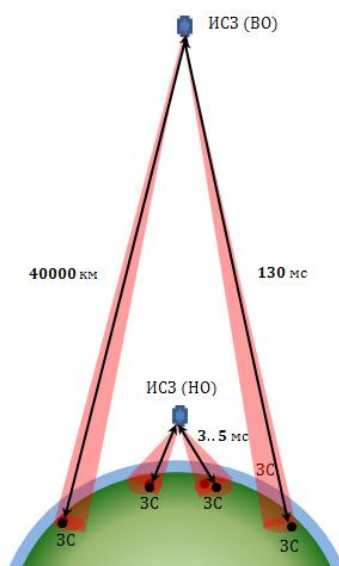


Рис. 2. Радиолинии спутниковой связи через один ИСЗ

Для низкоорбитальных спутников диаметр зоны обслуживания на земной поверхности из-

меряется, в лучшем случае, всего в 1,5-3,5 тыс. км¹, причём эта зона достаточно быстро перемещается по поверхности земного шара. Непосредственная ретрансляция сигнала через один ИСЗ возможна только для *земных станций* (ЗС)², находящихся в пределах обслуживаемой этим спутником зоны.

Для обеспечения связи «постоянной готовности» необходимо создание *орбитальной группировки* (ОГ) из нескольких десятков, а иногда и сотен таких ИСЗ (рис. 3, показана ОГ ССС Iridium-Next). При этом спутники должны быть расположены на круговых околоземных орбитах в нескольких *орбитальных плоскостях* с наклоном последних к плоскости экватора близком к полярному (что необходимо для обслуживания высокоширотных территорий земного шара и, в частности, России).

Поэтому их использование в качестве таких ретрансляторов возможно лишь для организации *локальных сетей* связи на ограниченных территориях [8]. Например, в случае ликвидации последствий чрезвычайных ситуаций, над полем боя и т.п. (Впрочем, в этих случаях целесообразнее применение ретрансляторов, установленных на *беспилотных летательных аппаратах* (БПЛА), барражирующих на значительно меньших высотах – 20-25 км [14]).

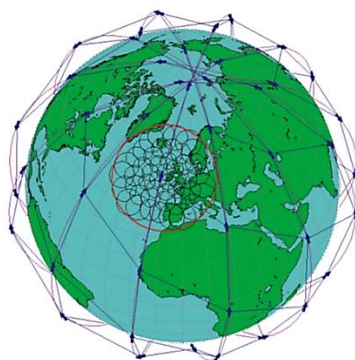


Рис. 3. Низкоорбитальная группировка ИСЗ

Для организации спутниковой связи на обширных территориях, а тем более по всему земному шару, то есть в *глобальном масштабе*, необходимо создание *сети спутниковой связи*, состоящей из *орбитального и наземного сегментов*.

Спутники могут соединяться между собой с использованием *межспутниковых радиолиний* и/или через *наземный сегмент* сети. Этот сегмент включает в себя наземные *шлюзовые станции* (ШС), соединённые между собой и с другими сетями связи волоконно-оптическими или какими-либо другими линиями связи. Связь между ИСЗ и ШС обеспечивается *фидерными линиями* с достаточно высокой пропускной способностью.

В *сетях спутниковой связи* ситуация с задержками передачи сообщений совершенно иная, чем в радиолиниях. Как показывают расчёты автора [17], уменьшение высоты ОГ ИСЗ не приводит к пропорциональному сокращению среднего времени распространения сигналов в сети спутниковой связи.

Наращивание численности спутников в низкоорбитальной группировке также не даёт значимых результатов, но сильно усугубляет проблему «замусоренности» околоземного космического пространства [15].

В то же время, диаметр зоны обслуживания Земли с спутника-ретранслятора на ВЭО, например, типа «Молния», весьма велик – более 10 тыс. км (при том же угле места 30°) [2]. Поэтому ИСЗ с ретрансляторами на этих орбитах позволяют организовать передачу сигнала на значительных территориях (например, в Арктическом регионе) всего «в один скачок», т.е. через один

¹ При высоте орбиты от 600 до 1500 км и угле места 30°

² Под ЗС здесь понимаются не только стационарные, но и подвижные станции спутниковой связи наземного, морского и воздушного базирования.

из четырёх регулярно сменяющих друг друга через каждые шесть часов спутников-ретрансляторов [2, 12]. Кроме того, в наземном сегменте достаточно всего одной ШС, расположенной в средних широтах, например, в московском регионе. Следовательно, задержки передачи сообщений в наземном сегменте сети невелики.

Средние задержки передачи сообщений в ССС на базе ВЭО и ГСО оказались соизмеримыми с задержками как в глобальных, так и в региональных сетях спутниковой связи, построенных на базе низкоорбитальных группировок. Это объясняется тем, что в последних значительную долю времени занимает передача информации по наземному и орбитальному (между спутниками) сегментам сети.

С учётом процедур аналого-цифрового и цифро-аналогового преобразований, пакетирования цифрового потока, протоколов передачи данных (ПД), которые тоже вносят значительные задержки в передачу сообщений, разница ещё больше нивелируется.

В связи с этим низкоорбитальные спутниковые системы связи не могут эффективно использоваться и для удалённого дистанционного управления беспилотными транспортными средствами, в первую очередь, БПЛА (кроме случая, упомянутого выше).

Более того, быстрые изменения задержек во время сеансов связи, присущие именно низкоорбитальным спутниковым группировкам, приводят к возникновению сильно выраженного эффекта Доплера, существенно затрудняющему приём радиосигналов [13]. Они также резко снижают эффективность работы большинства телекоммуникационных протоколов, в первую очередь высокопроизводительных синхронных протоколов *радиодоступа*. Примерно так же, как задержка распространения радиосигнала, меняется и его затухание в спутниковой радиолинии, что наряду с другими негативными факторами, превращают её в низкокачественную линию связи с явно выраженными переменными параметрами.

Поэтому, на фоне несоизмеримо меньших затрат на создание и эксплуатацию высокорбитальных ССС по сравнению с многоспутниковыми низкоорбитальными, меньшие задержки передачи сообщений не могут считаться серьёзным преимуществом последних.

Миф №3: «Низкоорбитальные многоспутниковые системы связи могут обеспечить высокоскоростной доступ во всемирную сеть Интернет с портативных смартфонов и планшетов».

Именно этот тезис активно «мультируется» в средствах массовой информации. Для его опровержения нет необходимости проводить расчёты энергетического баланса радиолиний спутниковой связи. Достаточно в качестве ориентира принять наиболее технически совершенную низкоорбитальную ССС Iridium-Next. Её *абонентский терминал* (АТ) с малонаправленной антенной обеспечивает максимальную скорость приёма-передачи в 2,4 кбит/с. При этом средняя излучаемая мощность передатчика АТ составляет примерно 400 мВт, пиковая – 4 Вт [18].

Чем выше скорость передачи информации в канале, тем большая мощность потребуется. Чтобы увеличить скорость всего лишь до 24 кбит/с, даже при прямо пропорциональном расширении полосы передаваемых частот³, необходимо повысить передаваемую мощность в 10 раз, а до приемлемых для примитивного Интернета 2,4 Мбит/с – уже в 1000 раз, что, конечно, никак недопустимо как по «санитарным» соображениям, так и из-за ограниченного энергоресурса АТ. Естественно, что аналогичное рассуждение справедливо и для радиолинии «вниз» – от спутника до АТ.

Поэтому обеспечить высокоскоростной доступ в Интернет непосредственно с портативного АТ с ненаправленной или малонаправленной антенной нереально – необходимо применять «базовые» или «терминальные» ЗС с *узконаправленными следящими антеннами*. Эти ЗС, в свою очередь и в пределах своих зон обслуживания, ретранслируют сигналы для одного или многих АТ, например, с использованием технологии *Wi-Fi*. Именно так и планируется сделать, например, в ССС *OneWeb*.

³ Увеличение скорости в той же полосе за счёт применения многоуровневых методов модуляции частот потребует увеличения мощности почти в квадратичной зависимости



Рис. 3. Земные станции ССС OneWeb

Миф №4: «Низкоорбитальные ССС позволяют использовать значительно менее мощные передатчики, чем высокоорбитальные»

Это утверждение было справедливо для спутников связи, в которых не применялись *многолучевые антенны* (МЛА). В современных ИСЗ, особенно, класса HTS, зона на земной поверхности, обслуживаемая каждым спутником, разделяется на отдельные участки по числу *лучей*⁴ – отдельных радиолиний (работающих, например, на разных частотах) [1, 3, 10, 11], то есть имеет *сотовую структуру*, как показано, например, на рис. 3 и рис. 4 (цифрами указаны номера лучей).

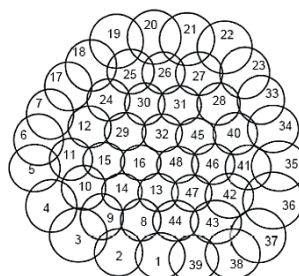


Рис. 4. Многолучевая зона обслуживания сотовой структуры

В принципе, одинаковые по площади «соты» могут быть сформированы как «низколётом», так и спутником на высокой орбите, например, «Экспресс-РВ» [12] (рис. 5) – это зависит только от ширины диаграмм направленности, а, следовательно, и коэффициента усиления приёмо-передающих антенн на борту ИСЗ.

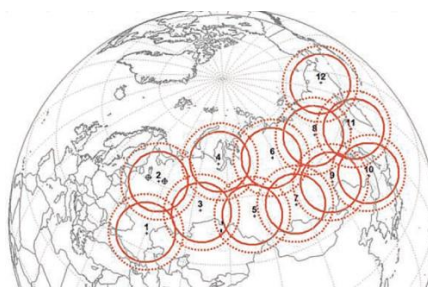


Рис. 5. Многолучевая зона обслуживания ССС «Экспресс-РВ»

Миф №5: «Будущее принадлежит микро- (до 50 кг) и нано- (менее 10 кг) спутникам связи на низких и средних орбитах»

Такие проекты крайне привлекательны ввиду относительно низких капитальных затрат за счёт возможности вывода на орбиту нескольких десятков ИСЗ одной ракетой-носителем. Они действительно разрабатываются, например, (*AISTech* в Испании, *Astrocast* в Швейцарии, *Blink Astro* в США) и даже реализуются (*Analytical Space* и *Kepler*, США). Их основное целевое назначение позиционируется как «*Интернет-вещей*» (*IoT*). Однако, для полноценного *IoT* такая идея сегодня технически сомнительна, в первую очередь, в силу необходимости обеспечения должной энерго-

⁴ ИСЗ "Экспресс-РВ" имеет только 12 лучей, *Iridium-Next* – 48, *Thuraya* – более 200.

вооруженности и коррекции положения спутников на орбите при *сроке его активного существования* хотя бы до 5-ти лет.

Для обеспечения даже низкоскоростной ПД, но в практически непрерывном режиме, ИСЗ должен обладать определённым энергетическим потенциалом (как минимум, десятки ватт мощности). Поэтому он должен нести на своём борту соответствующие солнечные батареи и энергоёмкие, а, следовательно, тяжёлые аккумуляторы (для работы в области земной тени)⁵.

Спутники на низких орбитах в большой степени подвержены тормозящему влиянию верхних слоёв атмосферы Земли и неравномерному притяжению последней (из-за аномалий её гравитационного поля). В результате этого высота орбиты ИСЗ постоянно уменьшается и должна периодически корректироваться реактивными двигателями, использующими то или иное рабочее тело, запас которого на борту, естественно, ограничен.

Для сохранения необходимой направленности антенн и солнечных батарей платформа такого ИСЗ должна иметь «трёхосную» ориентацию в пространстве с достаточно высокой точностью (не более одного или нескольких градусов), то есть она должна быть оснащена тяжёлыми маховиками, гироскопическим и прочим оборудованием с немалыми массогабаритными характеристиками.

Поэтому целям *IoT* более отвечают миниспутники массой 50 кг и более. Такой проект («Марафон»), также входящий в программу «Сфера» [6], разрабатывается, в частности, и в России.

Заключение

Таким образом, при анализе возможностей и перспектив развития спутниковой связи необходимо применять системный подход и обязательно учитывать самые разные факторы, присущие этой технологии.

Литература

1. *Немировский М.С., Локишин Б.А., Аронов Д.А.* Основы построения систем спутниковой связи / Под ред. д.т.н., проф. М.С. Немировского. М.: Горячая линия – Телеком, 2016. 432 с.
2. *Камнев Е.Ф., Аболищ А.И., Акимов А.А. и др.* Системы спутниковой связи с эллиптическими орбитами, разнесением ветвей и адаптивной обработкой. М.: Глобсатком, 2009. 724 с.
3. *Антилогов В.Р., Гриценко А.А.* Анализ многолучевой рабочей зоны спутников OneWeb // Технологии и средства связи. 2016, № 6-2, специальный выпуск «Спутниковая связь и вещание – 2017». С. 78-84.
4. *Антилогов В.Р., Гриценко А.А., Чекушин Ю.Н.* Результаты анализа совместной работы систем OneWeb и «Экспресс-РВ» в Ku-диапазоне // Технологии и средства связи. 2018, № 6-2, специальный выпуск «Спутниковая связь и вещание – 2019». С. 48-54.
5. *Водянова С.* Китай включился в гонку низкоорбитальных проектов // Технологии и средства связи. – 2018, № 6-2, специальный выпуск «Спутниковая связь и вещание – 2019». С. 76-78.
6. Обсуждение целесообразности комплексной программы «Сфера» // Технологии и средства связи. 2018, № 6-2, специальный выпуск «Спутниковая связь и вещание – 2019». С. 22-31.
7. Эволюция рынка спутниковой связи и вещания // Технологии и средства связи. 2016, № 6-2, специальный выпуск «Спутниковая связь и вещание – 2017». С. 54-55.
8. *Акимов А.А., Гриценко А.А., Степанов А.А., Чазов В.В.* Особенности построения и эксплуатации орбитальных группировок систем спутниковой связи // Технологии и средства связи. – 2015, № 6-2, специальный выпуск «Спутниковая связь и вещание – 2016». С. 72-87.
9. *Антилогов В., Шишлов А., Эйдус А.* Анализ систем LEO-HTS и реализуемости фазированных антенных решеток для абонентских терминалов // Технологии и средства связи. 2015, № 6-2, специальный выпуск «Спутниковая связь и вещание – 2016». С. 14–26.
10. *Гриценко А.А.* Спутниковые системы класса HTS // Connect – 2017, № 4. С. 120–126.
11. *Ланин К.* Спутники высокой пропускной способности // Технологии и средства связи – 2015, специальный выпуск «Спутниковая связь и вещание-2015». С. 22-23.

⁵ Здесь уместно заметить, что ИСЗ на рабочем участке ВЭО постоянно остаются освещёнными солнцем, так что необходимость в энергоёмких аккумуляторных батареях отпадает.

12. Локишин Б.А. «Экспресс-РВ» – перспективная система связи со спутниками на высокоэллиптических орбитах // Технологии и средства связи – 2018, № 6-2, специальный выпуск «Спутниковая связь и вещание – 2019». С. 62–71.

13. Карпунин Е.О., Мазена Р.Б., Михайлов В.Ю. Исследование перспективных сигнально-кодовых конструкций на основе FH-OFDM при воздействии доплеровского сдвига частоты // Научные технологии в космических исследованиях Земли – 2016, Т. 8, № 1. С. 12–16.

14. Аганесов А.В., Макаренко С.И. Балансировка информационной нагрузки между воздушным и космическими сегментами объединённой воздушно-космической сети связи, построенной на основе Mesh-технологий // Научные технологии в космических исследованиях Земли – 2016, Т. 8, № 1. С. 17–25.

15. Вениаминов С.С. Космический мусор – угроза человечеству. М.: ИКИ РАН, 2013. 207 с.

16. Peter B. de Selding Intelsat Asks FCC To Block SpaceX Experimental Satellite Launch / SpaceNews. July 22, 2015.

17. Алешин В.С., Догаев С.Г. Задержки распространения сигналов в сетях спутниковой связи // Т-Comm: Телекоммуникации и транспорт. 2019. Том 13. №5. С. 4–11.

18. Макаренко С. И. Описательная модель системы спутниковой связи Iridium // Системы управления, связи и безопасности. 2018. № 4. С. 1-34. URL: <http://sccs.intelgr.com/archive/2018-04/01-Makarenko.pdf>.

FIVE MYTHS ABOUT SATELLITE COMMUNICATIONS

Victor S. Aleshin

Leading scientific employee of MTUCI,

PhD, senior scientific researcher

abc082012@yandex.ru

Key words: *satellite communications, satellite communication networks, earth station, satellite radio line, frequency spectrum, orbital grouping, message transmission delay, space Internet.*

Shown: **satellite communication network can not be an alternative to terrestrial backbone networks and should complement the latter, as a means of access to them from sparsely populated and remote regions of the globe; that the average delays of message transmission in high-orbit satellite communication systems commensurate with the delay in the satellite communications networks built on the basis of low-orbit groups; that low-orbit satellite communication systems cannot provide high-speed Internet access directly from portable subscriber terminals (such as smartphones and tablets). Refute some other common misconceptions about the possibilities and prospects of satellite communications.**

ЭНЕРГЕТИЧЕСКИЙ ВЫИГРЫШ ОТ ВВЕДЕНИЯ АВТОМАТИЧЕСКОЙ РЕГУЛИРОВКИ РЕЖИМА ПО ПИТАНИЮ В ЛИНЕЙНЫЙ УСИЛИТЕЛЬ МОЩНОСТИ ПО СХЕМЕ У. ДОГЕРТИ

Разин Кирилл Олегович

Студент группы МРА1801 МТУСИ

razin.k2014@yandex.ru

Иванюшкин Роман Юрьевич

МТУСИ, к.т.н, доцент кафедры РОС

rivanyushkin@gmail.com

Ключевые слова: метод Уильяма Догерти, автоматическая регулировка режима по питанию, энергетический выигрыш, нелинейные искажения.

Рассмотрена идея совмещения двух методов повышения КПД усилителей мощности передатчиков цифрового телевидения: метода У. Догерти и метода нелинейной автоматической регулировки режима по питанию (АРР). Представлена компьютерная модель усилителя У. Догерти с введенной нелинейной АРР и полученные на ее основе характеристики, отражающие энергетическую эффективность и линейность предложенного комбинированного метода.

В связи с современной тенденцией совершенствования цифровых телевизионных передатчиков [5-17], связанной с активным внедрением цифрового телевизионного вещания, возрастает интерес к такому методу повышения КПД усилителей мощности этих передатчиков, как метод У. Догерти [1,3,4].

В усилителе мощности, реализованном по методу У. Догерти, применяется два усилительных прибора, работающих в существенно разных режимах на общую нагрузку [1]. Основной усилительный прибор (см. рис. 1) работает в классе АВ. Дополнительный усилительный прибор (см. рис. 1) работает в классе С. Его точка покоя выбирается таким образом, что он открывается при превышении амплитуды входного сигнала половины от пикового значения. При значениях амплитуды входного сигнала меньших, чем половинное значение, дополнительный усилительный прибор закрыт, и, как следствие, не потребляет энергии. При этом мощность обеспечивается основным усилительным прибором. За счет того, что дополнительный усилительный прибор подключается к общей нагрузке через четвертьволновую линию передачи (см. рис. 1), обеспечивается работа основного усилительного прибора в граничном режиме при значениях амплитуды входного усиливаемого сигнала от половины пикового значения до пикового. Для компенсации фазового сдвига, вносимого четвертьволновой линией передачи, на входе дополнительного усилительного прибора включена еще одна четвертьволновая линия передачи, идентичная по своим параметрам (см. рис. 1).

Таким образом, на интервале от нуля до половинного значения работает только основной усилительный прибор. Его режим работы выбирается так, что бы в точке половинного напряжения на входе он достигал граничного режима работы. В этой точке достигается максимальный КПД его работы, и, так как дополнительный усилительный прибор отключен, максимальный КПД всего усилителя. КПД на интервале от нуля до половинного значения амплитуды входного сигнала линейно возрастает (см. рис. 2). При возрастании амплитуды выше половинного значения, за счет того, что происходит включение дополнительного усилительного прибора, КПД немного уменьшается («провал» на графике

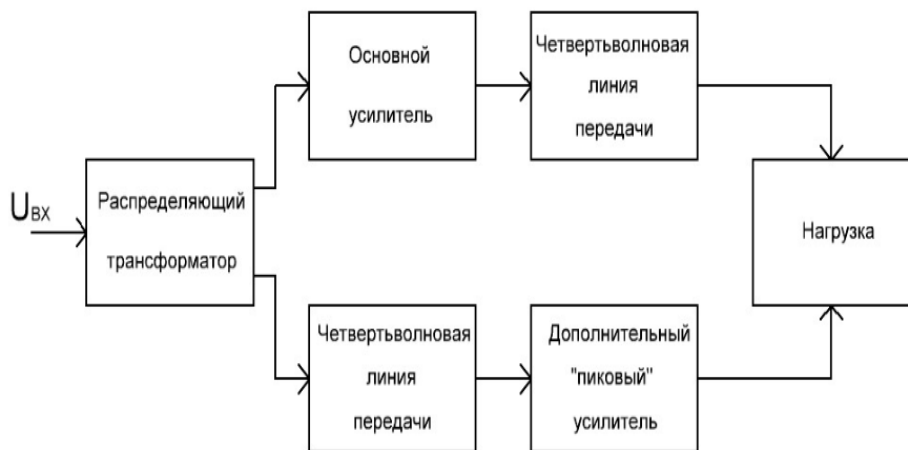


Рис. 1. Функциональная схема усилителя У. Догерти

на рис. 2), а затем, достижении максимальной амплитуды входного сигнала дополнительный усилительный прибор также достигает граничного режима работы, и КПД усилителя снова приобретает максимальное значение.

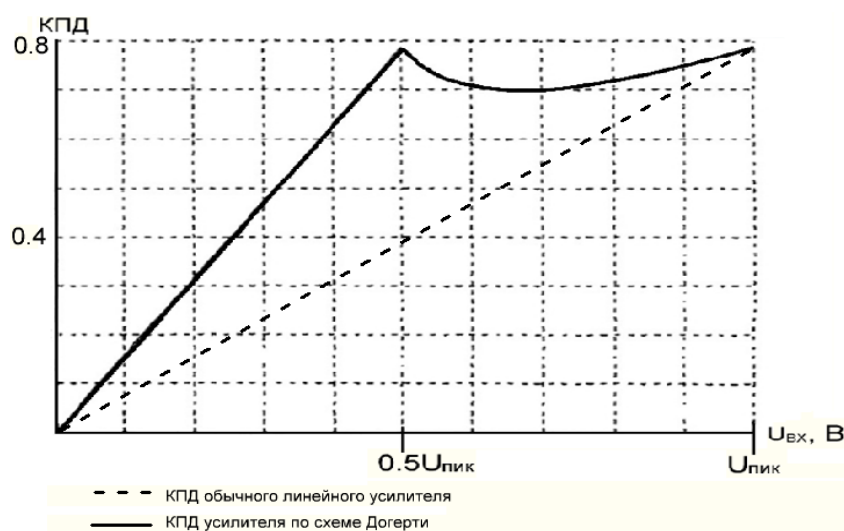


Рис. 2. Зависимость КПД усилителя У. Догерти от величины амплитуды усиливаемого радиосигнала

На графике, представленном на рисунке 2, так же видно, что при всех значениях входного сигнала КПД усилителя У. Догерти выше, чем КПД обычного линейного усилителя. Однако, при усилении OFDM сигналов, когда большую часть времени на входе усилителя амплитуды сигнала составляют не более 0.1 – 0.2 от максимальной, КПД усилителя Догерти оказывается всего на 10-15% выше, чем у обычного линейного усилителя. Этот недостаток можно устранить путем совместного применения метода Догерти и метода автоматической регулировки режима по питанию [1,2,3].

Поэтому, в качестве дальнейшего развития метода У. Догерти авторами предлагается введение в основной и дополнительный усилительный приборы метода автоматической регулировки режима по питанию.

Суть метода автоматической регулировки режима по питанию (АРР) в том, что напряжение питания усилительного прибора изменяется в соответствии с амплитудой входного сигнала таким образом, что бы поддерживать граничный режим работы усилительного прибора практически при всех значениях амплитуды входного сигнала [1]. Таким образом достигается высокий КПД усилителя.

Помимо классической (линейной) АРР по питанию, в которой напряжение питания и амплитуда усиливаемого сигнала связаны линейно, существует нелинейная АРР по питанию [2]. В случае нелинейной АРР эта зависимость имеет нелинейный характер, который учитывает нели-

нейность области полностью открытого состояния (насыщения) статических характеристик усилительного прибора. За счет этого можно добиться еще большего выигрыша в КПД. Характер такой зависимости необходимо подбирать экспериментально, что связано с различной степенью нелинейности статических характеристик у различных усилительных приборов [2].

Для предварительной оценки увеличения значения КПД, получаемого при введении нелинейной АРР по питанию, в студенческой версии программы MICROCAP 12 была создана компьютерная модель, принципиальная схема которой представлена на рисунке 3.

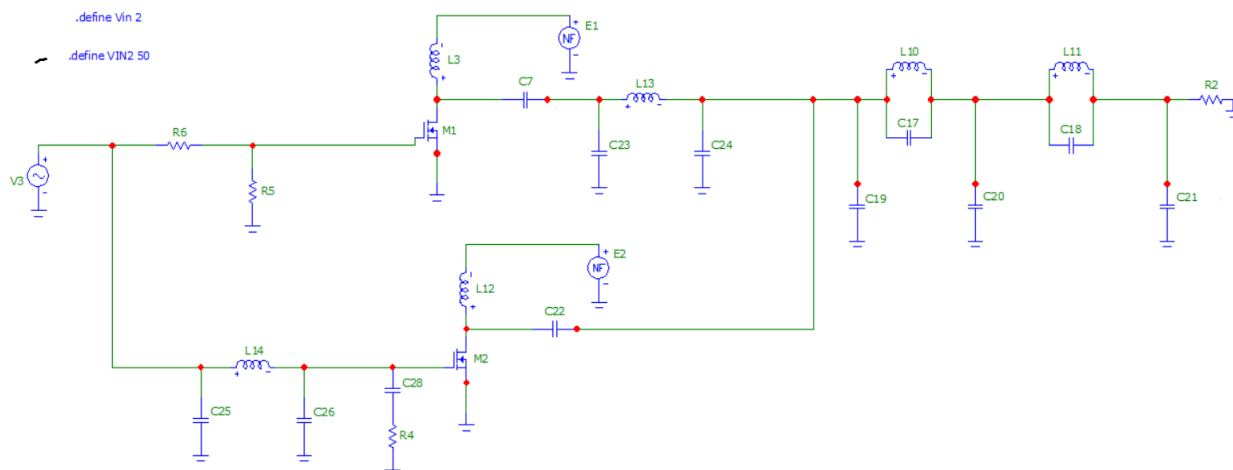


Рис. 3. Принципиальная схема смоделированного усилителя У. Догерти с введением нелинейной АРР по питанию

Элементы R5, R6 образуют делитель напряжения, создающий разные амплитуды возбуждающих колебаний на входах основного и дополнительного усилительных транзисторов. Это необходимо в связи с тем, что усилительные приборы (транзисторы) работают в существенно разных режимах [1, 3, 4]. Элементы C25, C26, L14, а так же, элементы C23, C24, L13 образуют четвертьволновые линии передачи (см. рис. 1). Источник V3 помимо входного возбуждающего сигнала вырабатывает постоянное напряжение смещения транзисторов, которое также различно, за счет делителя R5, R6. Элементы C19, L10, C17, C20, L11, C18, C21 образуют эллиптический фильтр третьего порядка, подавляющий побочные гармонические составляющие на выходе усилителя.

R2 – сопротивление нагрузки. C7, C22 – разделительные конденсаторы, не пропускающие постоянную составляющую тока в цепи фильтра. L3, L12 – блокировочные индуктивности, не пропускающие переменную составляющую со стока транзистора в источники питания. Источники питания E1 и E2 – зависимые источники питания, вырабатываемое напряжение питания которых изменяется в зависимости от амплитуды возбуждающего сигнала, вырабатываемого источником V3. Таким образом реализуется нелинейная АРР по питанию.

На основе приведенной модели была получены следующие графические зависимости:

Зависимость КПД усилителя У. Догерти от амплитуды входного усиливаемого сигнала без введения АРР по питанию (см. рис. 4).

Зависимость КПД усилителя У. Догерти с введением АРР по питанию от амплитуды входного усиливаемого сигнала (см. рис. 5).

Зависимость амплитуды выходного сигнала от амплитуды входного (амплитудная характеристика, см. рис. 6).

График зависимости сдвига фазы выходного сигнала относительно входного при изменении амплитуды усиливаемого сигнала (фазо-амплитудная характеристика, позволяющая оценить амплитудно-фазовую конверсию, см. рис. 7).

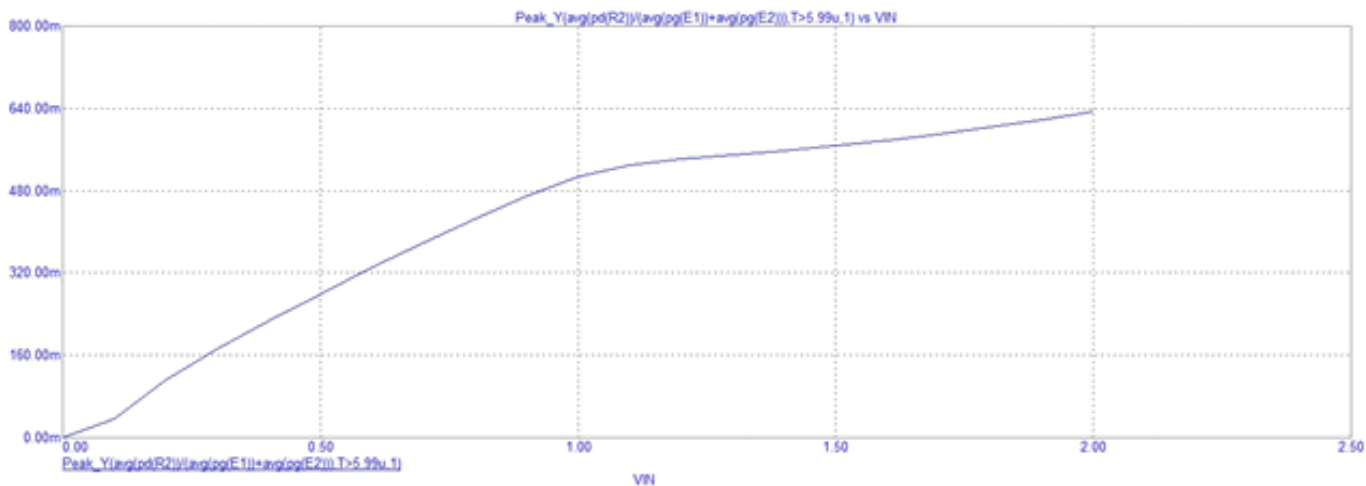


Рис. 4. Зависимость КПД усилителя У. Догерти от амплитуды входного усиливаемого сигнала без введения АРР по питанию

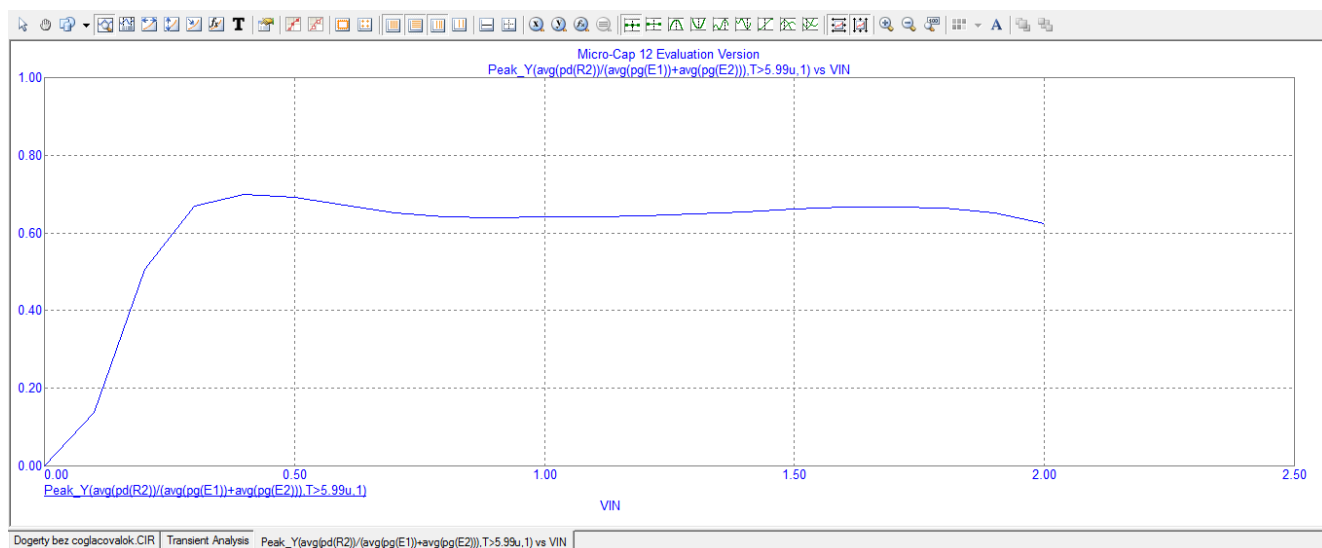


Рис. 5. Зависимость КПД усилителя У. Догерти с введением автоматической регулировки режима по питанию от амплитуды входного сигнала

Как видно при сравнении рисунков 4 и 5, применение АРР по питанию позволяет существенно повысить КПД усилителя в области малых амплитуд и средних амплитуд. При работе с OFDM сигналами КПД усилителя будет составлять порядка 40-50% (при средне-статистических значениях амплитуд OFDM сигналов). КПД классического усилителя Догерти составляет 10-15% (рис. 4).

Как видно из графика амплитудно-амплитудной характеристики (см. рис. 6), она имеет нелинейный характер. Излом характеристики при амплитуде входного сигнала, равной 0.4 вольта, связан с режимом работы основного усилительного прибора. При малых амплитудах входного сигнала основной усилительный прибор не достигает режима работы в классе АВ, а работает в классе А. В момент достижения режима работы в классе АВ амплитуда выходного сигнала начинает возрастать быстрее, за счет чего и появляется излом. В диапазонах входного сигнала от 0.6 до 0.9 Вольт и от 1.6 до 2 вольт амплитудная характеристика имеет нелинейности, хорошо аппроксимируемые полиномами второй степени. После достижения амплитуды в 2 Вольт (граница динамического диапазона) амплитудная характеристика начинает возрастать медленнее и вскоре начи-

нает идти параллельно оси абсцисс (наступает ограничение). Затем амплитуда входного сигнала начинает уменьшаться. Это связано с переходом работы усилительных приборов (транзисторов) в перенапряженный режим. Нелинейности полученной амплитудно-амплитудной характеристики целесообразно линеаризовать средствами адаптивной предкоррекции [3].

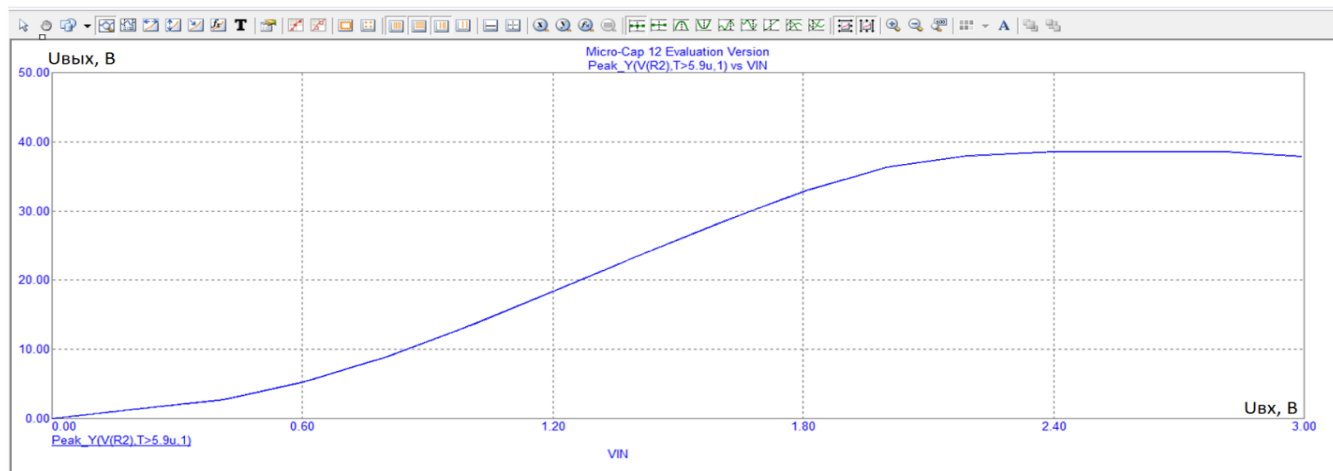


Рис. 6. Амплитудно-амплитудная характеристика

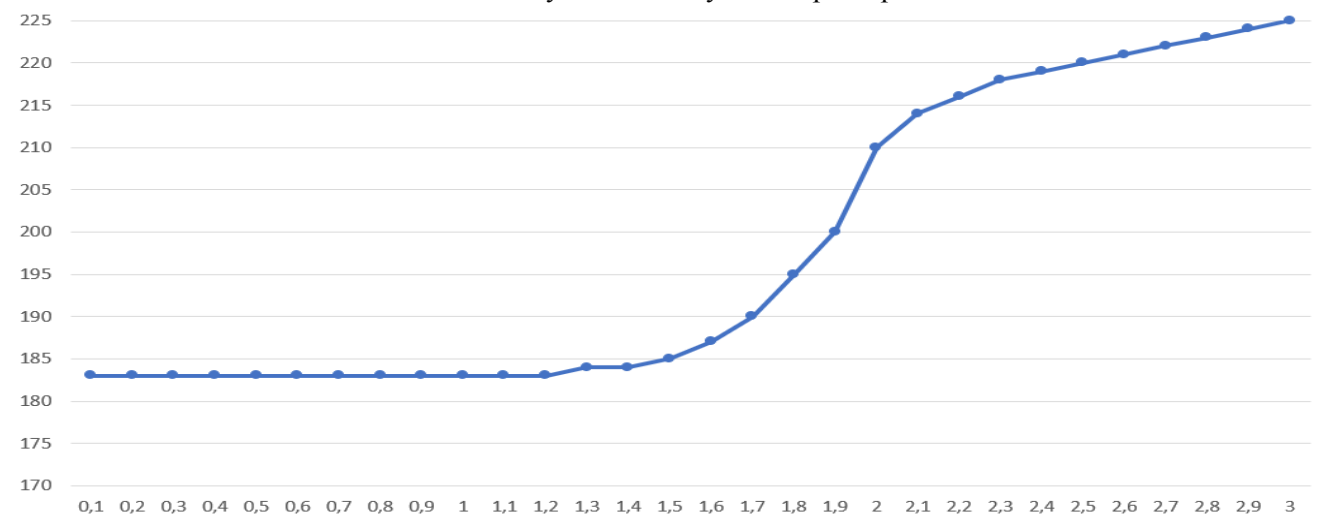


Рис. 7. Амплитудно-фазовая конверсия

Как видно из рисунка 7, на котором приведена фазо-амплитудная характеристика усилителя, сдвиг фазы выходного сигнала (относительно 180 градусов) не изменяет своего значения в диапазоне амплитуд входного усиливаемого сигнала от 0 до 1.2 Вольт. Далее сдвиг фаз начинает возрастать, что особенно резко выражено в диапазоне амплитуд входного усиливаемого сигнала от 1.7 до 2 Вольт. Такая амплитудно-фазовая конверсия связана с влиянием параметрических емкостей транзистора (емкости подложка-исток и емкости подложка-сток). Так как при увеличении амплитуды входного сигнала напряжение на стоке транзистора уменьшается, то увеличивается барьерная емкость р-п перехода сток-подложка. Это, в свою очередь, порождает увеличение фазового сдвига. Более резкое возрастание фазового сдвига в области больших амплитуд входного сигнала объясняется нелинейным характером изменения значения барьерной емкости р-п перехода, значение которой резко возрастает в области малых амплитуд на стоке. После достижения границы динамического диапазона, возрастание фазового сдвига происходит медленнее. Это связано с тем, что после выхода амплитуды входного напряжения за пределы динамического диапазона, амплитуда напряжения на стоке транзистора, вследствие его перехода в перенапряженный режим, уменьшается медленнее.

Таким образом, из полученных результатов можно сделать вывод, что введение нелинейной

автоматической регулировки режима по питанию в усилитель по схеме У. Догерти позволяет получить энергетический выигрыш, и эта задача требует дальнейших исследований: в первую очередь на предмет разработки алгоритмов линеаризации.

Литература

1. *Разин К.О., Иванюшкин Р.Ю.* Сравнение способов построения усилителей мощности передатчиков эфирного цифрового телевидения // Международная научно-техническая конференция "Перспективные технологии в средствах передачи информации – ПТСПИ–2017. 2017. Т. 3. С. 175-178.
2. *Дулов И.В., Иванюшкин Р.Ю.* Нелинейная АРР по питанию для усилителя мощности передатчика цифрового радиовещания // Т-Comm: Телекоммуникации и транспорт. 2012. Т. 6. № 9. С. 59-63.
3. *Разин К.О., Иванюшкин Р.Ю.* Проблематика построения усилителей мощности передатчиков эфирного цифрового телевидения на основе метода У. Догерти с введением автоматической регулировки режима по питанию // Фундаментальные проблемы радиоэлектронного приборостроения. 2017. Т. 17. № 4. С. 1131-1134.
4. *Nikita D. Shmakov, Roman Yu. Ivanyushkin, Kirill O. Razin* Perspective ways of construction of the tract of amplification of transmitters for digital television broadcasting // IEEE Conference Publication: 2018 Systems of Signal Synchronization, Generating and Processing in Telecommunication (SYNCHROINFO).
5. *Иванюшкин Р.Ю., Разин К.О., Шмаков Н.Д.* Перспективные пути построения тракта усиления мощности передатчиков эфирного цифрового телевизионного вещания // Системы синхронизации, формирования и обработки сигналов. 2018. Т. 9. № 3. С. 96-103.
6. *Шмаков Н.Д., Иванюшкин Р.Ю.* Моделирование усилителя с распределенным усилением на полевых транзисторах BLF278 для диапазонов ВЧ и ОВЧ // Т-Comm: Телекоммуникации и транспорт. 2017. Т. 11. № 11. С. 34-39/
7. *Шмаков Н.Д., Иванюшкин Р.Ю.* Исследование твердотельных усилителей бегущей волны диапазона ОВЧ при двух подходах к их расчету // Системы синхронизации, формирования и обработки сигналов. 2017. Т. 8. № 3. С. 88-92.
8. *Шмаков Н.Д., Иванюшкин Р.Ю.* Компьютерное моделирование усилителя с распределенным усилением на полевых транзисторах // Телекоммуникации и информационные технологии. 2017. Т. 4. № 1. С. 30-33/
9. *Шмаков Н.Д., Иванюшкин Р.Ю.* Применение технологии распределенного усиления при построении широкодиапазонных и сверхширокодиапазонных радиочастотных усилителей мощности // Телекоммуникации и информационные технологии. 2017. Т. 4. № 2. С. 46-50/
10. *Иванюшкин Р.Ю., Юрьев О.А.* Перспективы применения ключевых усилителей мощности классов D и DE при построении радиовещательных передатчиков диапазона ОВЧ // Т-Comm: Телекоммуникации и транспорт. 2016. Т. 10. № 5. С. 21-26.
11. *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.
12. *Берман А.А., Иванюшкин Р.Ю.* Проблематика построения тракта огибающей передатчиков с полярной архитектурой для цифрового радиовещания в диапазоне ОВЧ // Телекоммуникации и информационные технологии. 2016. Т. 3. № 2. С. 47-48.
13. *Шмаков Н.Д., Иванюшкин Р.Ю.* Гибридный телевизионный передатчик для цифрового и аналогового вещания // Телекоммуникации и информационные технологии. 2015. Т. 2. № 1. С. 30-33/
14. *Юрьев О.А., Иванюшкин Р.Ю.* Способы построения РЧ-тракта передатчика цифрового радиовещания диапазона ОВЧ // Телекоммуникации и информационные технологии. 2014. Т. 1. № 1. С. 8-11.
15. *Иванюшкин Р.Ю., Дулов И.В., Овчинникова М.В., Тришина Ю.А.* История и перспективы применения метода автоматической регулировки режима для повышения КПД радиопередатчиков // Т-Comm: Телекоммуникации и транспорт. 2012. Т. 6. № 9. С. 66-67.
16. *Шахгильдян В.В., Иванюшкин Р.Ю.* Методы повышения энергетической эффективности линейных усилителей мощности // Т-Comm: Телекоммуникации и транспорт. 2011. Т. 5. № 9. С. 143-145.
17. *Иванюшкин Р.Ю., Юрьев О.А.* Проблематика построения РЧ-тракта передатчиков цифрового радиовещания диапазона ОВЧ на основе метода Л. Кана // Т-Comm: Телекоммуникации и транспорт. 2013. Т. 7. № 9. С. 91-93

**RESEARCH OF ENERGY GAIN FROM USING ENVELOPE TRACKING OF POWER MODE
IN A LINEAR POWER AMPLIFIER MADE ACCORDING BY THE SCHEME OF W.
DOHERTY.**

Kirill O. Razin

Student group MPA1801 MTUCI

razin.k2014@yandex.ru

Roman Yu. Ivanushkin

MTUCI, Ph.D., associate professor of the Department of ROS

rivanyushkin@gmail.com

Keywords: *William Doherty amplifier, envelope tracking, efficiency increase, nonlinear distortion.*

The reviewed use of two methods for increasing the efficiency of power amplifiers of digital television transmitters is considered such as the method of W. Doherty and the method of envelope tracking. The computer model of the W. Doherty amplifier with envelope tracking is presented. The characteristics obtained on computer model, reflecting the energy efficiency and linearity of the combined method are presented in the article.

ПОВЫШЕНИЕ ПОМЕХОУСТОЙЧИВОСТИ МОДЕМА С ОПТИМАЛЬНЫМИ ФИНИТНЫМИ СИГНАЛАМИ, НЕ ВЫЗЫВАЮЩИМИ МЕЖСИМВОЛЬНУЮ ИНТЕРФЕРЕНЦИЮ В ЛИНЕЙНОМ КАНАЛЕ СВЯЗИ

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

Волчков Валерий Павлович
МТУСИ, д.т.н., профессор кафедры ОТС
volchkovvalery@mail.ru

Ключевые слова: цифровой когерентный модем, частотно-ограниченный канал связи, межсимвольная интерференция, оптимальные финитные сигналы, помехоустойчивость.

Рассматривается новый метод организации системы передачи данных на основе синтеза цифрового оптимального когерентного модема с сигналами, не вызывающих межсимвольной интерференции на выходе частотно-ограниченного канала связи. Приводятся новые соотношения для оптимальных финитных сигналов на входе и выходе низкочастотного эквивалента канала связи. Дается расчет потенциальной помехоустойчивости для цифрового оптимального когерентного модема. Осуществляется сравнительный анализ помехоустойчивости известной и новой схем оптимального когерентного модема. Численные результаты и эксперимент подтверждают повышенную энергетическую эффективность предложенного модема.

Введение

Важнейшей задачей развития современных систем передачи двоичных сообщений или данных (СПД) является постоянное повышение их помехоустойчивости в условиях вредного действия помех и линейных искажений в канале связи (КС) [1-4, 12-22]. Известно [5], что при увеличении скорости передачи данных по линейному, частотно-ограниченному КС все большую роль на качество их приема оказывает межсимвольная интерференция (МСИ).

Наиболее известный метод борьбы с МСИ основан на применении в приемнике СПД выравнивателя частотной характеристики КС или эквалайзера [4]. Данный метод связан с оптимизацией оператора приема, обеспечивающего максимально правдоподобные оценки сообщения по алгоритму Витерби на основе измеренной импульсной реакции канала связи и организации приема в целом. Однако, при значительных амплитудно-частотных замираниях такое выравнивание может приводить к усилению шумов на выходе демодулятора и, как следствие, к уменьшению помехоустойчивости СПД. Кроме того, с увеличением числа обрабатываемых цифровых сообщений сложность реализации такого приема резко возрастает, а помехоустойчивость СПД оценивается только приблизительно.

Альтернативный подход, рассмотренный в работе [8], связан с синтезом таких оптимальных финитных сигналов (ОФС), по форме отличных от прямоугольных, которые принципиально не вызывают МСИ на выходе частотно-ограниченного КС за счет согласования ОФС с импульсной реакцией или комплексным коэффициентом передачи КС. Таким образом, обеспечивается простой оптимальный поэлементный прием ОФС на основе согласованного фильтра (СФ) [10]. Такой метод передачи и приема ОФС позволяет точно оценивать помехоустойчивость систем передачи двоичных сообщений с многопозиционными сигналами, в частности, с квадратурной амплитудной модуляцией (КАМ).

Международным союзом электросвязи (МСЭ) разработан целый ряд методов для высокоскоростной передачи данных по частотно-ограниченным каналам связи, нашедших свое практиче-

ское применение в современных СПД на основе проводных и беспроводных технологий, закрепленных в различных стандартах [3]. Благодаря использованию ряда узкополосных фильтров на передаче и набора согласованных с ними фильтров на приеме такая технология имеет ряд преимуществ по сравнению с другими технологиями.

Так, использование узкополосных формирующих фильтров позволяет синтезировать для них оптимальные финитные сигналы (ОФС), принципиально не вызывающие на их выходах МСИ, что не требует введения защитных интервалов, а, следовательно, приводит к более высокой эффективности СПД. Приведенные достоинства СПД с узкополосными фильтрами и, согласованными с ними ОФС, обусловили повышенный интерес к ним различных исследователей. Поэтому задача разработки новых модемов с узкополосными сигналами и повышенной спектральной эффективностью, рассматриваемая в данной работе, является актуальной.

Цель настоящего исследования состоит в точной количественной оценке помехоустойчивости СПД с ОФС, не вызывающих МСИ, на основе двух вариантов реализации когерентного модема, содержащего каналы фильтры с некротными и кратными полюсами. Второй вариант рассматривается впервые.

Модель системы передачи данных

В работе рассматривается СПД, структурная схема которой приведена на рис. 1, где введены следующие обозначения: ФЗФ – формо-задающий фильтр передачи; ПФ – полосовые фильтры модема, определяющие модель частотно-ограниченного КС; СВН – схема восстановления несущей; СВТЧ – схема восстановления тактовой частоты; А/Ц – пороговый компаратор (решающее устройство). Принцип работы данной СПД очевиден и не требует детального описания.

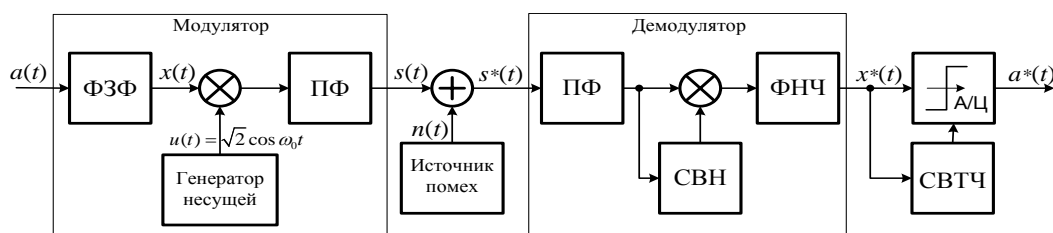


Рис. 1. Модель системы передачи данных с когерентным модемом

Анализ и реализация СПД с когерентными цифровыми модемами часто проводится на более простой эквивалентной модели в основной полосе частот (низкочастотном эквиваленте) [8].

Структурная схема низкочастотного эквивалента (НЧЭ) когерентного модема, на основе которой проводится исследование характеристик СПД, приведена на рис. 2.



Рис. 2. Низкочастотная эквивалентная модель когерентного модема

Характеристики фильтров нижних частот (ФНЧ) линейного НЧЭ КС могут быть описаны либо импульсной реакцией $q(t)$, либо комплексным коэффициентом передачи (передаточной функцией) $k(j\omega)$. В двоичных системах передачи термин символ эквивалентен биту – двоичному сигналу прямоугольной формы длительности T , со значениями $a(t) = \pm 1, 0 \leq t \leq T$, символьная скорость эквивалентна битовой скорости $V = 1/T$ (бит/с).

Постановка задачи синтеза ОФС

Пусть в схеме на рис. 2 передаточная функция $k_n(j\omega)$ порядка n НЧЭ КС, включающего ФНЧ передачи и приема, известна. Тогда, в соответствие с работой [8], синтез ОФС $x(t)$ со спектром $S_x(j\omega)$, не вызывающего МСИ на выходе данного КС, осуществляется на основе максимизации (по всем $S_x(j\omega)$) функционала следующего вида

$$J = \frac{1}{2\pi} \int_{-\infty}^{\infty} S_x(j\omega) k_n(j\omega) \Phi_n(-j\omega, T) d\omega = y(T/2), \quad (1)$$

при ограничениях: 1) значения сигнала $y(t)$ на выходе КС и его производные до n -ого порядка на концах символьного интервала T равны нулю; 2) энергия E_x сигнала $x(t)$ - фиксирована.

В (1) $y(T/2)$ - отклик КС в центре символьного интервала, $\Phi_n(j\omega, T)$ - сигнальная функция

$$\Phi_n(j\omega, T) = e^{-j\omega T/2} \left\{ 1 - 2 \sum_{k=0}^{n-1} \mu_k \omega^k \cos(\omega T/2 - k\pi/2) \right\}. \quad (2)$$

где $\mu_k, k = \overline{0, n-1}$, - множители Лагранжа решения в (1) экстремальной задачи.

На основе метода вариационного исчисления решается задача синтеза ОФС на входе $x(t)$ и выходе $y(t)$ НЧЭ КС по критерию максимизации (1). В результате искомые сигналы определяются так [8]

$$\begin{aligned} x(t) &= \varepsilon \left\{ q\left(\frac{T}{2} - t\right) - \sum_{k=0}^{n-1} \mu_k \left[\frac{d^k q(T-t)}{dt^k} + (-1)^k \frac{d^k q(-t)}{dt^k} \right] \right\}, \\ y(t) &= \varepsilon \left\{ G\left(\frac{T}{2} - t\right) - \sum_{k=0}^{n-1} \mu_k \left[\frac{d^k G(T-t)}{dt^k} + (-1)^k \frac{d^k G(-t)}{dt^k} \right] \right\}, \end{aligned} \quad (3)$$

где $q(t)$ - импульсная реакция КС, $G(t)$ - канальная функция, равная корреляционной функции от импульсной реакции $q(t)$, а ε^2 - постоянная, определяемая из соотношения

$$\varepsilon^2 = \frac{2\pi E_x}{\int_{-\infty}^{\infty} k_n^2(\omega) \Phi_n^2(\omega, T) d\omega}. \quad (4)$$

Сигналам в (3) соответствуют спектры ОФС на входе и выходе КС

$$\begin{aligned} S_x(j\omega) &= \varepsilon k_n(-j\omega) \Phi_n(j\omega, T), \\ S_y(j\omega) &= \varepsilon k_n^2(\omega) \Phi_n(j\omega, T). \end{aligned} \quad (5)$$

Особенности расчета множителей Лагранжа для различных моделей КС

Сигналы в (3) будут финитны на символьном интервале T только тогда, когда множители Лагранжа в (2) и (3) удовлетворяют заданной системе из n ограничений, введенных в (1), и определяются конкретным видом импульсной реакции КС. Так для линейного КС, характеристическое уравнение которого имеет n различных действительных корней (полюсов) $\alpha_k, k = \overline{1, n}$, импульсная реакция равна [6]

$$q(t) = \sum_{k=1}^n c_k e^{-\alpha_k t}, t \geq 0, \quad (6)$$

где $c_k, k = \overline{1, n}$ - постоянные, определяемые из n начальных условий.

В этом случае, как показано в [11], множители Лагранжа находятся из решения системы линейных алгебраических уравнений вида

$$1 - \sum_{i=0}^{n-1} \mu_i \alpha_k^i \left(e^{-\alpha_k T/2} + (-1)^i e^{\alpha_k T/2} \right) = 0, \quad k = \overline{1, n}. \quad (7)$$

В другом случае, когда характеристическое уравнение КС имеет действительные корни α_k и k -й

корень имеет кратность m_k (всего корней с учетом их кратности равно n), импульсная реакция такого КС записывается так [6]

$$q(t) = \sum_{r=0}^{m_k-1} \sum_k c_{kr} t^r e^{-\alpha_k t}, t \geq 0. \quad (8)$$

Тогда, после ряда вычислений, получена следующая система линейных алгебраических уравнений для определения множителей Лагранжа [11]

$$\frac{d^r}{d\alpha_k^r} \left\{ 1 - \sum_{i=0}^{n-1} \mu_i \alpha_k^i \left(e^{-\alpha_k T/2} + (-1)^i e^{\alpha_k T/2} \right) \right\} = 0, \quad k = 1, 2, \dots; r = \overline{0, m_k - 1}. \quad (9)$$

Характеристики помехоустойчивости приема ОФС

Перейдем к анализу помехоустойчивости СПД с когерентным модемом при различных моделях НЧЭ КС и методах организации оптимального приема ОФС. Полагаем, что эквивалентный источник помех в схеме на рис. 2 вырабатывает помеху $\xi(t)$ типа аддитивного белого гауссовского шума (АБГШ) с односторонней спектральной плотностью мощности (СПМ) G_0 . В этом случае с учетом того, что оптимальные сигналы $y(t)$ финитны на интервалах, кратных T , и не вызывают МСИ, оптимальный их прием на фоне АБГШ возможен на основе согласованной фильтрации (СФ) с импульсной реакцией [10]: $q_{SF}(t) = y(T-t), 0 \leq t \leq T$, и отношением сигнал/шум (ОСШ): $h_{SF}^2 = 2E_y / G_0$. С учетом коэффициента передачи по энергии (КПЕ) $k_E(V) = E_y / E_x$ ОСШ приводится к виду [9]:

$$h_{SF}^2 = \frac{2E_x}{G_0} k_E(V). \quad (10)$$

При равновероятностных символах и АБГШ средняя вероятность ошибки равна [9]:

$$p_e = \frac{1}{2} \operatorname{erfc} \left(\frac{h_{SF}}{\sqrt{2}} \right) = \frac{1}{2} \operatorname{erfc} \left(\sqrt{\frac{E_x}{G_0} k_E(V)} \right), \quad (11)$$

где $\operatorname{erfc}(\rho) = \frac{2}{\sqrt{\pi}} \int_{\rho}^{\infty} e^{-\lambda^2} d\lambda$.

Для получения численных результатов примем в качестве модели НЧЭ канала связи – канал первого порядка (КС1) с комплексным коэффициентом передачи

$$k_1(j\omega) = 1 / (1 + j\omega / \alpha_1), \quad (12)$$

где $\alpha_1 = \Delta\omega_{\gamma} = 2\pi F_{\gamma}$ - полоса частот КС1 по уровню ослабления, равном 3 дБ.

Помехоустойчивость передачи ОФС1 по КС1

Рассмотрим НЧЭ оптимального когерентного модема, изображенного на рис. 2, в котором КС1 – ФНЧ передачи с $k_1(j\omega)$ из (12), а ФНЧ приема представляет собой фильтр (СФ1), согласованный с ОФС1 $y_1(t)$, определяемый по (3) с учетом (12).

В таблице 1 приведены основные характеристики СПД с КС1, где КПЕ – коэффициент передачи по энергии.

Таблица 1

Основные характеристики оптимальной СПД с КС1.

| Импульсная реакция | Канальная функция | Множитель Лагранжа | КПЕ |
|---|---|--|---|
| $q_1(t) = \alpha_1 e^{-\alpha_1 t}, t \geq 0$ | $G_1(t) = 0.5 \alpha_1 e^{-\alpha_1 t }$ | $\mu_0 = 1 / 2 \operatorname{ch}(b_1),$ $b_1 = \alpha_1 / 2V$ | $k_{1,E} = 0.5 \left(1 - \frac{2b_1}{\operatorname{sh}(2b_1)} \right)$ |

На рис. 3 показаны формы и спектры оптимальных финитных сигналов первого вида (ОФС1) на входе согласованного фильтра (СФ1).

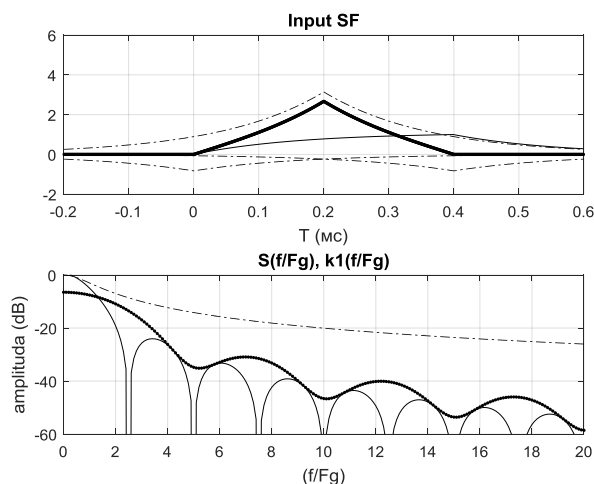


Рис. 3. Формы и спектры ОФС-1 на входе СФ1

На рис. 6 слева (нижняя кривая) приведена зависимость КПЕ $k_{1,E}$ от битовой скорости передачи V при граничной частоте ФНЧ1, равной $F_g = 1$ кГц; на правом графике этого рисунка (верхняя кривая) приведена зависимость средней вероятности ошибки от V , полученная на основе (11) с учетом $k_{1,E}$ из таб. 1.

Помехоустойчивость передачи ОФС2 по КС2

Анализируя схему СПД на рис. 1, заключаем, что в реальном случае модулятор и демодулятор имеют идентичные полосовые фильтры (ПФ), которым на рис. 2 соответствуют идентичные ФНЧ1 передачи и приема с коэффициентами передачи из (12). Но в этом случае сквозной коэффициент передачи НЧЭ КС определяется произведением вида

$$k_2(j\omega) = k_1(j\omega) \cdot k_1(j\omega) = \frac{1}{[1 + j\omega / \alpha_2]^2}, \quad (13)$$

который характеризуется двукратным полюсом, а величина α_2 определяется так, чтобы обеспечивалось ослабление модуля в (13) по уровню 3 дБ. Отсюда находим: $\alpha_2 = \alpha_1 / \sqrt{2^{1/2} - 1}$.

Новая реализация НЧЭ оптимального когерентного модема приведена на рис. 4.

Теперь задача синтеза ОФС2 должна решаться на основе соотношений в (3), с тем отличием, что в них используется новая импульсная реакция $q_2(t)$, как преобразование Фурье от (13), и новая канальная функция $G_2(t)$, как функция корреляции от $q_2(t)$, а множители Лагранжа ищутся из нового соотношения (9).

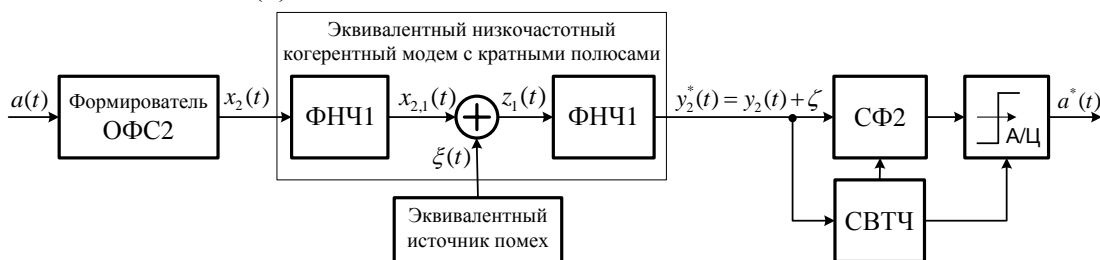


Рис. 4. Низкочастотный эквивалент оптимального модема с кратными полюсами

Отличительная особенность схемы на рис. 4 от схемы на рис. 2, в том, что здесь на входе согласованного фильтра СФ2 наблюдается смесь ОФС2 $y_2(t)$ и цветного шума $\zeta(t)$ с СПМ вида

$$G_{\zeta}(\omega) = 0.5 G_0 k_1^2(\omega, \alpha_2) = 0.5 G_0 k_2(\omega) = \frac{0.5 G_0}{[1 + (\omega / \alpha_2)^2]} \quad (14)$$

Согласно [10] ОСШ на выходе СФ2 определяется так:

$$h_{SF}^2 = \frac{1}{2\pi} \int_{-\infty}^{\infty} \frac{S_{y_2}^2(\omega) d\omega}{G_{\zeta}(\omega)}$$

С учетом (5) и (14) соотношение для нового ОСШ приводится к виду

$$h_{SF}^2 = \frac{2\varepsilon^2}{G_0} \frac{1}{2\pi} \int_{-\infty}^{\infty} k_2^3(\omega, \alpha_2) \Phi_2^2(\omega, T) d\omega,$$

Учитывая теперь соотношение (4), окончательно для ОСШ на выходе СФ2 имеем

$$h_{2,SF}^2 = \frac{2E_x \int_{-\infty}^{\infty} k_2^3(\omega) \Phi_2^2(\omega, T) d\omega}{G_0 \int_{-\infty}^{\infty} k_2^2(\omega) \Phi_2^2(\omega, T) d\omega} = \frac{2E_x}{G_0} k_{2,E}(V), \quad (15)$$

$$\Phi_2^2(\omega, T) = \left[1 - 2 \left(\mu_0 \cos \frac{\omega T}{2} + \mu_1 \omega \sin \frac{\omega T}{2} \right) \right]^2 \quad (16)$$

В таблице 2 приведены характеристики для КС2 с кратными полюсами.

Таблица 2

Основные характеристики СПД с КС2.

| Импульсная реакция | Канальная функция | Множители Лагранжа |
|---|--|---|
| $q_2(t) = \alpha_2^2 t e^{-\alpha_2 t}, t \geq 0$ | $G_2(t) = \frac{1}{4} \alpha_2 \left(\left \frac{t}{\alpha_2} \right + 1 \right) e^{-\alpha_2 t }$ | $\mu_0 = \frac{\text{sh}(b_2) + (b_2) \text{ch}(b_2)}{2b_2 + \text{sh}(2b_2)}, b_2 = \frac{\alpha_2}{2V}$ $\mu_1 = \frac{1}{2V} \cdot \frac{\text{sh}(b_2)}{2b_2 + \text{sh}(2b_2)}$ |

Формы и спектры оптимальных финитных сигналов второго вида (ОФС2) на входе согласованного фильтра (СФ2) показаны на рис. 5

На рис. 6 слева (верхняя кривая) приведена зависимость КПЕ $k_{2,E}$ от битовой скорости передачи V при граничной частоте ФНЧ2, равной $F_{\gamma} = 1$ кГц; на правом графике этого рисунка (нижняя кривая) приведена зависимость средней вероятности ошибки от V , полученная на основе (11) с учетом $k_{2,E}$ из таблицы 2.

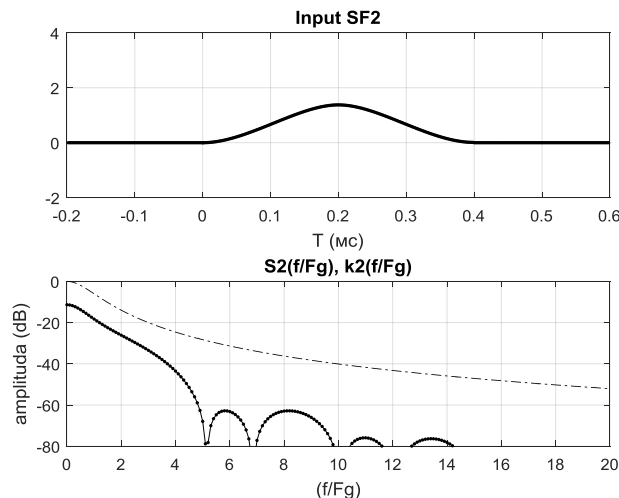


Рис. 5. Форма и спектр ОФС2 на входе СФ2

Полученные зависимости характеризуют собой потенциальную помехоустойчивость СПД с когерентным модемом, построенного на основе оптимальных финитных сигналов, не вызывающих межсимвольную интерференцию на входе решающего устройства демодулятора СПД. Причем новая схема организации оптимального НЧЭ модема с кратными полюсами (рис. 4), как это следует из рисунка 6, обладает лучшими энергетическими характеристиками по сравнению с известной. Это дополнительно подтверждается и зависимостями потенциальной помехоустойчивости для сравниваемых схем, приведенными на рис. 7.

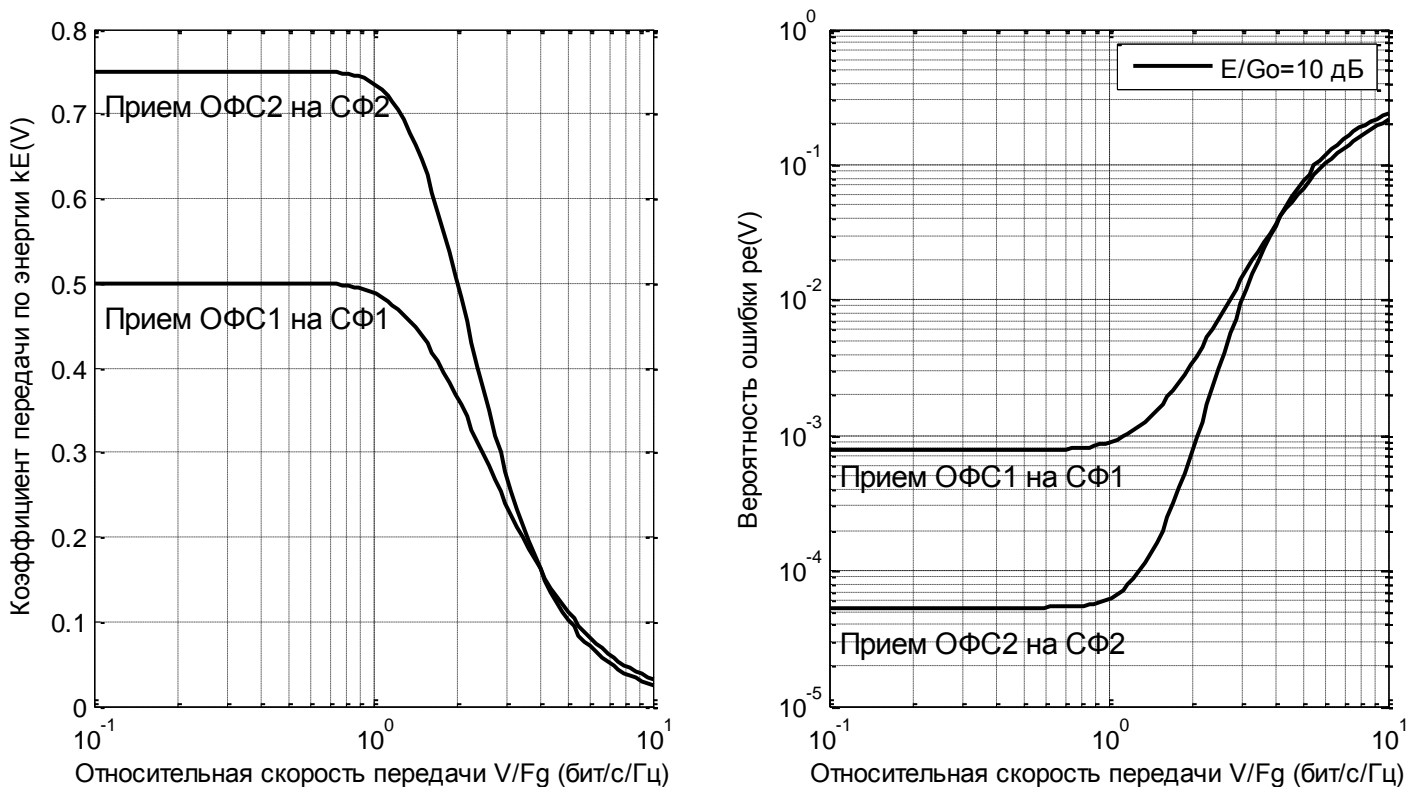


Рис. 6. Зависимости коэффициента передачи по энергии и средней вероятности ошибки от битовой скорости передачи

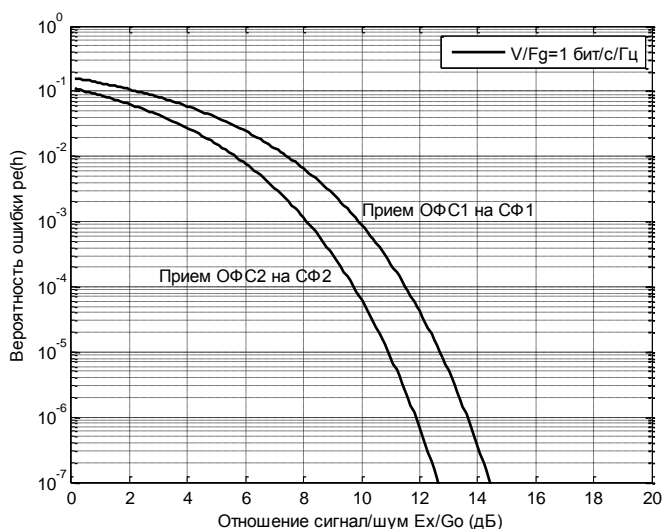


Рис. 7. Зависимости средней вероятности ошибки от отношения сигнал/шум

Эксперимент

На рис. 8 приведены экспериментальные результаты исследования (в среде MATLAB) си-

стемы передачи данных с оптимальными финитными сигналами (ОФС2), не вызывающих межсимвольной интерференции на выходе частотно-ограниченного канала связи с передаточной функцией (13). Здесь показаны (сверху в низ) сигналы в различных сечениях СПД: 1) исходное двоичное сообщение в виде меандра на входе ФЗФ (рис. 1); 2) оптимальный финитный сигнал на выходе ФЗФ (для сравнения здесь показан и сигнал прямоугольной формы); 3) оптимальный модулированный сигнал на выходе модулятора; 4) оптимальный модулированный сигнал на выходе частотно-ограниченного канала связи; 5) сигналы на выходе демодулятора (синий график – огибающая сигнала на входе демодулятора, штрих-пунктирный – отклик демодулятора на сигнал прямоугольной формы, сплошной график - ОФС); 6) сигналы на выходе согласованного фильтра демодулятора.

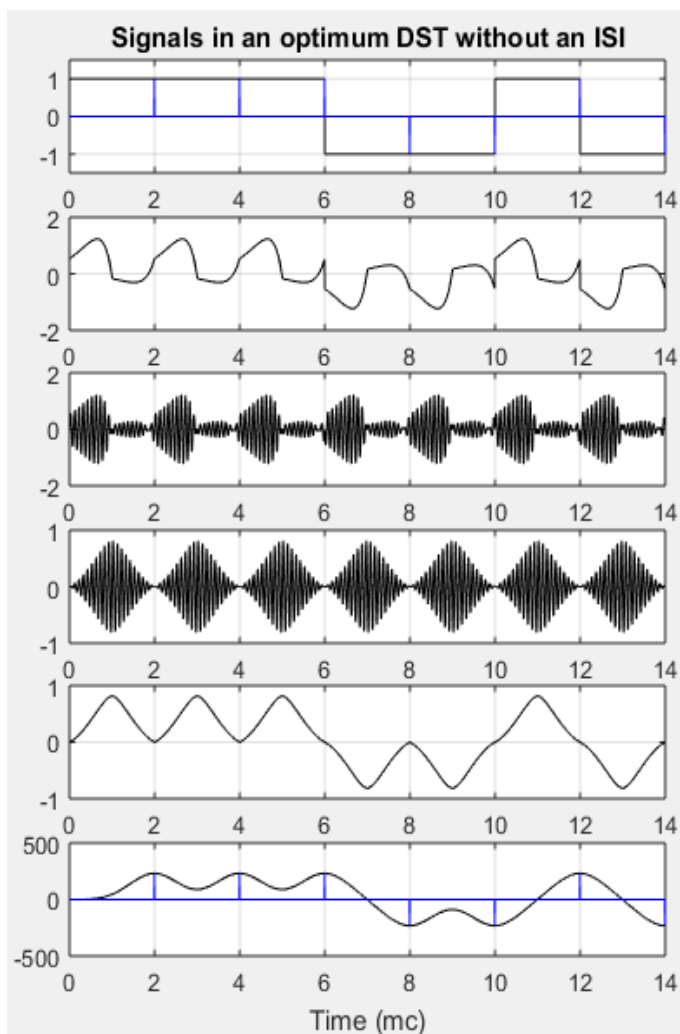


Рис. 8. Экспериментальные зависимости сигналов в различных сечениях СПД

Заключение

1. Предложена новая схема организации оптимального приема оптимальных финитных сигналов без межсимвольной интерференции в системе передачи данных с линейным частотно-ограниченным каналом связи, имеющего кратные полюса.

2. Сравнительный анализ помехоустойчивости новой схемы с известной из литературы показывает, что, например, при средней вероятности ошибки на бит 10^{-4} энергетический выигрыш нового оптимального модема составляет 1,9 дБ.

3. Разработанный метод передачи и приема оптимальных финитных сигналов по частотно-ограниченному каналу связи может быть использован в новых 5G технологиях.

Литература

1. *D. Gabor*, "Theory of communication", *J. Inst. Elect. Eng.* (London), vol. 93, no. 111, pp. 429-457, 1946.
2. *Прокуис, Джон*. Цифровая связь: Пер. с англ. / Под ред. Д.Д. Кловского. М.: Радио и связь, 2000. 800 с.
3. *Скляр, Бернард*. Цифровая связь. Теоретические основы и практическое применение. 2-е издание: Пер. с англ. Издательский дом «Вильямс», 2003. 1104 с.
4. *D. Fellows, D. Jones*, "DOCSIS Cable Modem Technology", *IEEE Communications, Mag.* no. 3, pp. 202–209, Mar. 2001.
5. *Феер К.* Беспроводная цифровая связь. Методы модуляции и расширение спектра: Пер. с англ. / Под ред. В.И. Журавлева. М.: Радио и связь. 2000. 520 с.
6. *Сиберт У.М.* Цепи, сигналы, системы: Пер. с англ. М.: Мир. 1988. 336 с.
7. *R.W. Chang*, "Synthesis of Band-Limited Orthogonal Signals for Multichannel Data Transmission", *Bell Syst. Tech. J.*, vol. 45, № 10, 1966, pp. 1775-1797.
8. *Санников В.Г.* Синтез финитных сигналов Найквиста, согласованных с телефонным каналом связи // *Электросвязь*. 2012. № 5. С. 9-12.
9. *Санников В.Г.* Помехоустойчивость системы передачи оптимальных финитных сигналов по телефонному каналу связи // *Электросвязь*. 2013. № 5. С. 39-44.
10. *Аджемов А.С., Санников В.Г.* Общая теория связи. Учебник для вузов. М.: Горячая линия – Телеком, 2018. 624.
11. *Sannikov V.G., Volchkov V.P.* "Noise stability of transmission of optimum finite signals on the linear communication channel with aliquot poles", in *IEEE Conference # 43613, 2018 Systems of Signal Synchronization, Generating and Processing in Telecommunications – SYNCHROINFO from 4 to 5 July 2018*.
12. *Волчков В.П., Уваров С.С.* Аппроксимация узкополосных случайных процессов с помощью комплексной рекуррентной m -модели скользящего окна второго порядка // *Т-Comm: Телекоммуникации и транспорт*. 2015. Т. 9. № 3. С. 54-61.
13. *Безруков И.М., Волчков В.П.* Исследование помехоустойчивости цифровой системы связи с канальным прекодером и финитной посимвольной передачей // *Телекоммуникации и информационные технологии*. 2016. Т. 3. № 1. С. 146-150.
14. *Волчков В.П., Санников В.Г.* Синтез оптимальных канальных прекодеров с весовым окном // *Системы синхронизации, формирования и обработки сигналов*. 2016. Т. 7. № 1. С. 19-21
15. *Волчков В.П., Санников В.Г.* Синтез предыскаженных финитных сигнальных базисов для борьбы с межсимвольной интерференцией // *Системы синхронизации, формирования и обработки сигналов*. 2017. Т. 8. № 3. С. 28-30.
16. *Волчков В.П., Санников В.Г.* Синтез канальных прекодеров для систем связи с финитным сигнальным базисом // *Системы синхронизации, формирования и обработки сигналов*. 2015. Т. 6. № 3. С. 152-154.
17. *Волчков В.П., Санников В.Г.* Алгебраический подход к оптимальному синтезу вещественных сигнальных базисов Вейля-Гейзенберга // *Системы синхронизации, формирования и обработки сигналов*. 2018. Т. 9. № 4. С. 19-25.
18. *Санников В.Г., Волчков В.П.* Помехоустойчивость передачи оптимальных финитных сигналов по линейному каналу связи с кратными полюсами // *Системы синхронизации, формирования и обработки сигналов*. 2018. Т. 9. № 2. С. 148-153.
19. *Санников В.Г., Алёшинцев А.В.* Многочастотный модем как один из основных элементов системы "интеллектуальное здание" при удаленном управлении объектами // *Т-Comm: Телекоммуникации и транспорт*. 2015. Т. 9. № 6. С. 21-27.
20. *Санников В.Г., Алёшинцев А.В.* Математическое моделирование многочастотного модема с повышенной помехоустойчивостью // *Т-Comm: Телекоммуникации и транспорт*. 2016. Т. 10. № 7. С. 52-58.
21. *Санников В.Г., Корольков А.А., Герасименко Х.В.* Метод совместного оценивания параметров и фильтрации зашумленной речи в голосовых кодеках // *Т-Comm: Телекоммуникации и транспорт*. 2014. Т. 8. № 12. С. 71-73.
22. *Санников В.Г., Алёшинцев А.В.* Синтез финитных сигналов, согласованных с характеристиками фильтра Баттерворта, по критерию максимума среднего значения его отклика // *REDS: Телекоммуникационные устройства и системы*. 2016. Т. 6. № 4. С. 477-481.

**IMPROVING THE NOISE IMMUNITY OF THE MODEM WITH THE OPTIMAL FINITE SIGNALS
THAT DO NOT CAUSE INTERSYMBOL INTERFERENCE IN A LINEAR COMMUNICATION
CHANNEL**

Vladimir G. Sannikov

MTUCI, PhD., professor of chair of General Communication Theory

tes_mtuci@mail.ru

Valery P. Volchkov

MTUCI, Dr.Sci.Tech, professor of chair of General Communication Theory

volchkovvalery@mail.ru

Keywords: *digital coherent modem, bandwidth-limited communication channel, intersymbol interference, optimal finite signals, noise immunity.*

A new structure of data transmission system proposed based on the synthesis of digital optimal coherent modem with signals that do not cause intersymbol interference at the output of a band limited communication channel. New analytic expressions for optimal finite signals at the input and output of the low frequency equivalent of the communication channel are presented. The potential noise immunity for the digital optimal coherent modem is calculated. A comparative analysis of noise immunity of the known and new schemes of the optimal coherent modem is carried out. Numerical results and experiment confirm the increased energy efficiency of the proposed modem.

ИССЛЕДОВАНИЕ ВОПРОСОВ ТОЧНОСТИ ФОРМИРОВАНИЯ ПОСЛЕДОВАТЕЛЬНОСТИ ЗАДОВА-ЧУ

Шубина Мария Валерьевна
МТУСИ, ассистент кафедры РТС
m.v.shubina@mtuci.ru

Косичкина Татьяна Павловна
МТУСИ, к.т.н., доцент кафедры РТС
t.p.kosichkina@mtuci.ru

Ключевые слова: синхронизация в сетях LTE, многофазные последовательности, последовательность Задова-Чу, автокорреляционная функция, последовательности с нулевой зоной автокорреляции, цифровое формирование сигналов.

Рассматриваются методы формирования синхросигналов стандарта LTE. В качестве таких сигналов, как правило, используются последовательности Задова-Чу, относящиеся к классу многофазных кодов. Обсуждается особенность структуры данных последовательностей и связанная с этим проблема их формирования в режиме реального времени. Этот вопрос является актуальным, поскольку реализация последовательности Задова-Чу в реальном времени является компромиссом между емкостью памяти и вычислительной сложностью. Приводится сравнение двух методов формирования: табличного метода и метода CORDIC, исследуется влияние этих методов на точность синхронизации, а именно проводится оценка уровня боковых лепестков периодической и аperiodической АКФ разных последовательностей данного класса при различных ограничениях.

Вопросы, связанные с обеспечением синхронизации всегда актуальны, так как синхронизация обеспечивает качественную и надежную работу любой системы. В системах стандарта LTE предъявляются высокие требования к временной, частотной и символьной синхронизации. Для обеспечения данной процедуры рекомендуется использовать псевдослучайные последовательности. Процесс синхронизации с использованием таких последовательностей основан на корреляционном анализе, посредством оценки функций авто- и взаимных корреляций. В стандарте LTE используются несколько видов псевдослучайных последовательностей, в частности, многофазные последовательности Задова-Чу (ZC). Последовательности Задова-Чу могут быть использованы в качестве следующих синхросигналов [1]: опорного сигнала восходящей линии связи (uplink, UL) канала PUSCH, опорного сигнала UL зондирования, процедуры ACK/CQI в UL, расширяющей последовательности канала UL PUSCH и в канале PRACH (физический канал произвольного доступа). Базовая станция может включать в себя генератор эталонной последовательности для приема всех каналов, использующих для синхронизации последовательности ZC.

Трудности реализации таких кодов связаны с тем, что с увеличением объема их алфавита возрастает вычислительная сложность. На данном этапе современные технические средства позволяют обеспечить быстрое формирование и обработку этих кодов, что делает их применение обоснованным и эффективным для современных систем связи [2]. Необходимость поиска упрощенных методов формирования последовательности Задова-Чу обусловлена тем, что такие последовательности трудно генерировать в режиме реального времени. Известно два метода формирования последовательностей ZC аппаратными средствами: табличный метод и метод CORDIC. И в том, и в другом случае формирование ZC последовательностей сопряжено либо с вычислительными затратами, либо с затратами на устройства памяти. Выбор того или иного метода зависит от характера применения этих последовательностей. Каждый из методов удобен для формирования сигналов различного назначения.

Наиболее простой метод формирования последовательностей ZC – это метод, когда элементы последовательности рассчитываются по каноническому уравнению с учетом четности или

нечетности длины последовательности [3]

$$\gamma_m = \exp(i\varphi_m), \text{ где}$$

$$\varphi_m = \begin{cases} \frac{2\pi}{N} q \frac{m^2}{2}, & \text{если } N - \text{четное число} \\ \frac{2\pi}{N} q \frac{m(m+1)}{2}, & \text{если } N - \text{нечетное число} \end{cases}$$

$$1 \leq m \leq N, q - \text{любое целое число взаимно простое с } N.$$

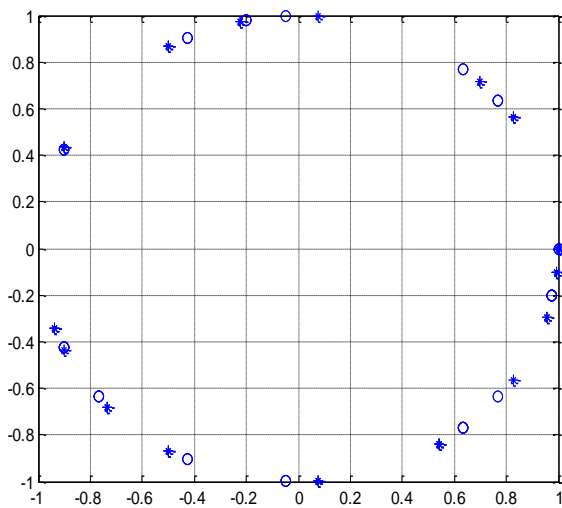
В этом случае прибегают к предварительному вычислению этих последовательностей в автономном режиме и квантованию их с требуемой точностью. Далее значения элементов последовательности могут быть записаны и в таблицу и в дальнейшем считываться из устройства памяти для быстрого формирования сигнала в режиме реального времени. Однако объем памяти, необходимый для этого, достаточно велик. Следует понимать, что точность представления данных в таблице, зависящая от разрядности цифровых устройств, может повлиять как на амплитуду последовательности, так и, в большей степени, на точность представления фазы. Очевидно, что это будет тем более критично, чем больше значение N и, соответственно, меньше расстояние между точками на фазовой плоскости.

Абонентские терминалы, напротив, вынуждены формировать данные последовательности «на лету», т.е. в режиме реального времени. Таким образом, для формирования опорных сигналов больше подходит метод CORDIC (Coordinate Rotation Digital Computer), в основу которого положен алгоритм цифрового вычисления поворота системы координат. Суть алгоритма CORDIC заключается в вычислении комплексных экспонент с использованием только операции сдвига и сложения. Алгоритм основан на рекурсивных вычислениях координат x_i, y_i - вектора и вычислении остаточного угла (по отношению к оси x) z_i который влияет на направление микро-поворота угла $\tan^{-1}(2^{-(i+1)})$. На $(i+1)$ -й итерации, где $i \in [1: B]$, B – число итераций, координаты определяются с помощью выражений [4]

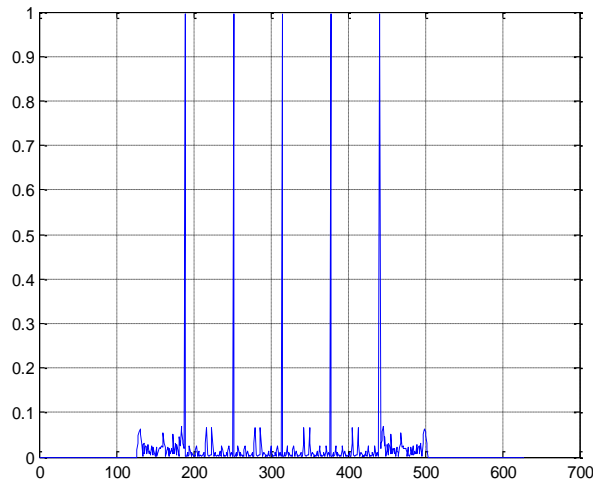
$$\begin{aligned} x_{i+1} &= x_i - y_i \cdot d_i \cdot 2^{-i} \\ y_{i+1} &= y_i + x_i \cdot d_i \cdot 2^{-i} \\ z_{i+1} &= z_i - d_i \cdot \tan^{-1}(2^{-i}) \end{aligned}$$

Для реализации алгоритма задаются начальные условия: $x_1 = K, y_1 = 0, z_1 = \theta \in [0: \pi/2]$, $K = 0,607$ – масштабирующая константа на которую умножается полученная координата. Окончательные координаты вычисленные после B итераций могут быть представлены косинусом и синусом функции $x_{B+1} = \cos(\theta)$ и $y_{B+1} = \sin(\theta)$. Нужная точность вычислений достигается увеличением числа итераций.

С целью изучения корреляционных свойств многофазных кодов последовательности Задова-Чу проведено моделирование в системе Matlab. Для начала проведена проверка точности совпадения результатов при достаточно большом числе итераций $B > 8$. Результаты вычисления значений фаз и аргументов, полученных с помощью расчета по каноническому уравнению и с помощью модифицированного алгоритма, соответственно, для последовательностей с различными значениями N и q показывают хорошее совпадение. Дальнейший интерес представляет исследование зависимости точности представления элементов последовательности Задова-Чу от числа итераций B . На рисунках 1-2 приведены результаты моделирования, а именно: а-фазовая плоскость, б-периодическая автокорреляционная функция (ПАКФ) для двух значений B : $B=4$ (рис.1) и $B=8$ (рис.2). На фазовых плоскостях осям x и y соответствуют действительная и мнимая составляющие комплексного сигнала, соответственно; обозначения (*) соответствуют значениям, рассчитанным по формуле, (о) соответствуют значениям, полученным с помощью модифицированного алгоритма. На графиках ПАКФ по оси x отложены номера отсчетов цифрового сигнала, по оси y – значение ПАКФ, нормированное к максимальному значению. Исследования показали, что уже при $B=8$ итерациях последовательности совпадают и ПАКФ стремится к идеальной.

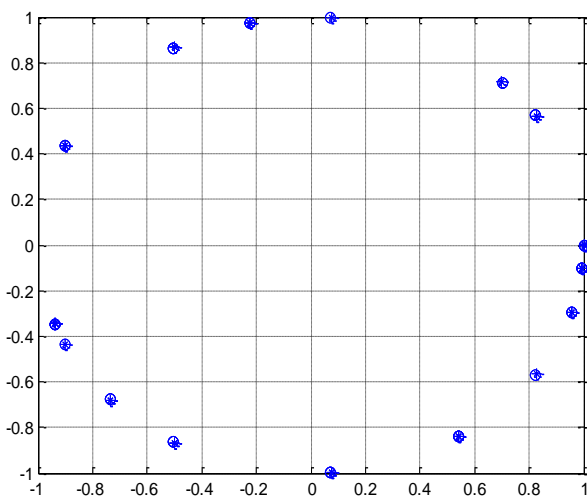


а

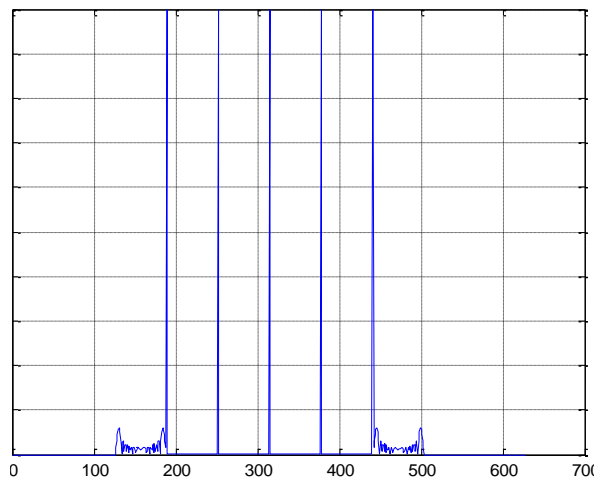


б

Рис. 1. Результаты моделирования для числа итераций $B=4$



а



б

Рис. 2. Результаты моделирования для числа итераций $B=8$

Дальнейший интерес представляет исследование средней погрешности между «идеальными», т.е. рассчитанными по формуле с двойной точностью элементами последовательности ZC и элементами последовательности ZC , генерируемой с использованием алгоритма CORDIC в зависимости от числа N элементов последовательности ZC , для различных значений B . Для различных значений длины последовательности и количества итераций погрешность вычислений определяется как абсолютная разница в величине дискретных значений между реализациями и усредняется по всем элементам последовательности.

Исследована разница в вычислении фазы для алгоритма с различным числом итераций. Замечено, что ошибка вычисления фазы носит шумовой характер и может вносить вклад в общий фазовый шум при приеме сигнала. Произведена проверка гипотезы о распределении данной ошибки с помощью метода гистограмм. Для небольшого значения длины последовательности не удалось получить удовлетворительного результата. При увеличении длины последовательности становится очевидным (рис. 3), что закон, по которому распределена ошибка формирования фазы, близок к равномерному распределению (на этом графике по оси x отложена абсолютная величина ошибки, по оси y – частота попадания значения величины ошибки в заданный интервал).

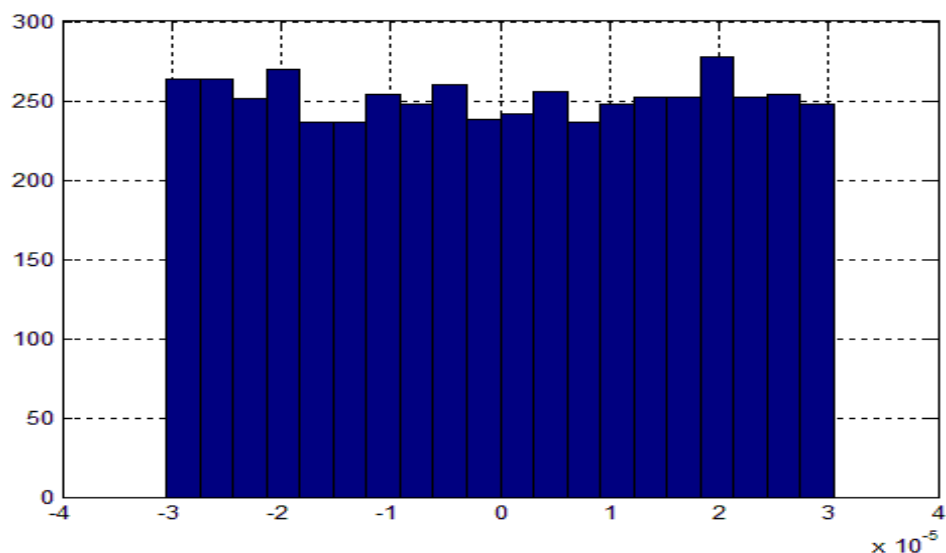


Рис. 3. Гистограмма распределения фазовой ошибки при длине последовательности Задова-Чу при числе элементов последовательности $N=5093$ и числе итераций $N=16$

На рисунке 4 приведена зависимость абсолютной величины ошибки формирования последовательности от числа итераций B и длины последовательности N . Как видно из рисунка, при числе итераций $B=8$ ошибка формирования становится незначительной и слабо зависит от длины последовательности N .

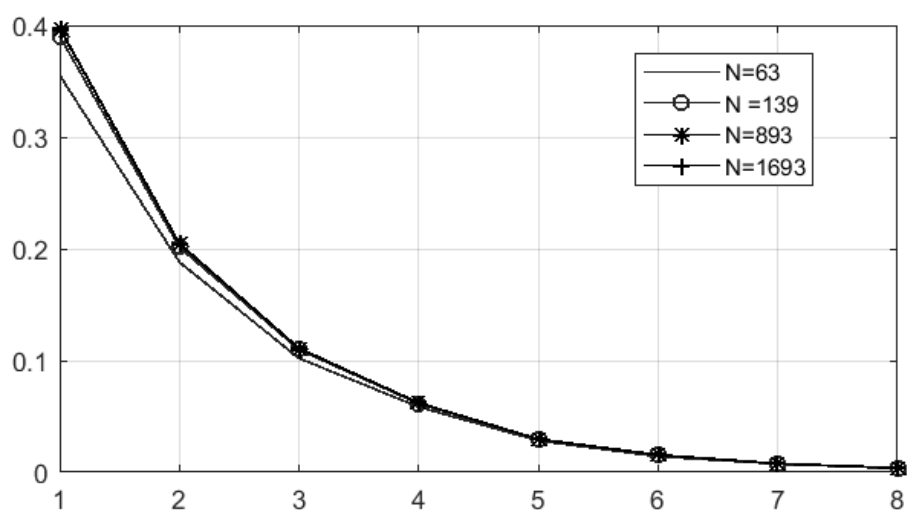


Рис.4. Зависимость абсолютной величины ошибки формирования последовательности от длины последовательности N и числа итераций B

Еще один эксперимент позволил определить уровень боковых лепестков ПАКФ и получить его зависимость от числа итераций B и длины последовательности N . В ходе эксперимента были найдены максимальные значения уровня боковых лепестков для каждой корневой последовательности длины N , затем было проведено усреднение по всем N . Измерения проводились по каждому из приведенных значений ($B=0\dots 8$, $N=63, 139, 8396, 1693$). Результат исследований приведен на рис. 5.

В реальных цифровых устройствах число разрядов конечно и представление данных производится, как правило, в формате с фиксированной точкой. Рассмотрено, как это может оказать влияние на корреляционные свойства последовательности. Была произведена оценка изменения уровня боковых лепестков ПАКФ в зависимости от формата представления чисел. Результат эксперимента представлен на рис.6. На данном рисунке по оси x отложены номера отсчетов цифрового сигнала, по оси y – значение ПАКФ.

Таким образом, следует ожидать, что для случая обнаружения преамбулы функция корреляции является весьма устойчивой к ошибкам квантования в представлении последовательностей Задова-Чу.

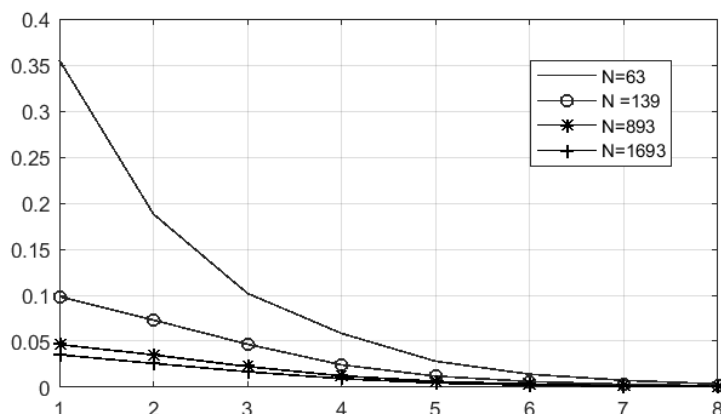


Рис. 5. Зависимость уровня боковых лепестков ПАКФ от длины последовательности N и числа итераций B

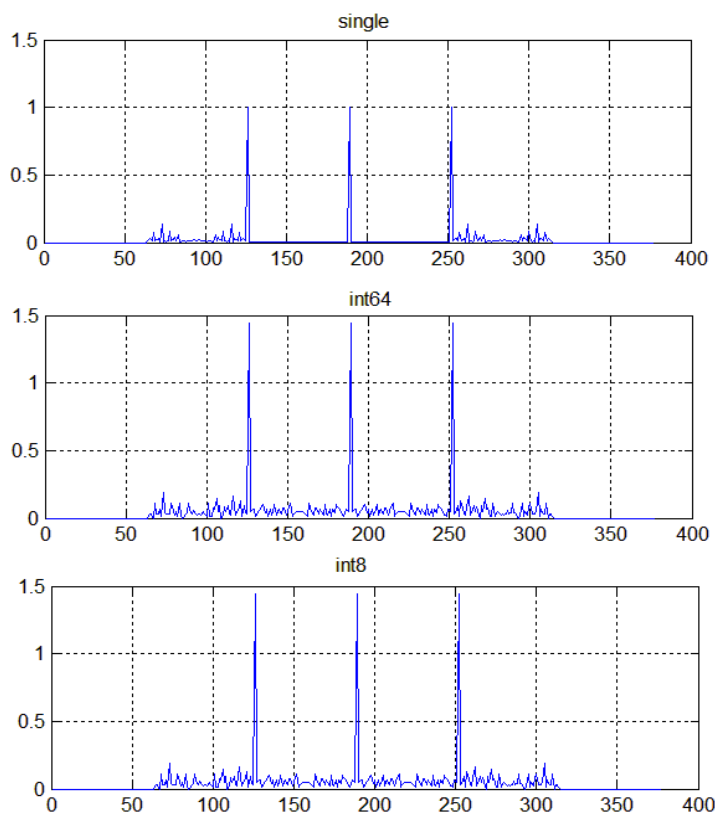


Рис. 6. Периодическая АКФ последовательности Задова-Чу с $N=63$, представленная с одинарной точностью в формате с плавающей запятой (верхний рисунок – single), в формате с фиксированной запятой с разрядностью 64 (средний рисунок – int64), в формате с фиксированной запятой с разрядностью 8 (нижний рисунок – int8)

Заключение

Проведено сравнение элементов последовательности Задова-Чу, вычисленных с помощью рекуррентной формулы, а также с использованием алгоритма CORDIC с различным числом итераций B . Показано, что ошибка формирования элементов последовательности с применением метода CORDIC носит случайный характер с распределением, приближающимся к равномерному. Получена зависимость абсолютной величины ошибки формирования последовательности от числа

итераций B и длины последовательности N . Показано, что при числе итераций $B=8$ ошибка формирования становится незначительной и слабо зависит от длины последовательности N . Получена зависимость максимальной величины боковых лепестков периодической автокорреляционной функции от числа итераций B и длины последовательности N . Показано, что при числе итераций $B=8$ максимальный уровень боковых лепестков становится незначительным и слабо зависит от длины последовательности N . Исследовано влияния ошибок квантования на представление последовательностей Задова-Чу. Показано, что на уровень боковых лепестков периодической автокорреляционной функции сильное влияние оказывает формат представления чисел и слабое разрядность представления чисел. Однако этот вопрос требует дальнейшего изучения.

Литература

1. 3GPP ETSI TS 136211 V10.0.0.(2011-01) / URL: http://www.etsi.org/deliver/etsi_TS/136200_136299/136211/10.00.00_60/ts_136211v100000p.pdf (дата обращения 10.10.2019г.)
2. Peter Gaal. «Efficient Zadoff-Chu Sequence Generation», WO 2011127007 A1, 2011 г.
3. Chu D.C. «Polyphase codes with good periodic correlation properties». IEEE Transactions on Information Theory, July 1972, V. IT-18, №4.
4. Потехин Д.С., Тарасов И.Е. Разработка систем цифровой обработки сигналов на базе ПЛИС. М.: Горячая линия-Телеком, 2007.

INVESTIGATION OF A ZADOV-CHU SEQUENCE GENERATION ACCURACY

Maria V. Shubina

MTUCI, Assistant of the Department of RTS

m.v.shubina@mtuci.ru

Tatyana P. Kosichkina

MTUCI, Ph.D., Associate Professor, Department of RTS

t.p.kosichkina@mtuci.ru

Keywords: *synchronization in LTE networks, multiphase sequences, Zadov-Chu sequence, auto-correlation function, sequences with zero autocorrelation zone, digital signal generation.*

Methods of generating LTE clock signals are considered. As such signals, as a rule, Zadov-Chu sequences belonging to the class of multiphase codes are used. A feature of the structure of these sequences and the related problem of their formation in real time are discussed. This issue is relevant because real-time implementation of the Zadov-Chu sequence is a compromise between memory capacity and computational complexity. A comparison is made of two formation methods: the tabular method and the CORDIC method, the influence of these methods on the synchronization accuracy is investigated, namely, the level of the side lobes of the periodic and aperiodic ACF. Different sequences of this class is estimated under various restrictions.

ХАРАКТЕРИСТИКИ КОВАРИАЦИОННЫХ ФУНКЦИЙ И ОЦЕНКА ПАРАМЕТРОВ ШУМОПОДОБНОГО СИГНАЛА

Гут Татьяна Михайловна
студентка группы МИТ 1901 МТУСИ
gut1551@mail.ru

Горгадзе Светлана Феликсовна
МТУСИ, д.т.н., профессор
svetlana-gorgadze@yandex.ru

Ключевые слова: шумоподобные сигналы, сигналы на негармонических поднесущих, оценка параметров, функция неопределенности, двумерная автоковариационная функция, аддитивная помеховая составляющая, коррелятор с комплексным перемножителем.

Для шумоподобного сложного сигнала с прямым расширением спектра рассмотрены алгоритмы оценки параметров его задержки по времени и частоты с точностью до фазы. Предложено использовать статистику, соответствующую действительной части функции неопределенности (ФН) сигнала при его формировании на негармонической поднесущей. Это позволяет существенно повысить точность оценки параметров такого сигнала, по сравнению с традиционным подходом на основе использования модуля ФН.

Шумоподобные сложные сигналы широко используются в настоящее время в системах связи и радионавигации – прежде всего там, где предъявляются повышенные требования к точности синхронизации их генераторов по частоте и задержке [1,2,5-9]. При этом часто используются дополнительные негармонические поднесущие – меандровые или многопозиционные [3,4]. Использование меандровых поднесущих приводит к расщеплению основного корреляционного пика автоковариационной функции (АКФ) сигнала на несколько пиков, что приводит также и к расщеплению и смещению в область более высоких и более низких частот его энергетического спектра. С целью иллюстрации данного обстоятельства на рис. 1 приводятся основные пики АКФ так называемых ВОС (binary offset carrier modulated)-сигналов [3] при соотношениях между их тактовыми частотами, рассматриваемыми ниже.

Центральный пик нормированной АКФ $\chi(\tau)$ огибающей шумоподобного сигнала, состоящей из элементарных импульсов прямоугольной формы длительностью $T_3 = 1/f_{\text{ПСП}}$, и полярностью, соответствующей закону следования символов применяемой бинарной псевдослучайной последовательности (ПСП), показан на рис.1 черной пунктирной линией, где $f_{\text{ПСП}}$ – тактовая частота ПСП. Кроме того, на этом рисунке черной сплошной линией показана форма основного пика АКФ ВОС(1,1), у которого тактовая частота меандровой поднесущей f_M в два раза выше его $f_{\text{ПСП}}$, так что на длительности элементарного импульса основной ПСП укладывается $f_M / f_{\text{ПСП}} = N_M = 2$ импульса меандра. Штрих-пунктирной линией показана АКФ ВОС (10,5), у которого $N_M = 4$, а тактовая частота ПСП $f_{\text{ПСП}}' = 5f_{\text{ПСП}}$. При выборе соотношений между тактовыми частотами сигналов учитывалось, что у ВОС (10,5), используемого в современных радиосистемах, тактовая частота составляет 5,115МГц, а $f_M = 10,23$ МГц, в то время как у применяемых ВОС (1,1) тактовая частота ПСП в 5 раз ниже, и составляет 1,023МГц. Жирной штрих-пунктирной линией показан основной пик АКФ сигнала на основе бинарной ПСП без поднесущей с тактовой частотой $f_{\text{ПСП}}' = 5f_{\text{ПСП}}$ и с длительностью времени корреляции (шириной основного корреляционного пика) $\tau_K = 2T_3 / 5$, который огибает основной пик ВОС(10,5). При этом у АКФ ВОС (10,5) выделяется центральная область с шириной $\tau_K = 2T_3 / 35$. Очевидно, что для того, чтобы обеспечить такую же ширину основного корреляционного пика у шумоподобного сигнала традиционного типа необходимо, чтобы

его тактовая частота $f'_{\text{ПСП}} = 35f_{\text{ПСП}}$. Это потребует расширения ширины полосы частот, занимаемой сигналом в 35 раз по сравнению с исходным сигналом без меандровой поднесущей. Основным пик его АКФ показан на рис.1 жирной черной линией.

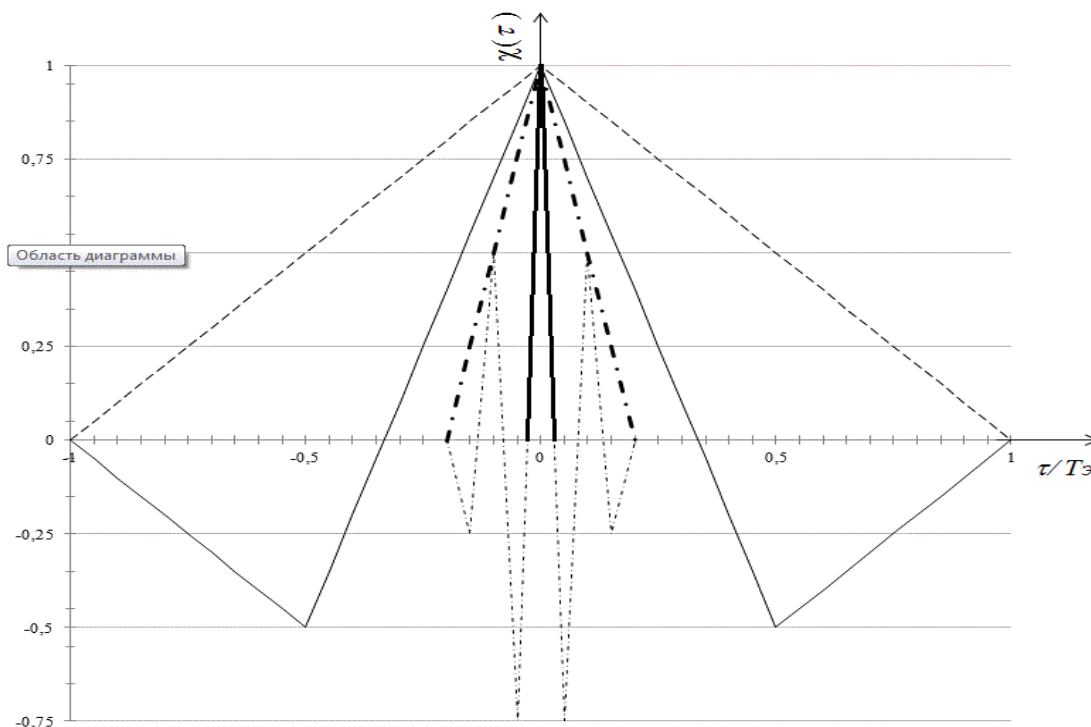


Рис. 1. Формы основных пиков нормированных АКФ ВОС-сигналов

Использование МОС (multi position offset carrier modulated)-сигналов приводит к уменьшению флуктуаций действительной части АКФ сигнала в области основного корреляционного пика. В качестве иллюстрации данного вывода на рис. 2 приведены типичные виды основных пиков МОС-сигналов (линии из точек) и ВОС-сигналов (пунктирные линии) при $N_M = 4$, взятые из [3].

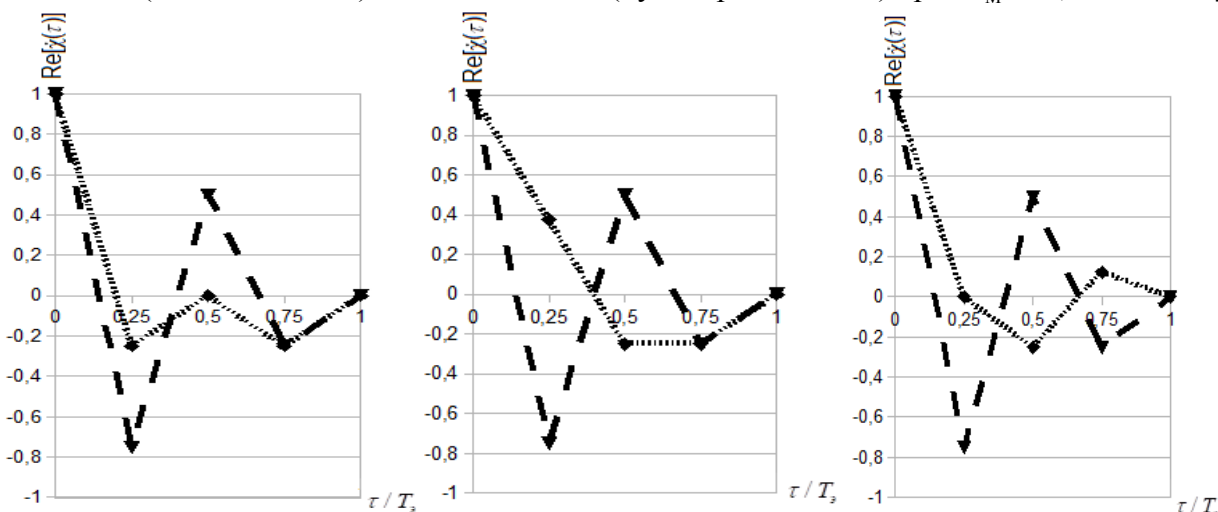


Рис. 2. Типичные виды основных пиков АКФ МОС- и ВОС-сигналов

Обобщая полученные результаты по характеристикам ВОС- и МОС-сигналов, можно сделать вывод и о соответствующем изменении структуры основного пика их функций неопределенности (ФН) $\dot{Z}(\tau, \Delta f)$. Таким образом, основным свойством современных шумоподобных сигналов на негармонических поднесущих является то, что у них в области основного пика ФН

$$\operatorname{Re} \left[\dot{Z}(\tau, \Delta f) \right] \neq \left| \dot{Z}(\tau, \Delta f) \right|.$$

Целью данной работы является разработка алгоритма оценки параметров шумоподобного сложного сигнала с прямым расширением спектра (задержки по времени и частоты с точностью до фазы) на основе действительной части его функции неопределенности.

Основной подход к оценке параметров синхронизации шумоподобного сигнала

Базовый алгоритм оценки медленно меняющихся отклонений задержки по времени τ , несущей частоты Δf и смещения фазы частоты φ сигнала от их известных значений t_0, f_0 и Φ на фоне белого гауссовского шума, соответствующий методу максимального правдоподобия, описывается правилом [1,2,5]:

$$\hat{\tau}, \hat{\Delta f}, \hat{\varphi} = \arg \max \operatorname{Re} \left[\dot{Z}(\tau, \Delta f, \varphi) \right], \operatorname{Re} \left[\dot{Z}(\tau, \Delta f, \varphi) \right] \geq 0, \quad (1)$$

где $\hat{\tau}, \hat{\Delta f}, \hat{\varphi}$ – оценки τ, f и φ , а $\dot{Z}(\tau, \Delta f, \varphi)$ – скалярное произведение комплексных огибающих наблюдения и опорного сигнала, формируемого в приемнике. С учетом компенсации комплексно-сопряженных составляющих с одинаковыми значениями t_0, f_0 и Φ , можно записать:

$$\dot{Z}(\tau, \Delta f, \varphi) \equiv \int_{-T}^T \dot{Y}(t) \dot{S}_{\text{оп}}^*(t - \tau) dt, \quad (2)$$

где $\dot{Y}(t)$ – комплексная огибающая наблюдения, представляющего собой комплексную огибающую аддитивной смеси белого гауссовского шума $\dot{\xi}(t)$ и полезного сигнала $S_c(t) = U(t) \exp(j\varphi_c(t))$ с известными амплитудой $U(t)$ и зависимостью изменения его фазы от времени $\varphi_c(t)$. Огибающая опорного сигнала сдвинута относительно $\dot{S}_c(t)$ по времени на τ и по фазе на $\varphi(t)$, в результате чего $\dot{S}_{\text{оп}}^*(t - \tau) = aU(t - \tau) \exp[-j(\varphi_c(t - \tau) - \varphi(t))]$,

где $*$ – знак комплексного сопряжения, a – размерная константа, $\varphi(t) = 2\pi\Delta f t + \varphi$.

Отбросив аддитивную помеховую составляющую $\dot{Z}(\tau, \Delta f, \varphi)$, легко увидеть, что оставшаяся часть с точностью до постоянного нормирующего множителя совпадает с так называемой функцией неопределенности (ФН) сигнала [5], называемой также его двумерной автоковариационной функцией (ДАКФ) [3]. Таким образом, правило (1), (2) предполагает отыскание максимального значения действительной части ФН сигнала и совместную оценку трех его параметров $\hat{\tau}, \hat{\Delta f}$ и $\hat{\varphi}$. Точность оценки определяется тем, насколько правильно найдено максимальное значение так называемого основного пика ФН при $\tau \leq 1/2\Delta F$ и $\Delta f \leq 1/2\Delta T$, где ΔF – ширина спектра сигнала, а T – его длительность (период). Причем в области основного пика ФН практически всех шумоподобных сигналов, применявшихся до последнего времени, при оценке φ с пренебрежимо малой погрешностью справедливо двойное тождество: $\operatorname{Re} \left[\dot{Z}(\tau, \Delta f) \right] \equiv \left| \dot{Z}(\tau, \Delta f, \varphi) \right| \equiv \left| \dot{Z}(\tau, \Delta f) \right|$, то есть модуль ФН не зависит от φ . Поэтому в (1), (2) $\operatorname{Re} \left[\dot{Z}(\tau, \Delta f, \varphi) \right]$, обычно заменяется на $\left| \dot{Z}(\tau, \Delta f, \varphi) \right|$, параметр φ считается неинформативным и не оценивается.

Таким образом, вариант алгоритма (1), (2) со статистикой $\operatorname{Re} \left[\dot{Z}(\tau, \Delta f, \varphi) \right]$, требует оценки φ . При этом аддитивная шумовая помеха описывается гауссовским законом. При замене

$\text{Re} \left[\dot{Z}(\tau, \Delta f, \varphi) \right]$ на $\left| \dot{Z}(\tau, \Delta f) \right|$ помеховая составляющая имеет обобщенное χ^2 – распределение, чем можно было бы объяснить выигрыш первого варианта рассматриваемого алгоритма, по сравнению со вторым по точности оценок τ и Δf . Но в действительности выигрыш практически отсутствует из-за увеличения размерности области определения функции $\dot{Z}(\tau, \Delta f, \varphi)$, по сравнению с $\dot{Z}(\tau, \Delta f)$ и влияния точности $\hat{\varphi}$ на точность $\hat{\tau}$ и $\hat{\Delta f}$.

Как показывает сравнение форм сечений $\text{Re} \left[\dot{Z}(\tau, \Delta f) \right]$ и $\left| \dot{Z}(\tau, \Delta f) \right|$ по оси времени, соответствующих АКФ, проведенное в первой части данной работы, совместная оценка $\hat{\tau}$, $\hat{\Delta f}$ и $\hat{\varphi}$ по алгоритму (1) позволит повысить точность оценки $\hat{\tau}$, $\hat{\Delta f}$, по сравнению с правилом, когда $\text{Re} \left[\dot{Z}(\tau, \Delta f, \varphi) \right]$ заменяется на $\left| \dot{Z}(\tau, \Delta f) \right|$. Этим может быть объяснен выигрыш в точности вышеуказанных оценок, проводимых с использованием измерительных ВОС- и МОС-сигналов, по сравнению с традиционными ШПС.

Алгоритм оценки параметров шумоподобного сигнала

Таким образом, сигнальная составляющая (2) имеет вид:

$$\dot{Z}(\tau, \Delta f, \varphi) = \int_{-T}^T \dot{S}_C(t) \dot{S}_{\text{оп}}^*(t - \tau) dt. \quad (3)$$

Вводя обозначения $\dot{S}_{\text{с.оп}}(t) = \dot{S}_C(t)$, $\dot{S}_{\text{оп}}(t) = \dot{S}_{\text{с.оп}}(t) \exp[j(2\pi\Delta f t - \varphi)]$, запишем:

$$\begin{aligned} \dot{Z}(\tau, \Delta f, \varphi) &= \text{Re}[\dot{Z}(\tau, \Delta f, \varphi) + j \text{Im}[\dot{Z}(\tau, \Delta f, \varphi)]] = \\ &= \text{Re} \left[\int_{-T}^T \dot{S}_C(t) S_{\text{с.оп}}^*(t - \tau) \exp[-j(2\pi\Delta f t + \varphi)] dt \right] \\ &+ j \text{Im} \left[\int_{-T}^T \dot{S}_C(t) S_{\text{с.оп}}^*(t - \tau) \exp[-j(2\pi\Delta f t + \varphi)] dt \right], \end{aligned} \quad (4)$$

$$\begin{aligned} \left| \dot{Z}(\tau, \Delta f) \right| &= \left\{ \left(\text{Re} \left[\int_{-T}^T \dot{S}_C(t) S_{\text{с.оп}}^*(t - \tau) \exp[-j(2\pi\Delta f t + \varphi)] dt \right] \right)^2 + \right. \\ &\left. \left(\text{Im} \left[\int_{-T}^T \dot{S}_C(t) S_{\text{с.оп}}^*(t - \tau) \exp[-j(2\pi\Delta f t + \varphi)] dt \right] \right)^2 \right\}^{1/2}. \end{aligned} \quad (5)$$

Как видно, в (4) и (5) входит комплексная огибающая принимаемого сигнала, которую необходимо выделить из действительного сигнала на входе приемника. Последний при принятой системе отсчета времени, значений несущей частоты и фазового сдвига частоты имеет вид: $S_C(t) = U(t) \cos(2\pi f_0 t + \varphi_C(t))$. То есть для упрощения выкладок принято, что у сигнала на входе приемника отсутствуют временной, частотный и фазовый сдвиги, а опорный сигнал имеет соответствующие сдвиги, которые неизвестны. В действительности в приемнике формируются опорный сигнал с известными значениями этих параметров, а затем формируется функция неизвестных значений разностей этих параметров у опорного и принимаемого сигналов.

С целью выделения комплексной огибающей принимаемого сигнала рассмотрим возможность использования одного из известных алгоритмов. Он показан на рис. 3 в части схемы между точками 0 и 2, 2' и отличается от известного алгоритма тем, что опорные гармонические сигналы, подаваемые на множители, имеют неизвестные фазовые сдвиги $\varphi(t)$.

В точке 1 схемы получим:

$$\begin{aligned}
S_C(t) * 2 \cos(2\pi f_0 t + \varphi(t)) &= 2U(t) \cos(2\pi f_0 t + \varphi_C(t)) \cos(2\pi f_0 t + \varphi(t)) = \\
&= 2U(t) \left[(\cos 2\pi f_0 t)^2 \cos \varphi_C(t) \cos \varphi(t) + (\sin 2\pi f_0 t)^2 \sin \varphi_C(t) \sin \varphi(t) \right] - \\
U(t) \sin(4\pi f_0 t) \sin(\varphi_C(t) + \varphi(t)) &= \\
2U(t) \left[(\cos 2\pi f_0 t)^2 \cos(\varphi_C(t) + \varphi(t)) + \sin \varphi_C(t) \sin \varphi(t) \right] - U(t) \sin(4\pi f_0 t) \sin(\varphi_C(t) + \varphi(t)) &= \\
U(t) \left[(1 + \cos(4\pi f_0 t)) * \cos(\varphi_C(t) + \varphi(t)) + 2 \sin \varphi_C(t) \sin \varphi(t) \right] - U(t) \sin(4\pi f_0 t) \sin(\varphi_C(t) + \varphi(t)) &= \\
U(t) \cos(\varphi_C(t) - \varphi(t)) + U(t) (\cos(4\pi f_0 t + (\varphi_C(t) \varphi(t))). &
\end{aligned}$$

В данном случае $\varphi(t)$ – неизвестный полный фазовый сдвиг опорного сигнала относительно принимаемого.

Отбрасывая высокочастотный компонент, не пропускаемый ФНЧ, получим в точке 2 схемы:

$$\begin{aligned}
U(t) \cos(\varphi_C(t) - \varphi(t)) &= U(t) \cos \varphi_C(t) \cos \varphi(t) + U(t) \sin \varphi_C(t) \sin \varphi(t) = \\
\operatorname{Re} \left[\dot{S}_C(t) \right] \cos \varphi(t) + \operatorname{Im} \left[\dot{S}_C(t) \right] \sin \varphi(t) &= \operatorname{Re} \left[\dot{S}_C(t) \exp(-j2\pi\varphi(t)) \right].
\end{aligned}$$

Аналогично в точке 2' схемы:

$$\begin{aligned}
U(t) \sin(\varphi_C(t) - \varphi(t)) &= U(t) \sin \varphi_C(t) \cos \varphi(t) - U(t) \cos \varphi_C(t) \sin \varphi(t) = \\
\operatorname{Im} \left[\dot{S}_C(t) \right] \cos \varphi(t) + \operatorname{Re} \left[\dot{S}_C(t) \right] \sin \varphi(t) &= \operatorname{Im} \left[\dot{S}_C(t) \exp(-j2\pi\varphi(t)) \right].
\end{aligned}$$

Между точками 2, 2' и 3, 3' схемы реализован комплексный перемножитель (КП) огибающих принимаемого и опорного сигналов. Опорный сигнал является комплексно-сопряженной копией принимаемого сигнала и сдвинут по фазе относительно него на неизвестное значение $\varphi(t)$.

В точке 3' схемы получаем:

$$\begin{aligned}
&(\operatorname{Re} \left[\dot{S}_C(t) \right] \operatorname{Re} \left[\dot{S}_{C.оп}(t - \tau) \right] + \operatorname{Im} \left[\dot{S}_C(t) \right] \operatorname{Im} \left[\dot{S}_{C.оп}(t - \tau) \right]) \cos \varphi(t) - \\
&(\operatorname{Re} \left[\dot{S}_C(t) \right] \operatorname{Im} \left[\dot{S}_{C.оп}(t - \tau) \right] - \operatorname{Im} \left[\dot{S}_C(t) \right] \operatorname{Re} \left[\dot{S}_{C.оп}(t - \tau) \right]) \sin \varphi(t) = \\
&\operatorname{Re} \left[\dot{S}_C(t) \dot{S}_{C.оп}(t - \tau) \right] \cos \varphi(t) + \operatorname{Im} \left[\dot{S}_C(t) \dot{S}_{C.оп}(t - \tau) \right] \sin \varphi(t),
\end{aligned}$$

а в точке 3' соответственно

$$\begin{aligned}
&-(\operatorname{Re} \left[\dot{S}_C(t) \right] \operatorname{Re} \left[\dot{S}_{C.оп}(t - \tau) \right] + \operatorname{Im} \left[\dot{S}_C(t) \right] \operatorname{Im} \left[\dot{S}_{C.оп}(t - \tau) \right]) \sin \varphi(t) - \\
&(\operatorname{Re} \left[\dot{S}_C(t) \right] \operatorname{Im} \left[\dot{S}_{C.оп}(t - \tau) \right] - \operatorname{Im} \left[\dot{S}_C(t) \right] \operatorname{Re} \left[\dot{S}_{C.оп}(t - \tau) \right]) \cos \varphi(t) = \\
&-\operatorname{Re} \left[\dot{S}_C(t) \dot{S}_{C.оп}(t - \tau) \right] \sin \varphi(t) + \operatorname{Im} \left[\dot{S}_C(t) \dot{S}_{C.оп}(t - \tau) \right] \cos \varphi(t).
\end{aligned}$$

В точке 4

$$\begin{aligned}
&\cos \varphi \left\{ \int_{-T}^T \operatorname{Re} \left[\dot{S}_C(t) \dot{S}_{C.оп}(t - \tau) \right] \cos(2\pi\Delta f t) dt + \int_{-T}^T \operatorname{Im} \left[\dot{S}_C(t) \dot{S}_{C.оп}(t - \tau) \right] \sin(2\pi\Delta f t) dt \right\} - \\
&\sin \varphi \left\{ \int_{-T}^T \operatorname{Re} \left[\dot{S}_C(t) \dot{S}_{C.оп}(t - \tau) \right] \sin(2\pi\Delta f t) dt - \int_{-T}^T \operatorname{Im} \left[\dot{S}_C(t) \dot{S}_{C.оп}(t - \tau) \right] \cos(2\pi\Delta f t) dt \right\} = \\
&\cos \varphi \operatorname{Re} \left[\int_{-T}^T \dot{S}_C(t) \dot{S}_{C.оп}(t - \tau) \exp(-2\pi\Delta f t) dt \right] + \sin \varphi \operatorname{Im} \left[\int_{-T}^T \dot{S}_C(t) \dot{S}_{C.оп}(t - \tau) \exp(-2\pi\Delta f t) dt \right] = \\
&\operatorname{Re} \left[\exp(-j\varphi) \int_{-T}^T \dot{S}_C(t) \dot{S}_{C.оп}(t - \tau) \exp[-j(-2\pi\Delta f t)] dt \right] = \operatorname{Re} \left[\dot{Z}(\tau, \Delta f, \varphi) \right],
\end{aligned}$$

а в точке 4'

$$\begin{aligned}
& -\sin \varphi \left\{ \int_{-T}^T \operatorname{Re} \left[\dot{S}_C(t) \dot{S}_{C.оп} (t-\tau) \right] \cos(2\pi\Delta f t) dt + \int_{-T}^T \operatorname{Im} \left[\dot{S}_C(t) \dot{S}_{C.оп} (t-\tau) \right] \sin(2\pi\Delta f t) dt \right\} + \\
& \cos \varphi \left\{ \int_{-T}^T \operatorname{Re} \left[\dot{S}_C(t) \dot{S}_{C.оп} (t-\tau) \right] \sin(2\pi\Delta f t) dt + \int_{-T}^T \operatorname{Im} \left[\dot{S}_C(t) \dot{S}_{C.оп} (t-\tau) \right] \cos(2\pi\Delta f t) dt \right\} = \\
& \cos \varphi \operatorname{Im} \left[\int_{-T}^T \dot{S}_C(t) \dot{S}_{C.оп} (t-\tau) \exp(-2\pi\Delta f t) dt \right] - \sin \varphi \operatorname{Re} \left[\int_{-T}^T \dot{S}_C(t) \dot{S}_{C.оп} (t-\tau) \exp(-2\pi\Delta f t) dt \right] = \\
& \operatorname{Im} \left[\exp(-j\varphi) \int_{-T}^T \dot{S}_C(t) \dot{S}_{C.оп} (t-\tau) \exp[-j(-2\pi\Delta f t)] dt \right] = \operatorname{Im} \left[\dot{Z}(\tau, \Delta f, \varphi) \right].
\end{aligned}$$

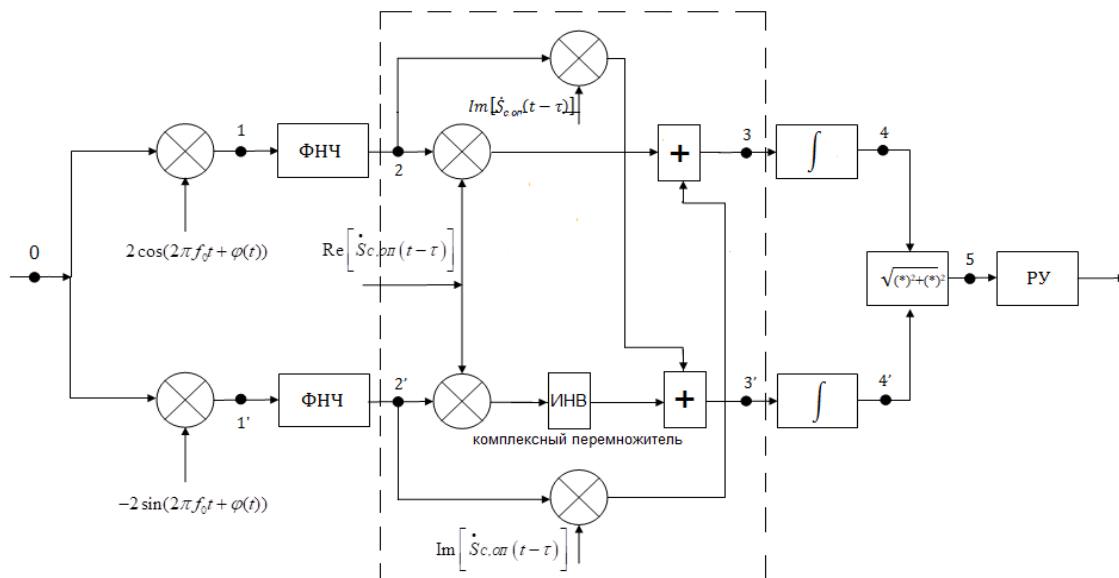


Рис. 3. Структурная схема, иллюстрирующая алгоритм измерения задержки по времени и частоты многопозиционного сигнала, ИНВ – инвертор

Тогда в точке 5 схемы будет:

$$\left\{ \left(\operatorname{Re} \left[\int_{-T}^T \dot{S}_C(t) S_{C.оп}^*(t-\tau) \exp(-j2\pi\Delta f t) dt \right] \right)^2 + \left(\operatorname{Im} \left[\int_{-T}^T \dot{S}_C(t) S_{C.оп}^*(t-\tau) \exp(-j2\pi\Delta f t) dt \right] \right)^2 \right\}^{1/2} = \left| \dot{Z}(\tau, \Delta f) \right|.$$

Последнее выражение не зависит от φ .

Заключение

Таким образом, рис. 3 иллюстрирует алгоритм оценки параметров синхронизации шумоподобного сигнала – задержки по времени τ , несущей частоты Δf и смещения фазы частоты φ от их известных значений t_0 , f_0 и Φ на фоне белого гауссовского шума, соответствующий методу максимального правдоподобия. Для оценки этих параметров необходимо путем варьирования частоты и фазы гармонических сигналов на входах перемножителей, находящихся перед точками 1 и 1' схемы и задержки τ действительной и мнимой частей опорных сигналов, подаваемых на комплексный перемножитель, найти максимальное значение напряжения в точке 4 схемы, в которой получаем действительную часть ФН. Как известно, мнимая часть ФН шумоподобного сигнала не имеет ярко выраженного основного пика и не важна для оценки параметров сигнала. Ее значение имеет существенное значение только для случая, когда не оценивается φ и определяется максимальное значение модуля функции неопределенности, которую можно получить в точке 5 схемы.

Разработанный алгоритм оценки параметров синхронизации представляет собой последовательный когерентный коррелятор, включающий комплексный перемножитель опорного и прини-

маемого шумоподобных сигналов.

Литература

- 1) *Shulze H., Luders C.* Theory and Applications of OFDM and CDMA. Chichester, UK.: John Wiley & Sons, 2005. 421 p.
- 2) *Fazel K., Kaizer S.* Multicarrier and Spread spectrum Systems. From OFDM to MC CDMA to LTE and WIMAX. Chichester, UK.: John Wiley & Sons, 2008. 360 p.
- 3) *Горгадзе С.Ф., Бойков В.В.* Измерительные сигналы с многопозиционными поднесущими для спутниковых радионавигационных систем // Радиотехника и электроника. 2014. №3. С. 264-278.
- 4) *Горгадзе С.Ф.* Сложные композитные сигналы с равномерной амплитудной огибающей // Радиотехника и электроника. 2017. №4. С. 1-15.
- 5) *Iratov V.P.* Spread Spectrum and CDMA. Principles and Applications. John Wiley & Sons, 2006. 488 p.
- 6) *Косичкина Т.П., Сперанский В.С., Спиринов А.П., Фролов А.А.* Когнитивные сверхширокополосные радиосистемы как метод повышения эффективности использования радиочастотного спектра // Т-Comm: Телекоммуникации и транспорт. 2015. Т. 9. № 12. С. 37-43.
- 7) *Сперанский В.С., Спиринов 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.
- 8) *Горгадзе С.Ф.* Ускоренный цифровой алгоритм синхронизации шумоподобных сигналов по времени и частоте // Системы синхронизации, формирования и обработки сигналов. 2016. Т. 7. № 4. С. 16-18.
- 9) *Горгадзе С.Ф., Дроздова Л.А., Максимова Д.Ю.* Сравнительный анализ и выбор измерительных сигналов для высокоэффективной спутниковой радионавигационной системы // Т-Comm: Телекоммуникации и транспорт. 2013. Т. 7. № 10. С. 44-47.

CHARACTERISTICS OF COVARIANCE FUNCTIONS AND ESTIMATION OF NOISE-LIKE SIGNAL PARAMETERS

Tatiana M. Gut

Student of the MIT 1901 MTUCI group

gut1551@mail.ru

Svetlana F. Gorgadze

Ph.D., professor MTUCI

svetlana-gorgadze@yandex.ru

Keywords: *noise-like signals, signals on non-harmonic subcarriers, parameter estimation, uncertainty function, two-dimensional autocovariance function, additive noise component, correlator with a complex multiplier.*

For a noise-like complex signal with direct spectrum spreading, algorithms for estimating the parameters of its delay in time and frequency up to phase are considered. It is proposed to use statistics corresponding to the real part of the uncertainty function (UF) of a signal during its formation on a non-harmonic subcarrier. This allows one to significantly increase the accuracy of estimating the parameters of such a signal, compared with the traditional approach based on the use of the UF module.

ЭКСТРАПОЛЯЦИЯ НЕСТАЦИОНАРНЫХ ЗАШУМЛЕННЫХ СИГНАЛОВ С ПОМОЩЬЮ ФРЕЙМОВ ВЕЙЛЯ-ГЕЙЗЕНБЕРГА

Мирошниченко Антон Валерьевич

Студент группы М61801 МТУСИ

Mirosh.A.V@yandex.ru

Волчков Валерий Павлович

МТУСИ, д.т.н., профессор кафедры ОТС

volchkovvalery@mail.ru

Ключевые слова: фрейм Вейля-Гейзенберга, WH-фрейм, экстраполяция, авторегрессия, нестационарный сигнал

Для решения задачи экстраполяции нестационарных зашумленных сигналов предлагается использовать жесткие фреймы Вейля-Гейзенберга. Строится алгоритм экстраполяции таких сигналов с использованием прямого и обратного фрейм-преобразований и приводится его экспериментальное сравнение с классическим алгоритмом экстраполяции на основе авторегрессионной модели. Показано, что фрейм-экстраполяция позволяет получить гораздо меньшие экстраполяционные ошибки.

Введение

Современный этап развития техники наполнен различными датчиками, это могут быть датчики «умных домов», медицинская техника или радиолокаторы, которые формируют сигнал и передают его на устройство обработки сигналов. Сигнал, полученный с датчиков, должен быть обработан и проанализирован. Классический алгоритм анализа сигналов – частотный, основанный на его разложении в ряд Фурье. Дискретный его вариант – дискретное преобразование Фурье зарекомендовал себя как универсальный, однако при анализе сложных нестационарных сигналов он уступает некоторым другим алгоритмам. Так в работах [1,2, 6-9] приводится сравнение алгоритмов спектрального анализа, экстраполяции и интерполяции построенных на основе дискретных фреймов Вейля-Гейзенберга (WH-фреймов) и дискретного преобразования Фурье (ДПФ), где показано преимущество первых. Это связано прежде всего с хорошей частотно-временной локализацией фреймовых функций, что позволяет более адекватно аппроксимировать реализации случайных нестационарных сигналов в виде импульсов или функций, содержащих резкие перепады.

С другой стороны, известно, что для многих практических задач, связанных с экстраполяцией, т.е. предсказанием сигнала за пределами известного диапазона, широко применяются авторегрессионные модели [3,4]. Это связано с тем, что очень часто механизм формирования сигнала случаен, а его реализации плохо аппроксимируются степенными полиномами. Авторегрессионная случайная модель позволяет генерировать реализации более сложного вида, и в то же время ее параметры могут быть достаточно просто рассчитаны на основе автокорреляционной функции сигнала или оценены по выборочным отсчетам сигнала. Однако, данная модель может адекватно представлять только стационарные «в широком смысле» случайные процессы, у которых математическое ожидание и дисперсия не зависят от времени. Кроме того, известно, что авторегрессионная модель плохо аппроксимирует зашумленные сигналы.

В то же время, многие наблюдаемые процессы, например, радиолокационные или акустические эхо сигналы как правило зашумлены, не обладают стационарностью, или интервал их стационарности весьма ограничен. Это может приводить к значительным ошибкам и сбоям в алгоритмах экстраполяции, построенным на основе авторегрессионной модели.

В данной работе предлагается новый алгоритм экстраполяции таких сигналов, основанный на разложениях в частотно-временной фрейм Вейля-Гейзенберга и приводится его сравнение с известным авторегрессионным алгоритмом экстраполяции. Решение задачи экстраполяции рассмотрено для двух случаев: *граничного* – когда неизвестный отрезок сигнала находится в конце заданного интервала времени и *центрального* – когда неизвестные значения сигнала находятся в центре

заданного интервала. Показано, что в обоих случаях применение фреймовой экстраполяции к реализациям нестационарного сигнала приводит к значительно меньшим ошибкам экстраполяции, по сравнению с авторегрессионным алгоритмом.

Описание процедуры построения WH-фрейма

Дискретный фрейм Вейля-Гейзенберга (WH-фрейм) в данной работе определяется как некоторая система функций, состоящая из сдвинутых по времени и частоте импульсов, причем все они могут быть получены из одного формирующего импульса, обладающего специальными свойствами. Эти свойства должны обеспечивать, во-первых, возможность однозначного разложения любого дискретного сигнала по данной системе функций и его обратного восстановления и, во-вторых, удовлетворять заданному критерию частотно-временной локализации. Последнее позволяет говорить об оптимальности WH-фрейма в смысле этого критерия. Для лучшего понимания проблемы, приведем ниже основные этапы построения оптимального WH-фрейма, описанные в работе [2], с учетом специфики решаемой в статье задачи.

Так как обрабатываемые с помощью фрейма дискретные сигналы заданы на конечном дискретном интервале времени $J_N = \{0, 1, \dots, N-1\}$, то и функции, входящие в WH-фрейм должны быть такими же, а структура их частотно-временных сдвигов должна быть согласована с конечным интервалом J_N .

1. Зададим формирующий фреймовый импульс в виде дискретного гауссиана:

$$g_0[t] = (2\sigma_g)^{1/4} \exp(-\pi\sigma(\Delta t)^2), \quad t \in J_N, \quad (1)$$

где σ_g - ширина эталонного гауссиана, определяющая его частотно-временные свойства, $\Delta = 1/f_d$ - интервал дискретизации.

2. Проведем циклическую модификацию функции (1) на конечном дискретном интервале J_N

$$g_c[t] = \begin{cases} g_0[t], & \text{если } t \in \{0, \dots, N/2\} \\ g_0[N-1-t], & \text{если } t \in \{N/2+1, \dots, N-1\} \end{cases} \quad (2)$$

3. Определим исходные параметры, необходимые для построения WH-фрейма.

$P \geq 2$ - коэффициент перенасыщения отсчетами, четное натуральное число;

$M > P$ - количество базовых частотных сдвигов фрейма, натуральное число, кратное P ;

$L \geq 2$ - количество базовых временных сдвигов фрейма;

$N = LM$ - число отчетов фреймовой функции;

$L_o = M / P$ - временной шаг, на который сдвигаются функции фрейма.

$K = N / L_o$ - общее количество временных сдвигов фрейма.

Сформируем на основе (2) эталонную систему функций $\{G_{k,l}[n]\}$ со структурой частотно-временных (ЧВ) сдвигов WH-фрейма в виде матрицы \mathbf{G} размером $N \times NP$, так чтобы по ее столбцам находились $N_o = NP$, сдвинутых во времени и по частоте импульсов.

$$\mathbf{G} = \{G_{k,l}[t]\} = \{g_c(t - lL_o) \exp(j2\pi kt / M)\} \in \mathbb{C}^{N \times NP}. \quad (3)$$

где $t \in J_N = \{0, \dots, N-1\}$, $k \in J_M$, $l \in J_K$, причем $MK = NP$.

Рассмотрим дискретный вещественный сигнал $s[t] = s(\Delta \cdot t)$, $t \in J_N = \{0, 1, \dots, N-1\}$, полученный из непрерывного сигнала $s(\mu)$, $\mu \in [0, T]$ путем его дискретизации с частотой $f_d = 1/\Delta$ на конечном интервале $[0, T]$ и содержащий N отсчетов. Образует из него сигнальный вектор $\mathbf{s} = (s[0], \dots, s[N-1])^T \in \mathbb{R}^N$.

Определение 1. Система функций $\mathbf{U} = \{U_{k,l}[n]\} \in \mathbb{C}^{N \times NP}$ со структурой ЧВ-сдвигов эталона

(3) называется жестким WH-фреймом, если для любого сигнального вектора $\mathbf{s} = (\hat{s}[0], \dots, \hat{s}[N-1])^T$ выполняется неравенство

$$A \|\mathbf{s}\|^2 \leq \|\mathbf{U}^* \mathbf{s}\|^2 \leq A \|\mathbf{s}\|^2, \quad 0 < A < \infty,$$

где $(\cdot)^*$ – оператор эрмитового сопряжения матриц, $\|\cdot\|$ – евклидова норма.

Прямоугольная матрица $\mathbf{U}^* \in \mathbb{C}^{P \cdot N \times N}$ описывает фреймовый оператор, а его действие на сигнал $\mathbf{s} \in \mathbb{C}^N$ приводит к вектору $\mathbf{x} = \mathbf{U}^* \mathbf{s}$, который называется *фреймовым частотно-временным спектром*. При этом сигнал может быть восстановлен из фреймового спектра \mathbf{x} по формуле $\mathbf{s} = \mathbf{U} \mathbf{x}$, если $\mathbf{B} = \mathbf{U} \mathbf{U}^* = \mathbf{I}_N$ – единичная матрица, что соответствует $A = 1$.

Определение 2. WH-фрейм будем называть оптимальным, если он жесткий, а его матрица $\mathbf{U}_{opt} \in \mathbb{C}^{N \times NP}$ состоит из ортонормированных строк и наилучшим образом приближает заданный эталон \mathbf{G} по критерию минимума среднеквадратической ошибки

$$\mathbf{U}_{opt} : \min_{\mathbf{U} \in \mathbf{A}} \|\mathbf{G} - \mathbf{U}\|^2, \quad \mathbf{A} = \{\mathbf{U} \in \mathbb{C}^{N \times PN} : \mathbf{U}^* \mathbf{U} = \mathbf{I}_N\}$$

Как показано в работе [2], решение экстремальной задачи (2) имеет следующий вид

$$\mathbf{U}_{opt} = \mathbf{P}^T \mathbf{F} \mathbf{\Sigma}^{-1} \mathbf{F}^* \mathbf{P} \mathbf{G} \in \mathbb{C}^{N \times NP}. \quad (4)$$

Здесь $\mathbf{P} = (P_{i,j})$ – квадратная $N \times N$ -матрица перестановки, с элементами

$$\mathbf{P}_{i,j} = \begin{cases} 1, & j = L_0(i \bmod(K)) + \lfloor i / K \rfloor \\ 0, & \text{иначе} \end{cases} \quad i, j \in J_N.$$

где $\lfloor \cdot \rfloor$ – оператор округления вниз. $\mathbf{\Lambda}$ – Диагональная матрица

$$\mathbf{\Sigma}_{i,j} = \begin{cases} \sqrt{\lambda_i}, & i = j \\ 0, & \text{иначе} \end{cases} \quad i, j \in J_N,$$

где λ_i – собственные числа матрицы $\mathbf{B} = \mathbf{G} \mathbf{G}^*$. \mathbf{F} – $(N \times N)$ -матрица, состоящая из расположенных на диагонали матриц Фурье \mathbf{F}_0 , размером $K \times K$:

$$\mathbf{F} = \mathbf{I}_{L_0} \otimes \mathbf{F}_0, \quad \mathbf{F}_0 = K^{-1/2} (\exp(j2\pi pq / K))_{p,q \in J_K}, \quad K = PL = N / L_0$$

(\otimes – оператор прямого произведения матриц).

Постановка задачи экстраполяции сигнала

Пусть на входе канала действует сигнал $s_o[t], t \in J_N$ состоящий из N дискретных отсчетов. Предполагается, что при прохождении сигнала по каналу он, во-первых, искажается аддитивным белым гауссовским шумом $\xi[t]$

$$s[t] = s_o[t] + \xi[t], \quad t \in J_N, \quad (5)$$

во-вторых, из-за особенностей канала (частичной наблюдаемости) n отсчетов утрачивается, известны номера утраченных отсчетов и необходимо их потом как-то восстановить, используя процедуру экстраполяции. Сигнал на после утраты части отсчетов на исходной временной сетке и содержащий пустые области обозначим $s'[t], t \in J_N$.

Далее сигнал на выходе канала записывается и подается на вход системы обработки. Для экономии памяти записываются подряд только $N' = N - n$ сигнальных отсчетов, где N' – количество наблюдаемых отсчетов, n – число утраченных отсчетов, в результате чего получается редуцированный дискретный сигнал

$$s'[t'], t' \in J_{N'} = \{0, \dots, N'-1\} \quad (6)$$

на редуцированной временной сетке $J_{N'}$ с меньшим числом $N' < N$ отсчетов.

Задача экстраполяции, которую мы рассматриваем, состоит в наилучшем восстановлении зашумленного сигнала $s[t], t \in J_N$ (5) по его редуцированной версии $s'[\cdot]$ на исходной или редуцированной сетке. Полученный в результате этого экстраполированный сигнал будем обозначать $\tilde{s}[t], t \in J_N$. Кроме того, далее мы задачу экстраполяции рассматриваем для двух случаев: когда утраченные отсчеты сигнала находятся в центре или в конце исходного дискретного временного интервала J_N .

Конкретизируем все компоненты и параметры сигналов (5), (6), чтобы решение задачи параллельно вести с экспериментальными расчетами.

Сформируем сигнал $s_0[t]$, на входе канала из 80 дискретных отсчетов

$$s_0[t] = 0.005t^3 e^{-0.15t}, \quad t \in \{1, \dots, 80\}. \quad (7)$$

Добавим согласно (5) к сигналу (7) аддитивный белый гауссовский шум $\xi[t]$.

$$s[t] = s_0[t] + \xi[t], \quad t \in \{1, \dots, 80\}, \quad (8)$$

с дисперсией $D_\xi = \sigma_\xi^2 = M[\xi^2[t]] = 0,2$.

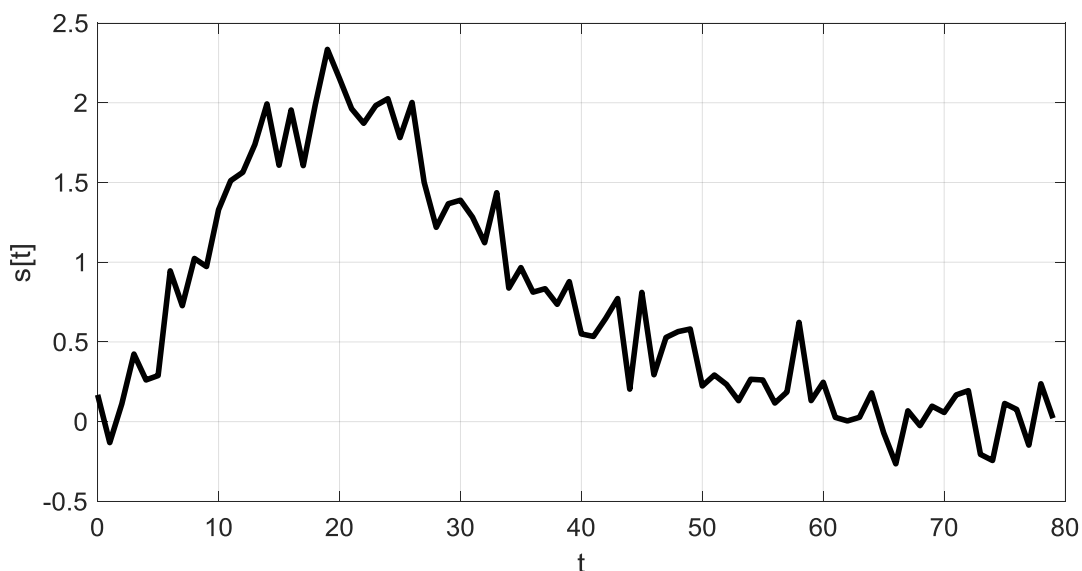


Рис. 1. Оциллограмма зашумленного сигнала $s[t]$

Сформируем два исходных сигнала $s'_1[t]$ и $s'_2[t]$ размерностью 64 временных отсчета, путем удаления из сигнала $s[t]$ 16 дискретных отсчетов. Для сигнала $s'_1[t]$ удалены отсчеты с 65 по 80, а для сигнала $s'_2[t]$ с 33 по 48. На рисунке 2 приведены оциллограммы сигналов редуцированных $s'_1[t], s'_2[t]$ на исходной временной сетке J_{80} .

Задача экстраполяции сигнала может быть сформирована следующим образом: зная соответствующие редуцированные сигналы $s'_1[t']$ и $s'_2[t']$, а также номера удаленных отсчетов, наилучшим образом экстраполировать сигнал $s[t]$, вычислив его экстраполированную оценку $\tilde{s}[t]$ тем или иным методом. В статье исследуются два метода экстраполяции – один основан на применении WN-фреймов, другой на использовании классической модели авторегрессии. Разбиение на два случая – с вырезами «в конце» и «в центре», позволит провести исследование качества экстраполяции для двух разных практических ситуаций.

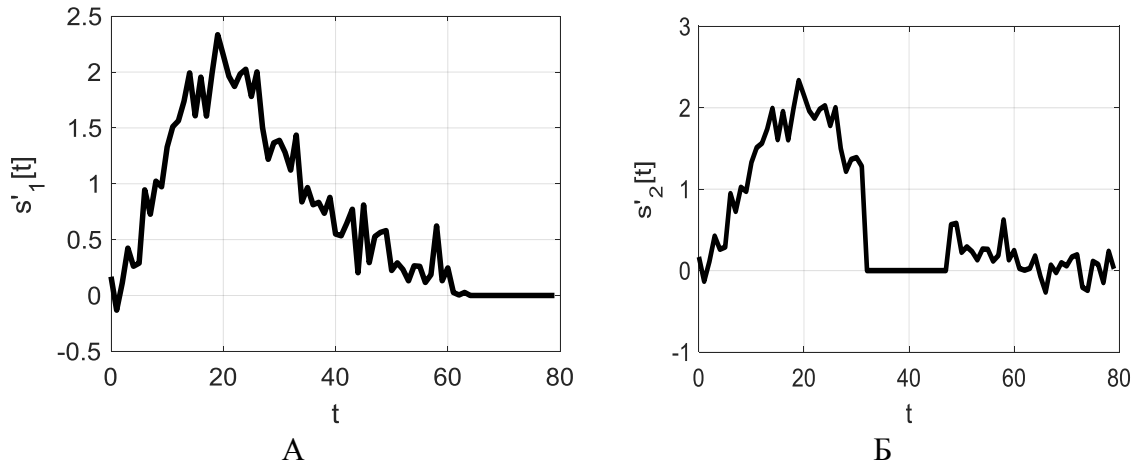


Рис. 2. Осциллограммы сигналов: А - $s'_1[t]$, сигнал с удалением отсчетов в конце,
Б - $s'_2[t]$, сигнал с удалением отсчетов в середине

Решение задачи экстраполяции с помощью WH-фрейма

Алгоритм экстраполяции состоит в следующем:

1. Нахождение фреймового частотно-временного спектра $\mathbf{x}' = \mathbf{U}'^* \mathbf{s}'$ сигнала $s'[t']$, где \mathbf{U}' – фреймовый оператор для редуцированного сигнала размерности $(N' \times N'P)$.

2. Добавление nM отсчетов, рассчитанных с помощью весовой функции во фреймовый спектр сигнала \mathbf{x}' и тем самым формирование спектра $\tilde{\mathbf{x}}$ экстраполированного сигнала $\tilde{\mathbf{s}} = (\tilde{s}[0], \dots, \tilde{s}[N-1])^T$.

3. Построение экстраполированного сигнала $\tilde{s}[t]$, $t \in J_N$ обратным фреймовым преобразованием $\tilde{\mathbf{s}} = \mathbf{U}\tilde{\mathbf{x}}$.

Важно заметить, что обратное преобразование занимает не N' точек, а N точек. Поэтому необходимо заново формировать фреймовый матричный оператор \mathbf{U} размерности $(N \times NP)$.

Рассмотрим задачу экстраполяции для сигналов $s'_1[t']$ и $s'_2[t']$.

Зададим исходные параметры для синтеза WH-фрейма, при этом нам нужно два фреймовых оператора для 64 и 80 отсчетов сигнала.

Для $N = 64$:

$P = 8$ – коэффициент перенасыщения;

$M = 16$ – количество сдвигов по частоте;

$\sigma_g = 2.5/M^2 = 0.0098$ – параметр формирующего гауссиана;

$K = \frac{NP}{M} = 32$ – количество сдвигов по времени;

$L_0 = N/K = 2$ – шаг по времени.

Для $N = 80$:

$P = 8$ – коэффициент перенасыщения;

$M = 16$ – количество сдвигов по частоте;

$\sigma = 2.5/M^2 = 0.0098$ – параметр гауссиана;

$K = \frac{NP}{M} = 40$ – количество сдвигов по времени;

$L_0 = N/K = 2$ – шаг по времени.

По формулам (1-4) найдем прямой фреймовый оператор для 64 отсчетов – \mathbf{U}'_{64} и обратный для 80 отсчетов – \mathbf{U}_{80} .

Частотно-временные фреймовые спектры сигнальных векторов \mathbf{s}'_1 и \mathbf{s}'_2 путем преобразований $\mathbf{x}'_1 = \mathbf{U}'_{64} \mathbf{s}'_1$ и $\mathbf{x}'_2 = \mathbf{U}'_{64} \mathbf{s}'_2$. В векторах \mathbf{x}'_1 и \mathbf{x}'_2 представлены частотные и временные сдвиги по следующему алгоритму: $x[1]$ - 1-ый временной и 1-ый частотный сдвиг, $x[2]$ - 2-ый временной и 1-ый частотный сдвиг, $x[32]$ - 32-ой временной (последний) и 1-ый частотный сдвиг, $x[33]$ - 1-ый временной и 2-ой частотный сдвиг и т.д.

Следовательно, для экстраполяции необходимо преобразовать \mathbf{x}'_1 и \mathbf{x}'_2 путем добавления 16 отсчетов в позиции: 65-80, 125-140, ..., 625-640 - для экстраполяции в конце сигнала, и в позиции 33-48, 113-128, ..., 593-608 - для экстраполяции в середине сигнала.

Добавление отсчетов происходит согласно следующей весовой функции

$$x[i] = \frac{n-i+1}{n-m} x[m-1] + \frac{m-n+i+1}{n-m} x[n+1], \quad i \in [m, \dots, n], \quad (9)$$

где m и n - верхняя и нижняя граница вставки отсчетов.

Весовая функция предназначена для интерполяции добавляемых спектральных отсчетов, в результате спектры \mathbf{x}'_1 и \mathbf{x}'_2 расширяются до спектров $\tilde{\mathbf{x}}_1$ и $\tilde{\mathbf{x}}_2$ экстраполированных сигналов. Функция (9) фактически выполняет специфическую линейную интерполяцию нужных участков спектра, однако возможно применение квадратичной или кубической интерполяции, что может улучшить полученный результат.

Восстановим сигнал с помощью обратного фреймового преобразования \mathbf{U}_{80} :

$$\tilde{\mathbf{s}}_{WH1} = \mathbf{U}_{80} \tilde{\mathbf{x}}_1, \quad \tilde{\mathbf{s}}_{WH2} = \mathbf{U}_{80} \tilde{\mathbf{x}}_2, \quad (10)$$

где нижний буквенный индекс у сигналов указывает на используемый WH-метод экстраполяции. На рисунке 3 приведены графики исходного и экстраполированного сигналов.

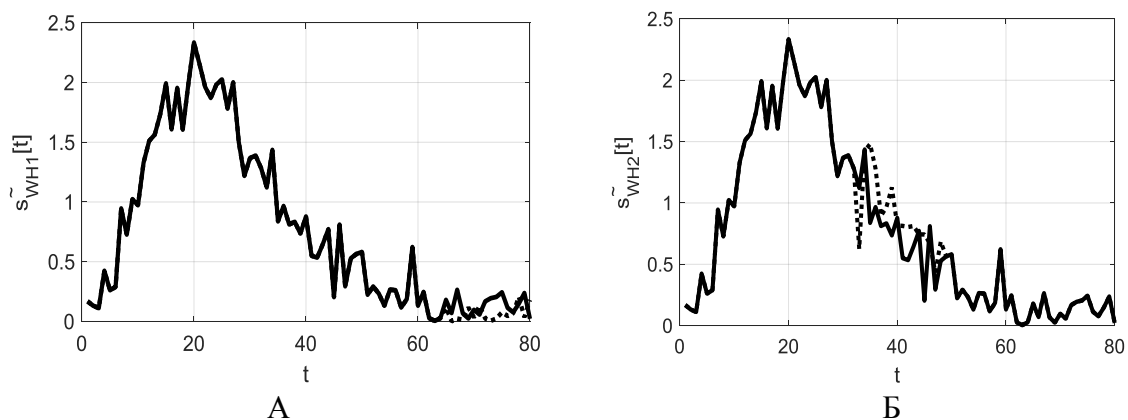


Рис. 3. Графики исходного сигнала – сплошная линия и экстраполированного сигнала – пунктирная линия: А – экстраполяция в конце, Б – экстраполяция в середине

Решение задачи экстраполяции с помощью авторегрессионной модели

Другим, альтернативным, рассмотренному выше, алгоритмом экстраполяции является авторегрессионная (AR) модель [3,4]. Она описывается выражением

$$\tilde{s}_{AR}[t] = \sum_{j=1}^Q a_j s_{AR}[t-j] + \psi[t], \quad t \in \{65, \dots, 80\} \quad (11)$$

где a_j – коэффициент авторегрессионной модели, $\psi[t]$ – ее формирующий шум.

Параметр Q – называют порядком модели. Для построения модели был выбран порядок $Q = 32$, по аналогии с количеством временных сдвигов в WH-фрейме.

Коэффициенты a_j можно найти из уравнений Юла-Уолкера [3]

$$\sum_{m=1}^Q a_m R[m-i] = R[i], \quad i = 1, 2, \dots, Q.$$

Можно записать эту систему в матричном виде

$$\mathbf{R}\mathbf{a} = \mathbf{r},$$

где $\mathbf{R} = (R[i-j])$ - корреляционная матрица сигнала $s[t]$, \mathbf{a} - столбец коэффициентов a_m , \mathbf{r} - столбец значений корреляционной функции $R(k)$ для $k \in [1, \dots, Q]$.

Тогда, оптимальные коэффициенты AR-модели можно найти из выражения

$$\mathbf{a} = \mathbf{R}^{-1} \mathbf{r}. \quad (12)$$

Корреляционная функция незашумленного сигнала $s_o[t]$ (7) известна [5] и равна:

$$R_o[\tau] = \frac{3}{8} \left(15(0.15)^{-7} + 15(0.15)^{-6} |\tau| + 6(0.15)^{-5} \tau^2 + (0.15)^{-4} |\tau|^3 \right) e^{-0.15|\tau|}, \quad \tau \in \{0, \dots, Q\}.$$

Корреляционная функция белого шума $\xi[t]$ описывается выражением

$$R_\xi[\tau] = \begin{cases} \sigma_\xi^2, & \text{если } \tau = 0 \\ 0, & \text{иначе} \end{cases}, \quad \tau \in \{0, \dots, Q\}.$$

Суммарная корреляционная функция зашумленного сигнала (8) равна сумме

$$R[\tau] = R_s[\tau] + R_\xi[\tau] \quad (13)$$

На рисунке 4 приведен график корреляционной функции (13).

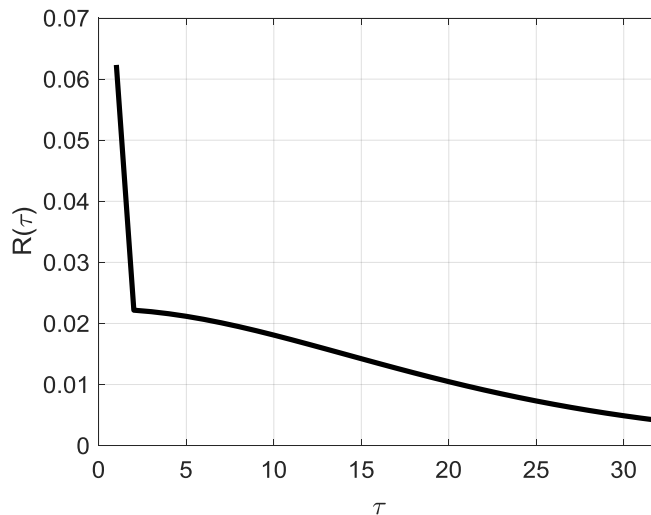


Рис. 4. Автокорреляционная функция анализируемого сигнала

На рисунке 5 приведены графики исходного и экстраполированного сигнала с помощью авторегрессионной модели.

Сравнение экстраполяции WN-фреймом и экстраполяции авторегрессионной моделью

Величину среднеквадратичного отклонения (СКО) исходного зашумленного сигнала и ее экстраполированной оценки для каждой реализации сигнала будем оценивать по формуле

$$\delta = \sqrt{\frac{\sum_{i=0}^{N-1} (s[i] - \tilde{s}[i])^2}{N}}, \quad (14)$$

где $s[i]$ – эталонный сигнал, $\tilde{s}[i]$ – экстраполированный сигнал, N – количество отсчетов этих сигналов на исходной временной сетке.

Для уменьшения статистической погрешности из-за случайного шума было проведено моделирование для 1000 его реализаций и вычислена усредненная по ансамблю реализаций ошибка экстраполяции $\tilde{\delta} = M[\delta]$. В таблице 1 приведены значения СКО для экстраполяции в конце, для различного значения отношения дисперсия шума/максимальная амплитуда сигнала (7).

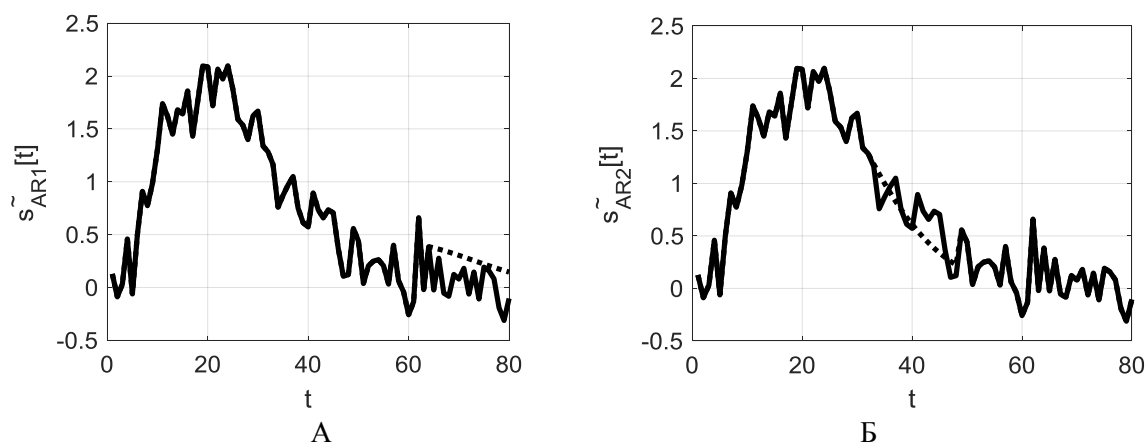


Рис. 5. Графики исходного сигнала – сплошная линия и экстраполированного сигнала – пунктирная линия: А – экстраполяция в конце, Б – экстраполяция в середине

Таблица 1

Значения $\tilde{\delta}$ для экстраполяции в конце сигнала, для различного значения отношения дисперсия шума/максимальная амплитуда полезного сигнала.

| Значение отношения дисперсия шума / максимальная амплитуда сигнала | 0.01 | 0.05 | 0.1 | 0.25 | 0.5 |
|--|--------|--------|--------|--------|--------|
| ВН-фрейм, $\tilde{\delta}$ | 0.0493 | 0.0852 | 0.1227 | 0.1898 | 0.2804 |
| Авторегрессионная модель, $\tilde{\delta}$ | 0.0477 | 0.1062 | 0.1484 | 0.2334 | 0.3462 |

В таблице 2 приведены значения $\tilde{\delta}$ для экстраполяции в середине для различного значения отношения дисперсия шума/максимальная амплитуда сигнала (7).

Таблица 2

Значения СКО для экстраполяции в середине, для различного значения отношения дисперсия шума/максимальная амплитуда полезного сигнала

| Значение отношения дисперсия шума / максимальная амплитуда сигнала | 0.01 | 0.05 | 0.1 | 0.25 | 0.5 |
|--|--------|--------|--------|--------|--------|
| ВН-фрейм, $\tilde{\delta}$ | 0.1343 | 0.1486 | 0.1698 | 0.2253 | 0.2972 |
| Авторегрессионная модель, $\tilde{\delta}$ | 0.2352 | 0.2471 | 0.2712 | 0.3385 | 0.4509 |

Из таблиц 1-2 видно, что при наличии шума дисперсия которого $\sigma_{\xi}^2 \geq 0.1\tilde{s}_{max}$, где \tilde{s}_{max} - максимальная амплитуда импульсного сигнала, авторегрессионная модель показала себя в 1.5 раза хуже, чем экстраполяция с помощью ВН-фрейма. Если же шума нет или его дисперсия $\sigma_{\xi}^2 \leq 0.1\tilde{s}_{max}$, оба метода дают сопоставимую погрешность.

При этом вычислительная сложность авторегрессионного метода ниже, чем у ВН-метода.

Выводы

1. Предлагается новый алгоритм экстраполяции сигнала на основе фреймов Вейля-Гейзенберга, который в отличие от авторегрессионной модели, позволяет с большей точностью экстраполировать нестационарные зашумленные сигналы.

2. Выполненный эксперимент показал, что в случае зашумленного сигнала точность экстраполяции WH-фреймом в 1.5 раза лучше, чем авторегрессионной моделью, если дисперсия зашумления $\sigma_{\xi}^2 \geq 0.1\tilde{s}_{max}$. Если дисперсия $\sigma_{\xi}^2 \leq 0.1\tilde{s}_{max}$, то оба метода дают сопоставимую погрешность.

3. Фреймовый подход не требует априорного знания корреляционной функции сигнала на входе канала и величины дисперсии шума. Оценка параметров авторегрессии по реализации сигнала может приводить к дополнительным значительным ошибкам экстраполяции. Поэтому реальная эффективность авторегрессионного (не смотря на его относительную простоту) в условиях отсутствия априорной информации о корреляционных характеристиках зашумленного сигнала и не большой длительности его реализаций может приводить к плохой оценке параметров авторегрессии и, как следствие, к расхождению экстраполированных оценок на участках прогноза

Литература

1. *Мирошниченко А.В., Волчков В.П.* Применение фреймов Вейля-Гейзенберга в задачах интерполяции и экстраполяции сигналов // Телекоммуникации и информационные технологии. 2019. Т. 6. № 1. С. 134-141.
2. *Волчков В.П., Мирошниченко А.В.* Синтез алгоритмов спектрального анализа сигналов на основе фреймов Вейля-Гейзенберга // Фундаментальные проблемы радиоэлектронного приборостроения. 2018. Т. 18. № 4. С. 1001-1006.
3. *Сергиенко А.Б.* Цифровая обработка сигналов. СПб.: Питер 2002. 608 с.
4. *С.Л. Марпл-мл.* Цифровой спектральный анализ и его приложения: Пер. с англ. М.: Мир, 1990. 584 с.
5. *Денисенко А.Н.* Сигналы. Теоритическая радиотехника. Справочное пособие. М.: Горячая линия-Телеком, 2005. 704 с.
6. *Волчков В.П., Уваров С.С.* Аппроксимация узкополосных случайных процессов с помощью комплексной рекуррентной m-модели скользящего окна второго порядка // Т-Comm: Телекоммуникации и транспорт. 2015. Т. 9. № 3. С. 54-61.
7. *Волчков В.П., Санников В.Г.* Алгебраический подход к оптимальному синтезу вещественных сигнальных базисов Вейля-Гейзенберга // Системы синхронизации, формирования и обработки сигналов. 2018. Т. 9. № 4. С. 19-25.
8. *Санников В.Г., Алёшинцев А.В.* Математическое моделирование многочастотного модема с повышенной помехоустойчивостью // Т-Comm: Телекоммуникации и транспорт. 2016. Т. 10. № 7. С. 52-58.
9. *Санников В.Г., Корольков А.А., Герасименко Х.В.* Метод совместного оценивания параметров и фильтрации зашумленной речи в голосовых кодеках // Т-Comm: Телекоммуникации и транспорт. 2014. Т. 8. № 12. С. 71-73.

EXTRAPOLATION OF NON-STATIONARY NOISED SIGNALS USING WEIL-HEISENBERG FRAMES

Anton V. Miroshnichenko
MTUCI, Student
Mirosh.A.V@yandex.ru
Valery P. Volchkov
MTUCI, Doctor of Science
volchkovvalery@mail.ru

Keywords: *Weil-Heisenberg frame, WH frame, extrapolation, autoregression, unsteady signal*

To solve the problem of extrapolating unsteady noisy signals, it is proposed to use Weyl-Heisenberg hard frames. An algorithm for the extrapolation of such signals using direct and inverse frame transforms is constructed and its experimental comparison with the classical extrapolation algorithm based on the autoregressive model is presented. It is shown that frame extrapolation allows one to obtain much smaller extrapolation errors.

МЕТОДЫ СИНХРОНИЗАЦИИ ПОТОКОВ В ВИДЕОИНФОРМАЦИОННЫХ СИСТЕМАХ

Валицкая Наталья Сергеевна
студент группы МРА1901 МТУСИ
nvalitskaya@mail.ru

Власюк Игорь Викторович
МТУСИ, к.т.н., доцент кафедры ТуЗВ
ru3dlp@yandex.ru

Ключевые слова: Видеоинформационные системы, Black Burst, Синхронизация, Временная метка, PCR, Джиттер пакетов.

Одним из основополагающих принципов систем телевидения, в частности, видеоинформационных систем, является синхронизация. Настоящая работа посвящена анализу проблемы синхронизации в видеоинформационных системах, имеющих такие параметры развертки, при которых требуется совмещать изображение из нескольких частей (тайлов), формируемых различными источниками, например, видеосерверами. Проанализированы основные применяемые в современных телевизионных системах виды синхронизации и возможности их применения для решения задачи формирования изображений сверхвысокой чёткости с нестандартными параметрами. Намечены пути совершенствования существующих практик решения задачи синхронизации субпоток в видеоинформационных системах.

Введение

Видеоинформационные системы (ВИС) – системы представления медиаданных с устройств отображения (дисплеев, проекционных систем и т. д.) различных размеров, установленных в местах массового скопления людей и предназначенных для коллективного просмотра.

Ввиду активного развития видеоинформационных систем актуальна проблема разработки методов синхронизации видеокадров, являющихся частями общего изображения, при отображении на соответствующих экранных модулях.

В ВИС изображение передается на экран с медиасервера или нескольких медиасерверов. В таких устройствах, фактически представляющих собой персональный компьютер в промышленном исполнении, есть один или несколько видеоадаптеров, для вывода на экраны сигнала изображения при этом используется какой-либо видеоинтерфейс. Обычно это интерфейс HDMI, больше подходящий для бытовой аппаратуры, или рекомендуемые для профессионального использования HDMI, SDI и DisplayPort.

Каждый видеоадаптер обычно имеет несколько независимых выходов, соответственно, к нему можно подключить несколько устройств отображения. При этом даже в пределах одного видеоадаптера не предусматривается возможности синхронизации выходных потоков. Интерфейс видеоадаптера генерирует видеосигнал с постоянной тактовой частотой, соответственно, и частотой кадров. При этом рендеринг передаваемого изображения осуществляется за время, определяемое сложностью обработки и производительностью системы, которая отправляет отсчеты сигнала в кадровый буфер. То есть, поскольку генерация потоков и их отображение происходят несинхронно, наблюдается так называемый тиринг, т.е. горизонтальные разрывы: например, снизу экрана отображается часть (n-1) кадра, а сверху уже начался следующий (n кадр). Такой эффект возникает в пределах одного видеовыхода. Частично устранить его позволяет алгоритм, запрещающий обновление кадрового буфера за пределами периода кадрового гасящего интервала. Однако это приводит к стробоскопическим эффектам, связанным с несинхронностью процессов рендеринга и вывода видеосигнала. Между двумя дисплеями может возникать вертикальный разрыв изображения, поскольку кадровые развертки отдельных выходов несинфазны.

Синхронизация призвана обеспечить отсутствие данных артефактов. Производители профессиональных видеоадаптеров NVIDIA и AMD разработали проприетарные технологии син-

хронизации отдельных видеовыходов адаптеров и выпустили соответствующие платы синхронизации, устанавливающиеся в видеосервер. Это решение позволяет синхронизировать до четырех видеоадаптеров в пределах одного или нескольких серверов. Таким образом, хотя в видеосервер можно поместить и больше видеоадаптеров, но существующие решения в области синхронизации не позволяют реализовать это. То есть, если видеоадаптеры имеют по 4 выхода, можно получить в пределах 16-ти выходов синхронное изображение с точностью до кадра.

Устройства отображения различаются разрешением и форматом кадра. В случае, когда от ВИС требуется очень высокое разрешение (например, 20000x16000), не только нет интерфейса, чтобы передать такое изображение, но и, ресурсов одного видеосервера будет недостаточно.

Для решения этой проблемы используют несколько видеосерверов, объединенных в сеть с помощью Ethernet-коммутатора. Видеосерверы параллельно обрабатывают отдельные части изображения. Каждый экранный модуль ВИС получает от коммутатора свою часть общего изображения, закодированного одним из видеокодеков с малыми задержками с инкапсуляцией в IP-пакеты и распаковывает с помощью встроенного IP-декодера. Схема видеоинформационной системы на базе IP показана на рисунке 1.

Сетевая технология может быть нестабильна по величине вносимых задержек вследствие заложенного принципа лучшей попытки (best effort), поэтому возникает проблема синхронизации сигналов при отображении на видеостене. В случае, если одно изображение разделяется на все экранные модули видеостены, наблюдается нарушение целостности изображения: один экран показывает часть изображения n кадра, а другой, например, часть $(n-1)$ кадра. Проблемы передачи по сети усиливаются при увеличении ее загруженности. При использовании технологии Ethernet не рекомендуется загрузка сети более 70%.

В ситуациях с повышенными требованиями к синхронизации (например, при проведении мероприятий с участием первых лиц государства), чтобы избежать проблем, фактически стандартом стала демонстрация статичных изображений, но даже в этом случае могут возникать артефакты в виде, например, временного «выпадения» части изображения из-за сбоев на уровне сети и программного обеспечения.

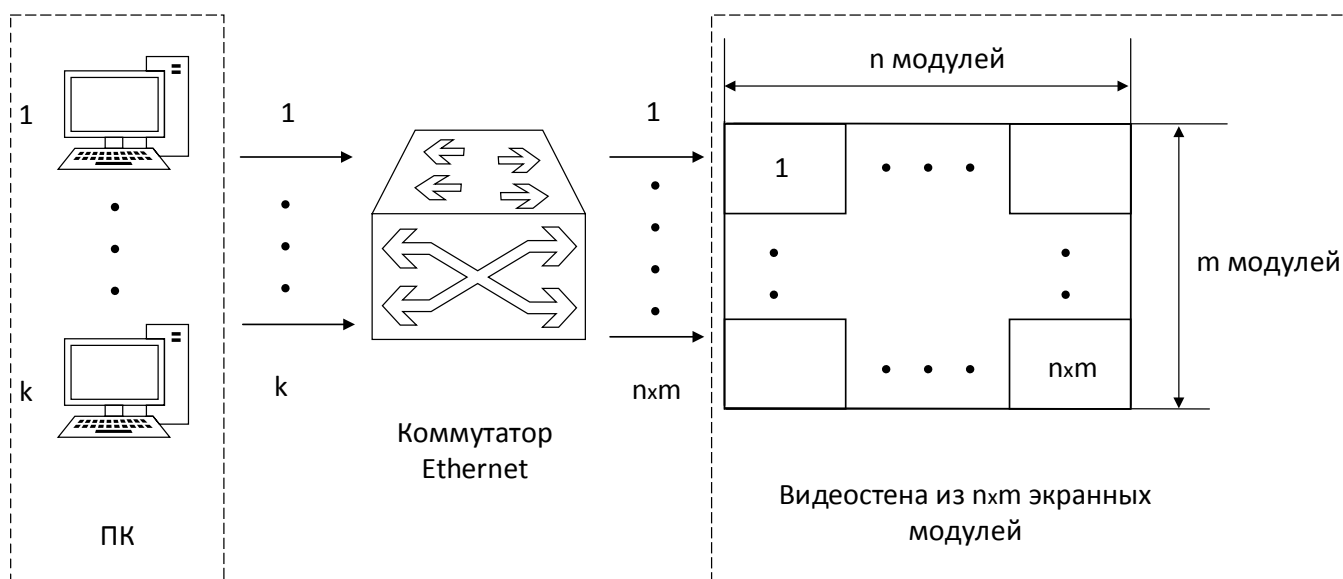


Рис. 1. Схема видеоинформационной системы на базе IP

Синхронизация в аналоговых системах

В аналоговых системах была простая, но очень надежная строчная и кадровая синхронизация: на развертывающие устройства вместе с сигналом изображения подавались специальные импульсы на уровне «чернее черного» (уровень черного – «0», уровень синхроимпульсов «-43%» – для удобства обнаружения), которые заставляют срабатывать их в строго определенный момент. Импульсы для строк и полей различались длительностью, поэтому легко выделялись дифференцирующей и интегрирующей цепями соответственно. После прохождения сигнала через диффе-

ренцирующую цепь во время кадрового синхроимпульса отсутствуют строчные, может сбиться строчная синхронизация, и чтобы этого не произошло, применяют специальные врезки. В таком случае, после прохождения сигнала через дифференцирующую цепь образуются положительные импульсы, следующие со строчной частотой. В случае чересстрочной развертки форма синхроимпульсов для четного и нечетного поля не совпадает, что приводит к сдвигу между моментами начала обратных ходов развертки, и в итоге – к «спариванию» строк. Для устранения этого эффекта применялись уравнивающие импульсы (врезки следуют с двойной строчной частотой). Принципы аналоговой синхронизации показаны на рис. 2 [1].



Рис. 2. Форма сигналов синхронизации: а) черного поля (black burst), б) трехуровневой

Синхронизация в цифровых системах

С появлением цифровых систем и сжатия начались проблемы с синхронизацией, поэтому SDI-интерфейсы синхронизировали с помощью аналогового сигнала Black Burst. Пример приведен на рисунке 3 из документа [2]. Цифровой сигнал передается по коаксиальным кабелям, коммутатор соединяет выход каждого устройства со входом каждого устройства.

Позже в цифровых системах были придуманы PCR (Program Clock Reference) метки, которые раз в некоторое время передаются в транспортном потоке MPEG и содержат информацию о том, сколько точно времени прошло от прошлой метки. Значение PCR содержится в поле адаптации и имеет длину 42 бита. 33 старших бита относятся к тактовой частоте 90 кГц, а 9 младших считаются от 0 до 299. Значение 33-битного счетчика увеличивается на единицу при каждом переходе из максимума (299) в минимум (0). Произведение $90000 \text{ Гц} \times 300$ как раз дает опорную частоту 27 МГц. Эти метки вставляются в поток кодером не реже, чем каждые 40 мс для DVB (или 100 мс MPEG-2), в противном случае при декодировании произойдет рассинхронизация, выражающаяся, например, в несовпадении воспроизведения видео- и аудиоданных, в артефактах на изображении. При увеличении битрейта, т.к. временной интервал между врезками не меняется, метки будут вставляться через большее количество пакетов.

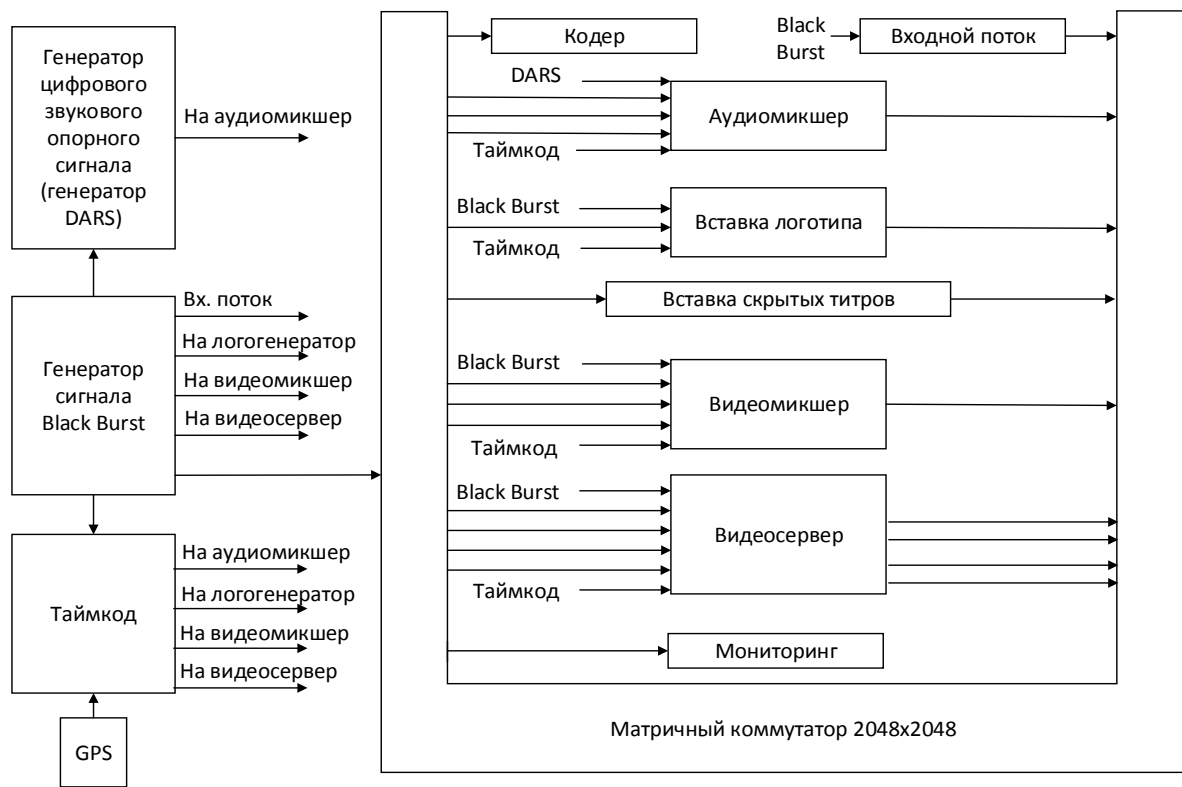


Рис. 3. Синхронизация SDI-интерфейсов с помощью сигнала Black Burst.

С помощью этих меток декодер вычисляет системную частоту (system clock frequency) и сравнивает ее с частотой своего опорного генератора (27 МГц). Если отклонение превышает ± 810 Гц, декодер выдает ошибку и запускает фазовую автоподстройку частоты. ФАПЧ – это система, которая подстраивает фазу управляемого генератора под фазу опорного сигнала.

Структурная схема приема и декодирования транспортного потока приведена на рис. 4 [3].

Декодер канала преобразует входной поток данных в транспортный поток, который уже поступает в блок декодирования и демультиплексирования. Из транспортного потока извлекаются пакеты PAT (program association table) и PMT (program map table), из которых получают необходимые при демультиплексировании идентификаторы пакетов (PID). PID PAT всегда 0x0. Пакеты, принадлежащие одному элементарному потоку, имеют одинаковый PID. При демультиплексировании элементарные потоки извлекаются из транспортного, группируются в потоки с одинаковым PID и поступают на декодеры.

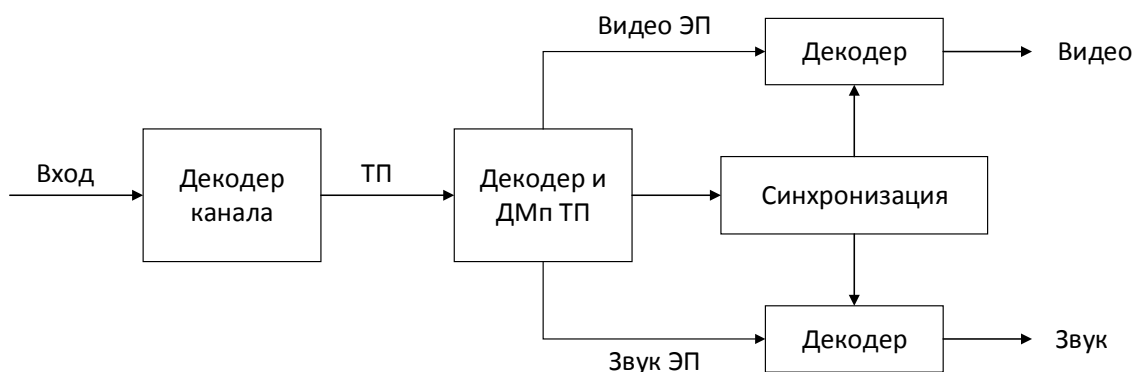


Рис. 4. Структурная схема приема и декодирования транспортного потока MPEG-2

Частоты генераторов тактовых импульсов в блоке синхронизации подстраивают по меткам времени PCR так, чтобы количество тактовых импульсов, сформированных в декодере между двумя метками, соответствовало интервалу между моментами времени, зафиксированными в этих метках.

Синхронизация в системах на базе IP-инфраструктуры

В настоящее время наблюдается переход на IP-инфраструктуру, что позволяет упростить и сделать более гибкой структуру распределения и коммутации медиаданных.

IP (Internet Protocol) - протокол сетевого уровня стека TCP/IP, объединяет сегменты сети в единую сеть путем построения логической адресации, обеспечивая доставку пакетов данных между любыми сетевыми узлами через произвольное число промежуточных узлов (маршрутизаторов). IP-пакеты состоят из заголовка и полезной нагрузки. В случае потоковой передачи по IP-сетям множество полезных протоколов, таких как RTP и UDP, могут передаваться в полезной нагрузке IP, каждый со своим собственным заголовком и полезной нагрузкой. Например, в Таблице 1 показаны видеоданные, переносимые в RTP-пакете, переносимом в UDP-пакете, переносимом в IP-пакете [4].

Таблица 1

Структура IP-пакета.

| IP-пакет (максимальная длина 1500 байтов) | | | |
|---|---------------------------|----------------------------|--|
| IP-заголовок (20 байт) | UDP-заголовок (8 байт) | RTP-заголовок (16 байт) | Нагрузка RTP-пакета, например, 7 TS-пакетов MPEG по 188 байтов каждый (4 байта – заголовок, 184 – полезная нагрузка) |

Источник генерирует пакеты с постоянной скоростью и отправляет через одинаковые интервалы времени. Пакетам нужно некоторое время, чтобы прибыть на сторону получателя (например, 1 с). В таком случае потребитель воспроизводит пакеты в те же моменты времени, что они были отправлены + 1 с. Но как уже было сказано выше, сетевая технология может быть нестабильна с точки зрения передачи данных, в таком случае задержки увеличиваются, при чем время между моментами прибытия двух пакетов неодинаково. Данные воспроизводятся с задержками, и чтобы этого избежать, используется буферизация: пакеты придерживаются на некоторое время и воспроизводятся с постоянной скоростью. Чтобы это все реализовать, нужно знать, когда воспроизвести содержимое одного пакета относительно предыдущего. Устранение перерывов между пакетами при воспроизведении иллюстрирует рисунок 5. t_1 , t_2 , t_3 – моменты времени, в которые отправляются 1-й, 2-й и 3-й пакеты соответственно. T – длительность информации, содержащейся в пакете, примем одинаковой для все пакетов. При отправлении 1-го пакета по сети в момент времени t_1 возникнет задержка Δt_1 (время доставки) и приемной стороной пакет будет получен в момент времени $t_1 + \Delta t_1$. В отсутствие других задержек, остальные пакеты будут получены в моменты времени $t_2 + \Delta t_1$, $t_3 + \Delta t_1$ и т.д. В случае задержек, второй и третий пакеты придут в моменты времени $t_2 + \Delta t_1 + \Delta t_2$ и $t_3 + \Delta t_1 + \Delta t_2 + \Delta t_3$ соответственно. Для устранения проблем необходимо разделить время приема и время воспроизведения. Тогда первый пакет будет воспроизведен в момент времени $t_1 + \Delta t_1 + \text{порог}$, где $\text{порог} = \Delta t_1 + \Delta t_2 + \Delta t_3$. Остальные пакеты будут воспроизводиться без задержек.

При использовании IP-инфраструктуры, полезную нагрузку, например, транспортные потоки MPEG, переносят с помощью протокола RTP (Real-time Transport Protocol). Протокол RTP используется для передачи трафика в режиме реального времени. В RTP-заголовке 32 бита отводится для поля «метка времени» и 16 бит для порядкового номера. Метка времени нужна синхронизации: чтобы получатель мог воспроизводить данные с такой же скоростью, что и отправитель. Значение метки для пакета – сумма значения метки предыдущего пакета и вырабатываемого (отсчитываемого) времени первого байта. Последовательный номер – для определения потери или нарушения порядка пакетов. Очевидно, что этот номер для каждого пакета больше на единицу, чем номер предшествующего ему пакета. Для первого пакета метка времени и номер – случайные числа. Также есть поле длиной 7 бит для информации о формате полезной нагрузки (типе кодирования данных, что необходимо для восстановления).

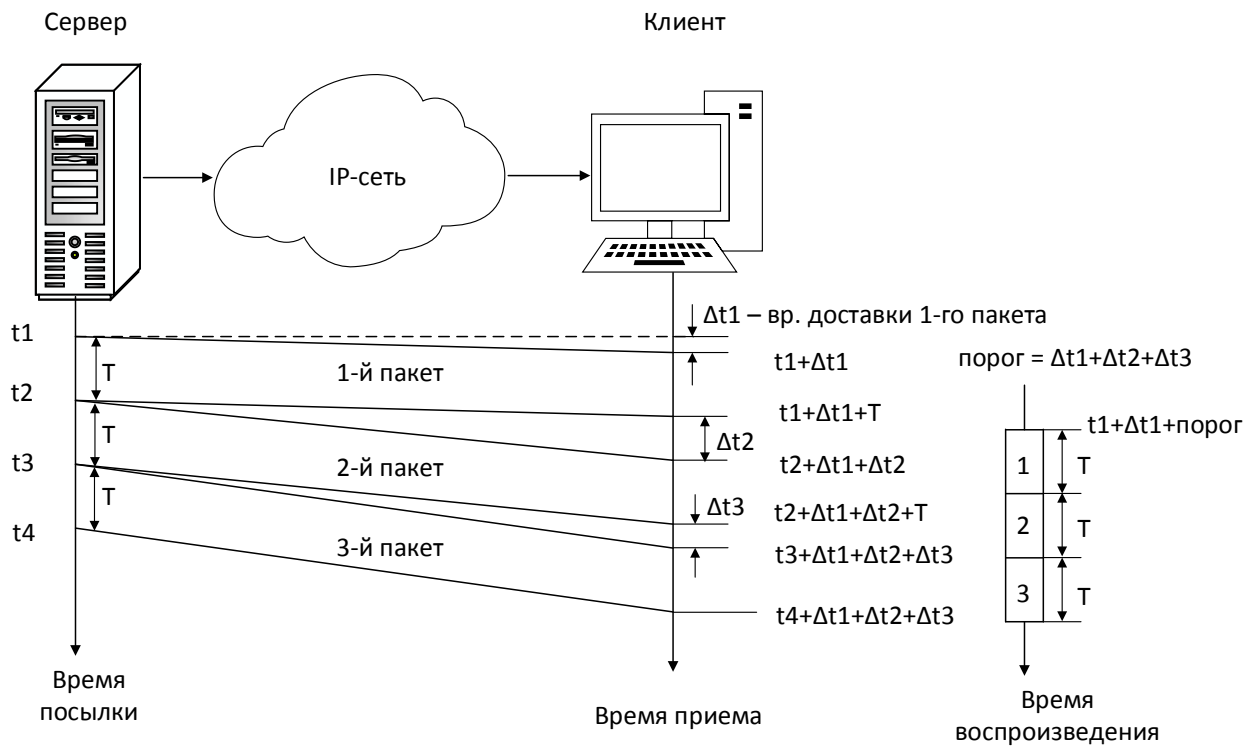


Рис. 5. Устранение перерывов между пакетами при воспроизведении

Однако анализ текущей ситуации в области исследования на базе литературных источников и научно-исследовательских работ показывает, что вопрос синхронизации при воспроизведении изображений очень высокого разрешения на многих экранах недостаточно решен.

Проблема рассматривалась, например, в статье [5]. В результате разработанного авторами алгоритма разница между временем воспроизведения компьютерами видеокладов, являющихся частями общего изображения составляет около 30 мс. Также упоминается проблема артефактов сжатия. Для тестирования алгоритма формирования потока кадров используется кодек H.264. H.264 кодирует информацию блоками (например, 16x16), и в случае, если размеры кадров не кратны этим блокам, на границах стыковок кадров при динамических изменениях появляются искажения квадратной формы. Методы видеокompрессии, применяемые в видеоинформационных системах на базе IP, как правило, основаны на блочном кодировании.

На рисунке 6 изображена схема передачи, состоящая из кодера, транспортного оборудования и IP-сети. Видео и звук поступают на вход кодера, преобразуются в транспортные потоки, затем – в IP-пакеты и передаются по сети.

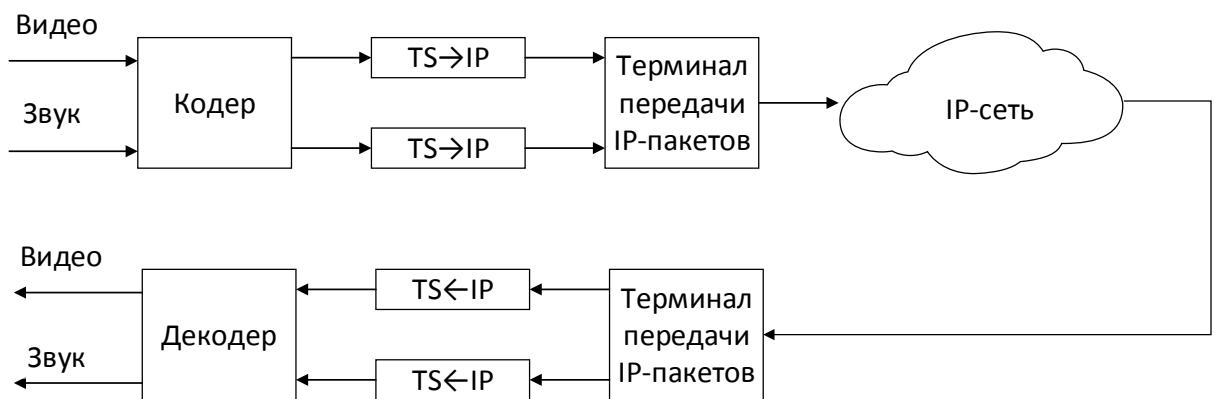


Рис. 6. Схема передачи медианных по IP-сети.

Существующие методы синхронизируют потоки в процессе упаковки, но, как следует

из рисунка 6, далее нужно декодировать транспортные потоки, и на этом этапе может произойти рассинхронизация: нет гарантии, что потоки декодируются за контролируемое время.

Заключение

Таким образом, можно утверждать, что синхронизация субпотоков должна быть реализована на уровне видеокodeка, который должен быть специально адаптирован для формирования тайлов общего изображения, поскольку это первая и соответственно последняя операция в процессе передачи контента. Синхронизация в момент упаковки не гарантирует при воспроизведении на экранных модулях частей одного и того же изображения. Следует отметить, что, наиболее эффективна синхронизация на уровне уже декодированного потока, но из-за возможной значительной рассинхронизации отдельных тайлов этот подход требует во-первых значительного буфера памяти для компенсации джиттера потоков, во-вторых, относительно сложной структуры данных, поскольку даже кадрового временного интервала на практике недостаточно для корректной синхронизации, и, в-третьих, указанное решение требует отдельных каналов связи для передачи такого синхросигнала с соответствующей системой контроля задержек распространения.

Литература

1. Джакония В.Е., Гоголь А.А., Друзин Я.В. и др.; Под ред. В. Е. Джаконии. Телевидение. Учебник для ВУЗов. 4-е изд., стереотип. М.: Горячая Линия – Телеком. 2007. 616 с.
2. J. Patrick Waddell. *Beyond Black Burst / SMPTE, 32NF-80 WG on Timing and Sync*. 2016. URL:https://www.smpte.org/sites/default/files/sectionfiles/Beyond%20Black%20Burst_0.pdf.
3. Безруков В.Н., Власюк И.В., Канев С.А. Оценка относительного отклонения частоты при приеме сигналов наземных цифровых телевизионных вещательных систем // Т-Comm: Телекоммуникации и транспорт. 2010. Т. 4. № 6. С. 9-11.
4. Alex MacAulay, Boris Felts, Yuval Fisher. IP Streaming of MPEG-4: Native RTP vs MPEG-2 Transport Stream // Envivio, Inc.(2005). URL:<http://pdf.textfiles.com/manuals/STARINMANUALS/Envivio/Manual>.
5. Городилов М.А., Долговесов Б.С., Храпцов И.Д., Радостев А.Х. Особенности построения систем для полиэкранного отображения распределенных мультимедийных данных // Институт автоматики и электротехники СО РАН // Вестник НГУ. Серия: Физика. 2015. Т. 10, выпуск 2. С. 91–98.

SYNCHRONIZATION METHODS FOR VIDEO-INFORMATION SYSTEMS

Natalya S. Valitskaya

student of group MRA1901, MTUCI

nvalitskaya@mail.ru

Igor V. Vlasyuk

MTUCI, Ph.D., associate professor of the TaAB Department

ru3dlp@yandex.ru

Keywords: *Video Information Systems, Black Burst, Synchronization, Timestamp, PCR, Packet Jitter*

One of the fundamental principles of television systems, in particular, video information systems, is synchronization. The present work is devoted to the analysis of the synchronization problem in video information systems having such scan parameters for which it is necessary to combine an image from several parts (tiles) formed by various sources, for example, video servers. The main types of synchronization used in modern television systems are analyzed and the possibilities of their application for solving the problem of forming ultra-high definition images with non-standard parameters. Ways of improving existing practices for solving the problem of synchronizing subflows in video information systems are outlined.

РАЗРАБОТКА И ПРОЕКТИРОВАНИЕ УСТРОЙСТВА ПРЕДВАРИТЕЛЬНОЙ ФИЛЬТРАЦИИ С ИСПОЛЬЗОВАНИЕМ ГИБРИДНЫХ БАНКОВ ФИЛЬТРОВ

Офицера Ксения Борисовна
студентка группы МИТ1901, МГУСИ

o.k.b.97@yandex.ru

Лобов Евгений Михайлович
МГУСИ, к.т.н., доцент кафедры РТС
lobovrts@yandex.ru

Ключевые слова: Фильтры, гибридные банки фильтров, узкополосная помеха, предварительная фильтрация

В данной работе объектом исследования является устройство предварительной фильтрации с использованием гибридных банков фильтров. Цель работы – разработка и реализация устройства предварительной фильтрации с использованием гибридных банков фильтров. Для достижения поставленной цели были решены следующие задачи: рассчитана и спроектирована принципиальная схема каскада аналоговых фильтров анализа, скорректирована принципиальная схема каскада аналоговых фильтров анализа с учетом типономиналов дискретных элементов, спроектирована и изготовлена печатная плата каскада аналоговых фильтров анализа, подобраны комплектующие и произведен их монтаж на печатной плате.

В настоящее время всё больше и больше приложений в системах радиодоступа, радиолокации и радиомониторинга требуют цифровой обработки широкого диапазона частот (порядка 10 ГГц). Устройства оцифровки со сверхвысокой скоростью используют архитектуру с чередованием отсчетов во времени с помощью применения набора АЦП и обладают относительно малой разрядностью преобразования. Дополнительные ошибки оцифровки также обуславливаются случайными сдвигами во времени чередующихся отсчетов друг относительно друга (рассинхронизация АЦП по фазе, временной джиттер) и неравенством опорных напряжений применяемых АЦП (рассогласование АЦП по уровню). Таким образом, упомянутые устройства требуют калибровочных схем для уменьшения ошибок, которые значительно увеличивают их потребление и габариты.

Использование гибридного банка фильтров позволяет альтернативным способом достичь высоких скоростей преобразования при высокой разрешающей способности [1]. Гибридный банк фильтров (НФВ) это аналого-цифровое устройство, которое использует как аналоговые (АФВ), так и цифровые фильтры [4-6]. Аналоговые и цифровые фильтры в гибридном фильтре банков должны быть разработаны таким образом, чтобы они адекватно изолировали каналы и не вносили ошибок реконструкции, которые ограничивают точность оцифровки сигнала банком фильтров. Целью этапа синтеза НФВ является восстановление исходного сигнала. Таким образом, НФВ ведет себя как идеальная задержка для желаемой полосы пропускания [2].

Устройство предварительной фильтрации с использованием гибридных банков фильтров служит защитой от узкополосных помех при оцифровке широкой полосы частот. Если оцифровывать широкополосный сигнал напрямую, то помеха может перегрузить АЦП. Используя устройство предварительной фильтрации, мы делим диапазон частот на отдельные поддиапазоны и оцифровываем каждый из них отдельно. Если где-то помеха перегрузит АЦП, то этот поддиапазон мы не включаем в «склейку» с другими поддиапазонами.

Проектирование, корректировка и расчет принципиальной схемы каскада аналоговых фильтров

Из общей структурной схемы гибридного банка фильтров (см. рисунок 1) видно, что широкополосный сигнал поступающий в аналоговый банк фильтров делится на более узкие полосы и поступают в АЦП, где потом “склеиваются” в один сигнал в цифровом банке фильтров. Гибридный банк фильтров позволяет работать с системами с высокими скоростями, само же устройство предварительной фильтрации не повышает скорость, оно позволяет использовать, низкоскоростные АЦП [3]. Описанное в статье устройство работает в низком диапазоне частот и не предназначено для высокоскоростной оцифровки сигналов. Оно используется как устройство защиты от узкополосных помех при оцифровке широкой полосы частот. Расчетная часть процесса разработки устройства состояла в выборе функции фильтрации, определении порядка фильтра, ширины полосы пропускания и диапазона частот. В результате была спроектирована принципиальная схема устройства предварительной фильтрации (см. рисунок 2).

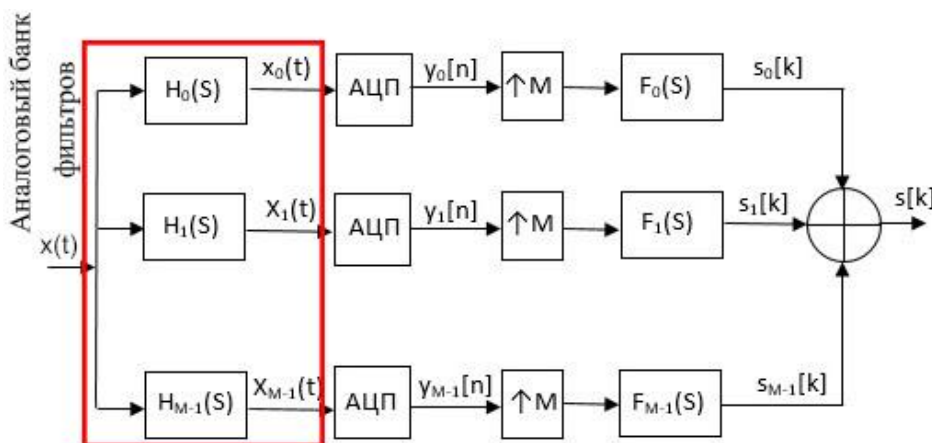


Рис. 1. Общая структурная схема гибридного банка фильтра

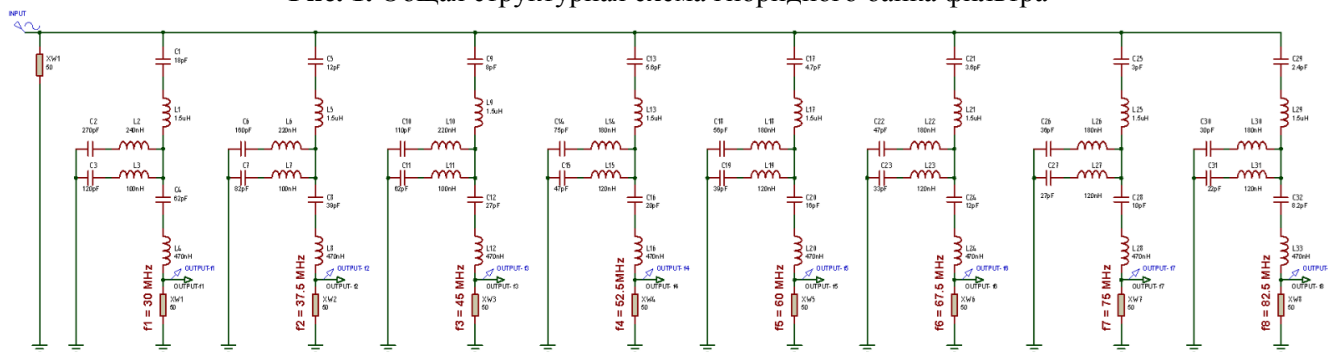


Рис. 2. Принципиальная схема устройства предварительной фильтрации с использованием гибридных банков фильтров

Проектирование печатной платы и ее монтаж

На этапе проектирования на плате были размещены восемь фильтров, выходы которых находятся вдоль двух краев платы по бокам, а общий вход расположился по середине. Элементная база для устройства включает только пассивные реактивные элементы - конденсаторы и катушки индуктивности. Резисторы, которые есть в принципиальной схеме заменяются разъемами с таким же сопротивлением. Реактивные элементы, в рамках одного канала были расположены таким образом, чтобы минимизировать их взаимное влияние друг на друга. Катушки индуктивности были расположены максимально далеко друг от друга, а если такой возможности не было, то ближайšie катушки устанавливались перпендикулярно друг к другу, но так чтобы длина проводников была минимальна. Все компоненты были расположены на одном слое. Полученная печатная плата изображена на рис. 3. Компоненты для монтажа печатной платы были подобраны с реальными номиналами близстоящими к моделированным, их подбор производился с помощью моделирования.

Результаты работы готового устройства

Устройство (рис. 4) было проверено с помощью измерителя АЧХ. Проведено сравнение частотной характеристики при моделировании (рис. 5) и эксперименте (рис. 6). Из эксперимента видно, что центральная частота сдвинулась по сравнению с результатами моделирования примерно на 2-5 МГц, а при увеличении частоты, это число увеличивалось до 10 МГц. Также по сравнению с моделированием затухание с увеличением частоты увеличивается. Одна из причин такого явления это использование монолитных многослойных ферритовых чип-индуктивностей общего применения. У таких индуктивностей при повышении частоты уменьшается добротность, а уменьшение добротности приводит к увеличению затухания.

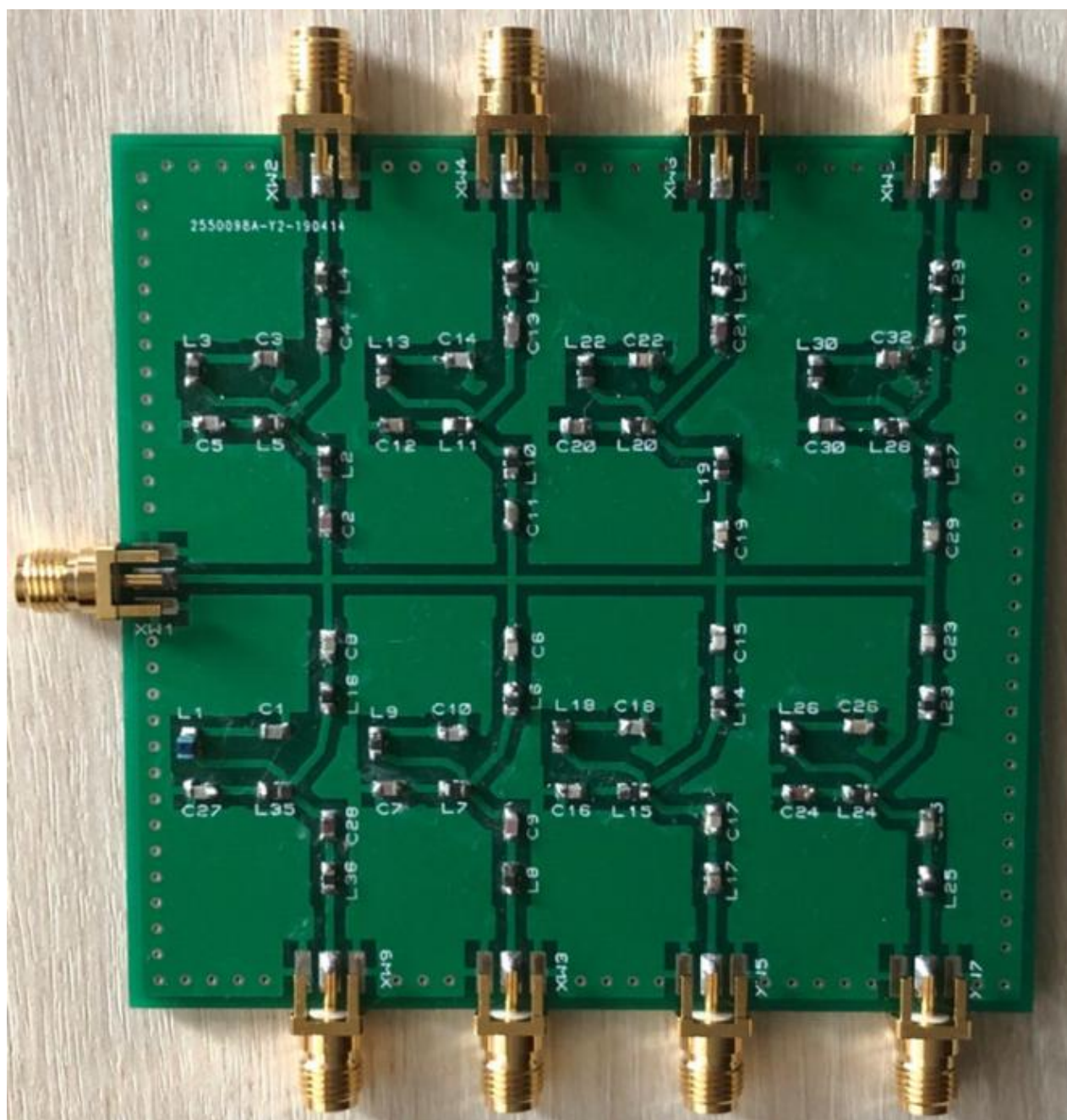


Рис. 3. Готовое устройство предварительной фильтрации с использованием гибридных банков фильтров

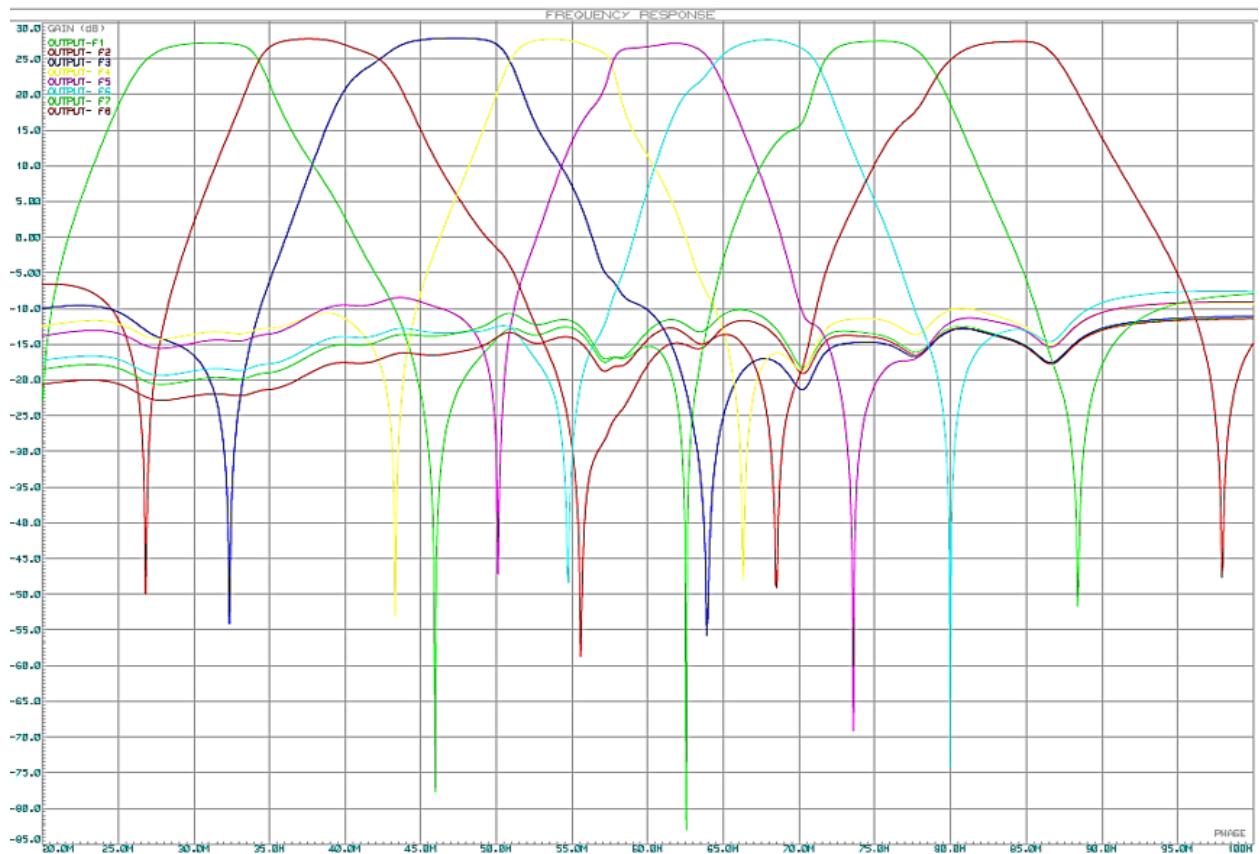


Рис. 4. Результаты моделирования аналоговых фильтров из состава банка фильтра

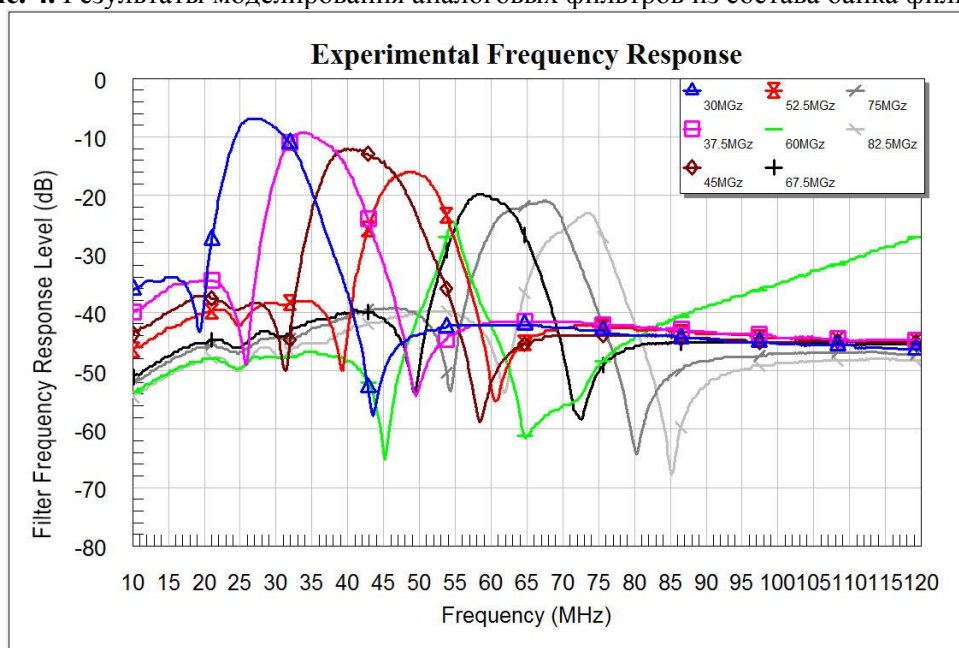


Рис. 5. Результаты экспериментального измерения частотных характеристик банка фильтров

Заключение

В результате работы был получен банк аналоговых фильтров анализа из состава гибридного банка фильтров устройства предварительной фильтрации. Разработанное устройство требует модификаций с целью корректировки его реальных характеристик, а именно проектирование принципиальной схемы банка фильтров и разводки печатной платы с учетом влияния характеристик материала печатной платы и геометрии проводящих дорожек на АЧХ фильтров. Также необходим подбор реальных элементов с номиналами наиболее близкими к расчетным или закупка более дорогостоящих с меньшим отклонением от заявленного номинала.

Литература

1. *Velazquez S.R.* Design of hybrid filter banks for analog/digital conversion // IEEE Transactions on Signal Processing, vol.46. April 1998. С. 1-6.
2. *José Pedro da Rocha Magalhães.* Desenvolvimento de um Front-end para Cognitive Radio. Master dissertation. Universidade de Aveiro, Aveiro, 2011.
3. *José P. Magalhães; José M. N. Vieira; Roberto Gómez-García; Nuno B. Carvalho.* Oversampled Hybrid Filter Banks for High-Dynamic-Range Cognitive-Radio/Software-Defined-Radio Receivers // 2013 IEEE MTT-S International Microwave Symposium Digest (MTT). June 2013. С. 1-2.
4. *Лобов Е.М., Лобова Е.О., Курочкин А.А.* Вычислительно упрощенная реализация компенсатора дисперсионных искажений на базе банка цифровых фильтров // Системы синхронизации, формирования и обработки сигналов. 2018. Т. 9. № 1. С. 106-110.
5. *Лобов Е.М., Лобова Е.О., Елсуков Б.А.* Устройство компенсации дисперсионных искажений широкополосных сигналов на базе банка цифровых фильтров // DSPA: Вопросы применения цифровой обработки сигналов. 2018. Т. 8. № 3. С. 60-65.
6. *Платонов О.И., Лобов Е.М.* Цифровые банки фильтров анализа-синтеза сигналов // Телекоммуникации и информационные технологии. 2017. Т. 4. № 1. С. 24-29.

DEVELOPMENT AND DESIGN OF DEVICES PRE-FILTERING USING A HYBRID FILTER BANK

Kseniya B. Ofitserova

*Student of group MIT 1901, MTUCI
o.k.b.97@yandex.ru*

Evgeniy M. Lobov

*MTUCI, PhD., associate professor of RTS department
lobovrts@yandex.ru*

Keywords: *Filter, hybrid bank filter, narrowband interference, pre-filtering*

In this work, the object of research and development is a pre-filtering device using hybrid filter banks. The purpose of the work is the development and design of a preliminary filtration device using hybrid filter banks. To achieve this goal, the following tasks were solved: the circuit diagram of the cascade of analogue analysis filters was calculated and designed, the circuit diagram of the cascade of analogue analysis filters was adjusted taking into account the typical values of discrete elements, the circuit board of the cascade of analogue analysis filters was designed and manufactured, the components were selected and the circuit board was soldered. At the end of the work, a finished, working device was obtained.

ИССЛЕДОВАНИЕ ПАРАМЕТРИЧЕСКИХ НЕЛИНЕЙНЫХ ИСКАЖЕНИЙ В УСИЛИТЕЛЯХ С РАСПРЕДЕЛЕННЫМ УСИЛЕНИЕМ

Шмаков Никита Дмитриевич

Аспирант кафедры РОС МТУСИ

Shmaki-shmak@yandex.ru

Иванюшкин Роман Юрьевич

МТУСИ, к.т.н., доцент кафедры РОС

rivanyushkin@gmail.com

Ключевые слова: усилитель с распределенным усилением, нелинейные искажения, искусственные длинные линии, амплитудная характеристика, фазоамплитудная характеристика

Обсуждаются причины возникновения параметрических нелинейных искажений в усилителях с распределенным усилением, построенных на полевых транзисторах. Приведены схемы построения усилителей с распределенным усилением для исследования параметрических нелинейных искажений. Представлены результаты исследования таких нелинейных искажений на упрощенной модели усилителя с распределенным усилением, построенного на источниках тока, управляемых напряжением.

На сегодняшний день построение усилителей мощности по технологии распределенного усиления перспективно в различных областях. Одной из таких перспективных областей являются предварительные усилители для передатчиков цифрового телевизионного вещания [6-17], поскольку в таких усилителях достаточно важны диапазонные свойства, простота реализации, универсальность. При этом, требования к энергетическим характеристикам не столь существенны.

Компьютерная модель усилителя с распределенным усилением [1,2,5], построенного на полевых транзисторах BLF 278, представлена на рис. 1.

На основе приведенной на рис.1 схеме исследованы амплитудная и фазоамплитудная характеристики УРУ. Амплитудная характеристика представлена на рис. 2.

Амплитудная характеристика имеет нелинейности во всей рабочей области, но жесткая отсечка на начальном участке отсутствует [2,3]. Фазоамплитудная характеристика представлена на рис.3. Фазоамплитудная характеристика существенно неравномерна в области до 1.5 В [2,3].

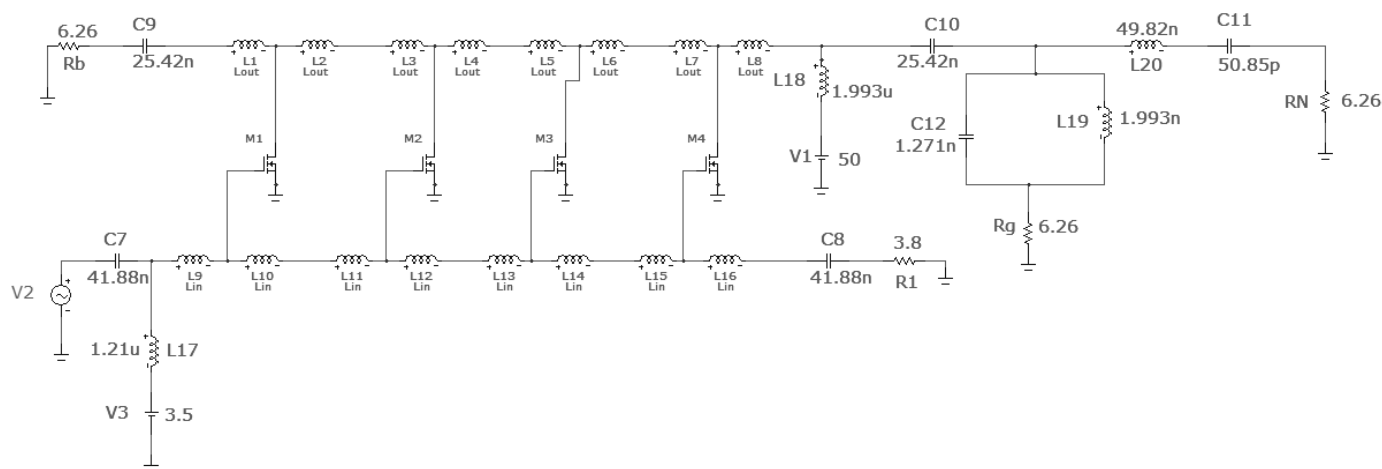


Рис. 1. Смоделированная схема УРУ на полевых транзисторах BLF 278

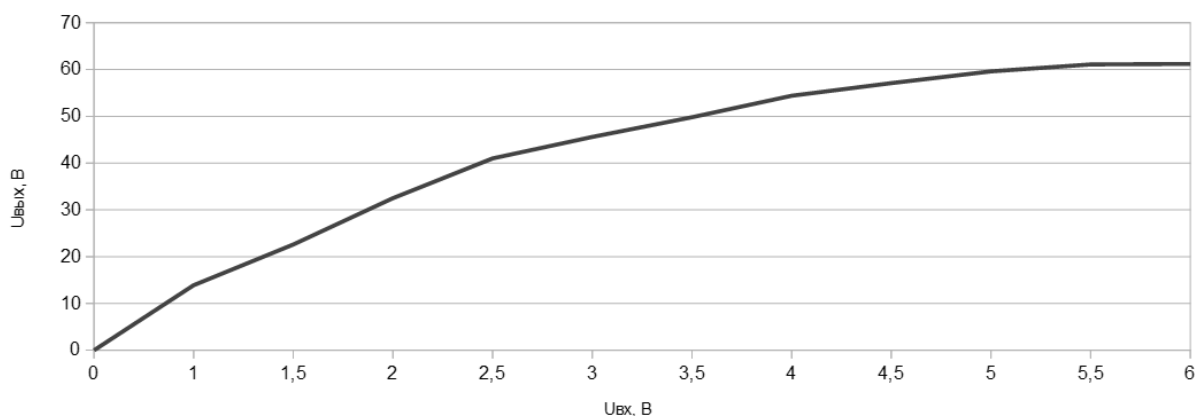


Рис. 2. Амплитудная характеристика УРУ

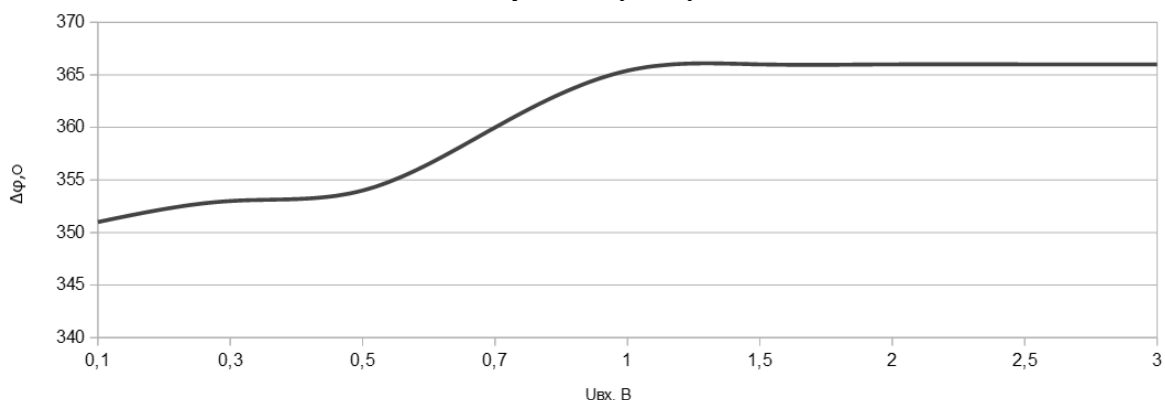


Рис. 3. Фазоамплитудная характеристика УРУ

Одной из причин существенных нелинейных искажений является существенная нелинейность вольт-фарадных характеристик транзисторов [3,4]. При различных амплитудах усиливаемого радиосигнала, из-за существенных изменений величин входной и выходной емкостей транзистора, обусловленных нелинейностями его вольт-фарадных характеристик, нарушается согласование искусственных длинных линий, что и приводит к дополнительным нелинейным искажениям, которые можно отнести к параметрическим. Такие параметрические искажения можно исследовать с помощью упрощенного моделирования усилителя с распределенным усилением, в котором в качестве активного элемента используются идеальные источники тока, управляемые напряжением. К таким идеальным источникам тока, параллельно входным и выходным клеммам, подключаются параметрические емкости. Нелинейность параметрических емкостей задается с помощью реальных вольт-фарадных характеристик транзисторов. Такая идеализация позволяет исключить нелинейные искажения, связанные с нелинейностями вольт-амперных характеристик транзистора, что существенно упрощает исследование [4]. Исследование влияния нелинейностей вольт-фарадных характеристик выходной и входной емкости осуществляется раздельно. Для этого построены две упрощенные модели усилителя с распределенным усилением, построенного на 4 источниках тока управляемых напряжением (см. рис. 4 и рис. 7).

В модели усилителя, представленной на рис.4, выходная емкость C_{out} задается полиномом шестой степени, полученным с помощью аппроксимирования значений вольт-фарадной характеристики выходной емкости полевого транзистора BLF 278. Вольт-фарадная характеристика моделировалась при изменении напряжения сток-исток. С помощью данной модели исследованы амплитудная и фазоамплитудная характеристики. Амплитудная характеристика представлена на рис. 5, а фазоамплитудная характеристика представлена на рис. 6.

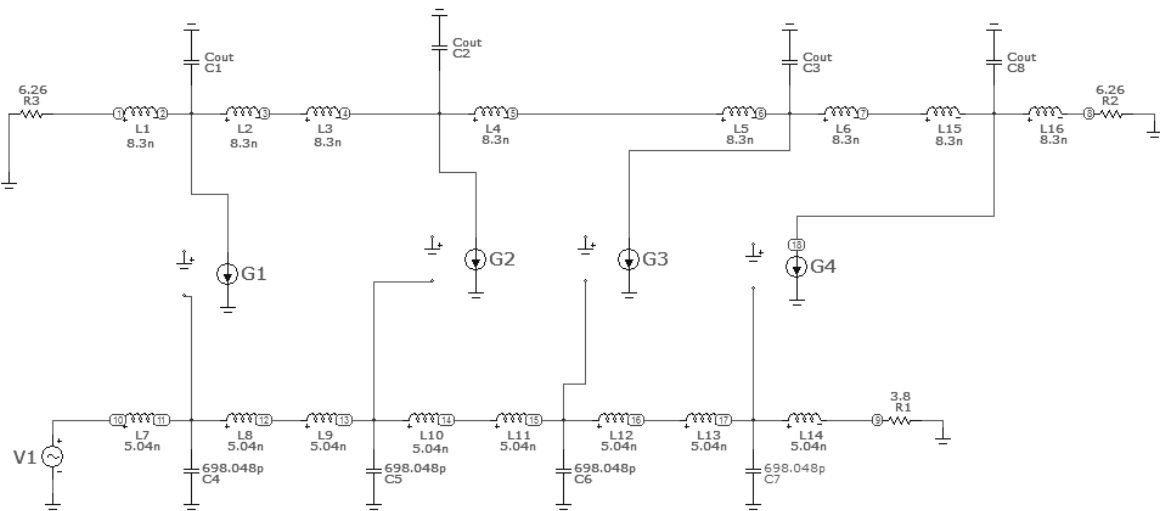


Рис. 4. Модель усилителя с распределенным усилением, построенная на источниках тока, управляемых напряжением, с параметрической выходной емкостью

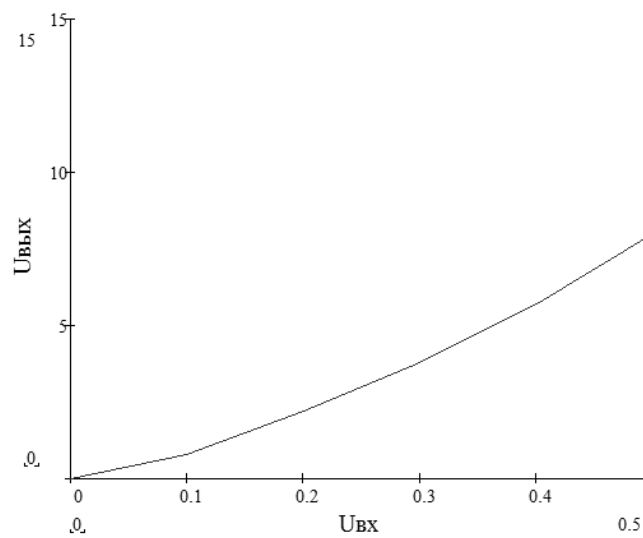


Рис. 5. Амплитудная характеристика

На рисунке 5 представлена только нелинейная часть амплитудной характеристики, поскольку при входных напряжениях выше 0.5 она приближается к линейной.

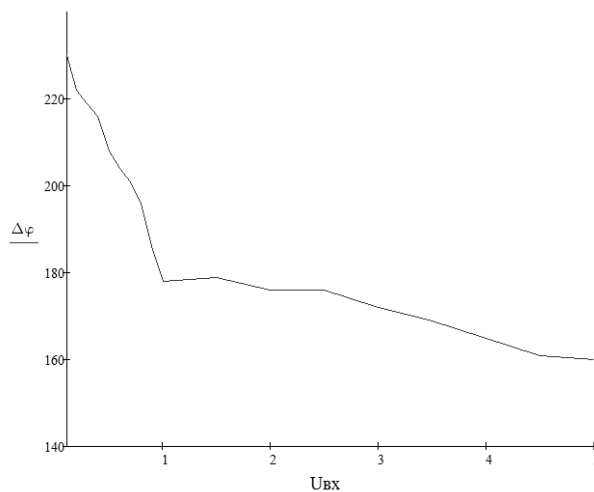


Рис. 6. Фазаамплитудная характеристика

Как видно из рис. 5 и 6, амплитудная характеристика имеет нелинейные участки, что характеризует исключительно параметрические искажения, вызываемые рассогласованием ИДЛ из-за нелинейности выходной емкости транзистора. Нелинейность фазоамплитудной характеристики обусловлена нелинейностью ВФХ выходной емкости полевого транзистора BLF278.

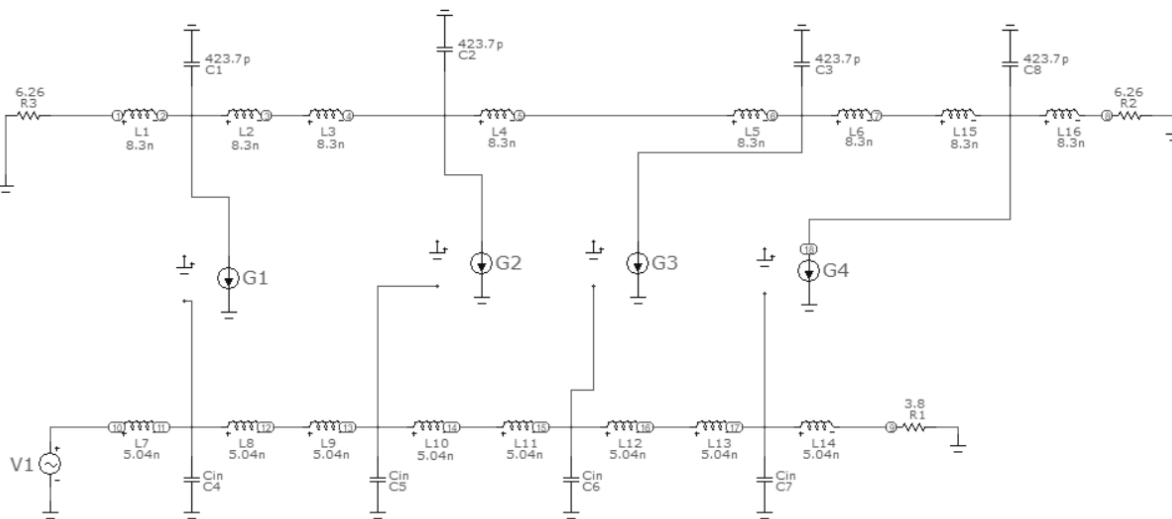


Рис. 7. Модель усилителя с распределенным усилением, построенная на источниках тока, управляемых напряжением, с параметрической входной емкостью

На рисунке 7 представлена упрощенная модель усилителя с распределенным усилением, также построенного на 4 источниках тока управляемых напряжением. В данной модели усилителя, входная емкость C_{in} задается полиномом четвертой степени, полученным с помощью аппроксимирования значений вольт-фарадной характеристики входной емкости полевого транзистора. Вольт-фарадная характеристика моделировалась при изменении напряжения затвор-исток. С помощью данной модели исследованы амплитудная и фазоамплитудная характеристики. Амплитудная характеристика представлена на рис. 8, а фазоамплитудная характеристика представлена на рис.9.

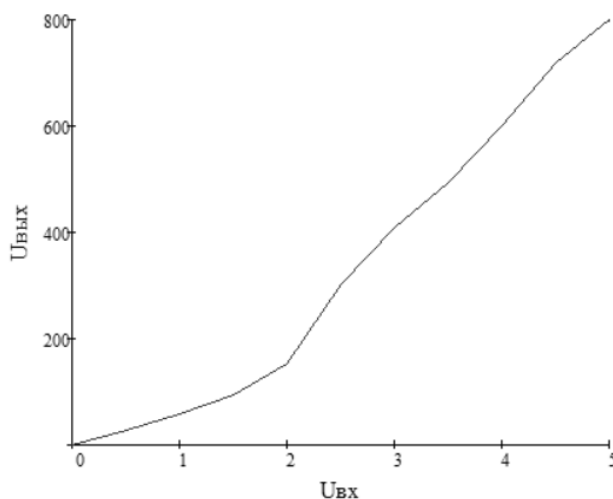


Рис. 8. Амплитудная характеристика

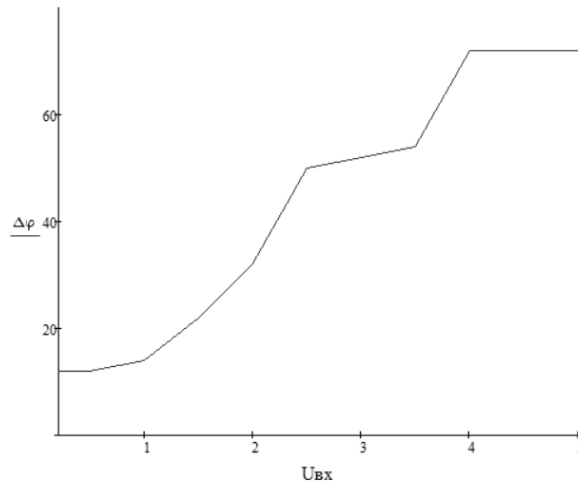


Рис. 9. Фазоамплитудная характеристика

Как видно из рис. 8 и 9, амплитудная характеристика также, как и в предыдущем случае имеет нелинейные участки, вызываемые рассогласованием ИДЛ из-за нелинейности входной емкости транзистора. Нелинейность фазоамплитудной характеристики обусловлена нелинейностью ВФХ входной емкости полевого транзистора

Заключение

Исследованные нелинейные искажения УРУ, вызываемые параметрическими входными и выходными емкостями транзисторов, входящими в звенья ИДЛ, показывают необходимость применения мер по линейризации УРУ, при усилении радиосигналов с непостоянной огибающей. Линейризацию усилителей мощности с распределенным усилением целесообразно выполнять с помощью методов предкоррекции.

Литература

1. Шмаков Н.Д., Иванюшкин Р.Ю. Моделирование усилителя с распределенным усилением на полевых транзисторах BLF278 для диапазонов ВЧ и ОВЧ // Т-Comm: Телекоммуникации и транспорт. 2017. Т. 11. № 11. С. 34-39.
2. 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).
3. Nikita D. Shmakov, Roman Yu. Ivanyushkin, Kirill O. Razin Perspective ways of construction of the tract of amplification of transmitters for digital television broadcasting // IEEE Conference Publication: 2018 Systems of Signal Synchronization, Generating and Processing in Telecommunication (SYNCHROINFO).
4. Nikita D. Shmakov, Roman Yu. Ivanyushkin. Research of nonlinear distortions in distributed amplifiers based on field-effect transistors // IEEE Conference Publication: 2019 Systems of Signal Synchronization, Generating and Processing in Telecommunication (SYNCHROINFO).
5. Шмаков Н.Д., Иванюшкин Р.Ю. Применение технологии распределенного усиления при построении широкодиапазонных и сверхширокодиапазонных радиочастотных усилителей мощности // Телекоммуникации и информационные технологии, 2017, Т.4, № 2, с.46-50.
6. Иванюшкин Р.Ю., Разин К.О., Шмаков Н.Д. Перспективные пути построения тракта усиления мощности передатчиков эфирного цифрового телевизионного вещания // Системы синхронизации, формирования и обработки сигналов. 2018. Т. 9. № 3. С. 96-103.
7. Шмаков Н.Д., Иванюшкин Р.Ю. Исследование твердотельных усилителей бегущей волны диапазона ОВЧ при двух подходах к их расчету // Системы синхронизации, формирования и обработки сигналов. 2017. Т. 8. № 3. С. 88-92.
8. Шмаков Н.Д., Иванюшкин Р.Ю. Компьютерное моделирование усилителя с распределенным усилением на полевых транзисторах // Телекоммуникации и информационные технологии. 2017. Т. 4. № 1. С. 30-33.

9. *Иванюшкин Р.Ю., Юрьев О.А.* Перспективы применения ключевых усилителей мощности классов D и DE при построении радиовещательных передатчиков диапазона ОВЧ // Т-Comm: Телекоммуникации и транспорт. 2016. Т. 10. № 5. С. 21-26.
10. *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.
11. *Берман А.А., Иванюшкин Р.Ю.* Проблематика построения тракта огибающей передатчиков с полярной архитектурой для цифрового радиовещания в диапазоне ОВЧ // Телекоммуникации и информационные технологии. 2016. Т. 3. № 2. С. 47-48.
12. *Шмаков Н.Д., Иванюшкин Р.Ю.* Гибридный телевизионный передатчик для цифрового и аналогового вещания // Телекоммуникации и информационные технологии. 2015. Т. 2. № 1. С. 30-33
13. *Юрьев О.А., Иванюшкин Р.Ю.* Способы построения РЧ-тракта передатчика цифрового радиовещания диапазона ОВЧ // Телекоммуникации и информационные технологии. 2014. Т. 1. № 1. С. 8-11.
14. *Иванюшкин Р.Ю., Дулов И.В., Овчинникова М.В., Тришина Ю.А.* История и перспективы применения метода автоматической регулировки режима для повышения КПД радиопередатчиков // Т-Comm: Телекоммуникации и транспорт. 2012. Т. 6. № 9. С. 66-67.
15. *Дулов И.В., Иванюшкин Р.Ю.* Нелинейная АРР по питанию для усилителя мощности передатчика цифрового радиовещания // Т-Comm: Телекоммуникации и транспорт. 2012. Т. 6. № 9. С. 59-63.
16. *Шахгильдян В.В., Иванюшкин Р.Ю.* Методы повышения энергетической эффективности линейных усилителей мощности // Т-Comm: Телекоммуникации и транспорт. 2011. Т. 5. № 9. С. 143-145.
17. *Иванюшкин Р.Ю., Юрьев О.А.* Проблематика построения РЧ-тракта передатчиков цифрового радиовещания диапазона ОВЧ на основе метода Л. Кана // Т-Comm: Телекоммуникации и транспорт. 2013. Т. 7. № 9. С. 91-93

RESEARCH OF PARAMETRIC NONLINEAR DISTORTIONS OF DISTRIBUTED AMPLIFIERS

Nikita D. Shmakov

Postgraduate student, MTUCI

Shmaki-shmak@yandex.ru

Roman Yu. Ivanyushkin

MTUCI, PhD., associate professor of ROS department

rivanyushkin@gmail.com

Key words: *Distributed amplifier, nonlinear distortions, artificial long lines, AM-AM characteristic, AM-PM characteristic.*

The reasons for the occurrence of parametric nonlinear distortion in distributed amplifiers built on field-effect transistors are discussed. The scheme of distributed amplifiers for research parametric nonlinear distortions are shown. The results of the research of nonlinear distortions on a computer model of a distributed amplifier based on voltage controlled current sources are presented.

МНОГОСТАНЦИОННЫЙ ДОСТУП В СЕТЯХ СИСТЕМЫ 5G

Овчинников Владислав Игоревич
студент группы MPT 1801 МТУСИ
vlad0518@yandex.ru

Логвинов Василий Васильевич
МТУСИ, к.т.н., доцент кафедры РОС
adlerbasil@rambler.ru

Ключевые слова: Многостанционный доступ в сетях LTE, LTE-Advanced, 5G, радиоблоки, ресурсные блоки, OFDM символы, циклический префикс, ортогональные поднесущие, многочастотная передача, технологии OFDM, F-OFDM, UFMС, FBMC

Приведены результаты сравнительного анализа вероятностных характеристик технологии ортогонального мультиплексирования с частотным разделением каналов (OFDMA), используемой в системах 4-го поколения LTE, LTE-Advanced и разновидностями технологий неортогонального мультиплексирования с частотным разделением каналов, предлагаемых для системы пятого поколения 5G, в среде MATLAB/Simulink. Анализ выполнен с применением имитационного моделирования и программных методов в частотной и временной области для условия идентичности радиоресурса, выделенного системам, и отношения С/Ш.

Активно растущий спрос на расширение перечня услуг, предоставляемых инфо/коммуникационными системами, от простейших - от передачи речи, до передачи данных, управляющих высокоскоростными процессами в режиме реального времени, привели к усложнению их структур и включению функции саморегулирования и оптимизации. Другое направление их совершенствования – создание глобальных коммуникационных систем, начавшееся с построения системы для передачи речи (GSM), получило свое развитие с внедрением универсальной системы широкополосного доступа (UMTS) и LTE.

Система поколения 5G является прорывной, в решении указанных задач, обеспечивая значительное увеличение скорости передачи и существенное повышение эффективности системы с меньшими временными задержками, чем в других системах. Использование диапазона частот от низких (несколько сотен МГц) до очень высоких (возможно до крайне высоких частот), позволит при блочной структуре радиокадра передавать огромный массив данных одновременно в нескольких блоках. Такие возможности заложены в новых технологиях многостанционного доступа к каналу, как развитие прогрессивной технологии OFDM, широко применяемой в сетях беспроводной связи WWAN 4-го поколения LTE, локальных сетях WLAN, WiFi, Hiperlan городских сетях радиодоступа WiMAX, персональных сетях Blue Tooth, Ultra Wide Band, Sensor Networks, так и изменившие ее идеологию. Благодаря этому формируемая глобальная сеть 5G будет включать перечисленные сети, обеспечивая бесконфликтную среду взаимодействия.

В работе проводится сравнительный анализ наиболее востребованной технологии широкополосного доступа с новыми, предлагаемыми к использованию в системе 5G (F-OFDM, FBMC, UFMС). Предлагаемые принципы формирования структуры радиоканала являются развитием базовой технологии OFDM или используют новые подходы к увеличению помехоустойчивости передаваемых информационных потоков. Анализ выполняется для системы LTE Advanced с использованием имитационной модели приема/передающего тракта мобильного терминала или в программными средствами в среде MATLAB, для идентичных условий (отношения сигнал/шум, ширине занимаемой полосы, структуры кадра).

Внедрение новых алгоритмов формирования информационных потоков, кроме достижения высокой скорости передачи данных, позволит обеспечить равномерное покрытие зоны обслуживания на основе гетерогенной структуры сети первичного и дополнительного спектров. Это предполагает применение многоуровневой сетевой структуры, использующей пико- и фемтосоты внутри помещения или в местах скопления мобильных абонентов и, работающих в высокочастот-

ной части выделенного спектра с высокой скоростью потока данных, с переходом в низкочастотный поддиапазон, обеспечивающий значительно большие размеры соты (до 100 км). АТ такой соты БС пикосот, образующих фантомную сеть внутри большой соты, остаются невидимыми.

В сети пятого поколения получил развитие принцип, работающий в сети *LTE-Advanced* – агрегация несущих [1]. В его основе лежит алгоритм, обеспечивающий обмен трафиком между АТ и БС в высокочастотных сетях, обладающих большим выделенным ресурсом. В более крупной сети низкочастотного диапазона взаимодействие между АТ и БС происходит только на уровне команд и служебной информации. Это позволяет снизить напряженность трафика в этом диапазоне частот, сохранив обмен данными между конечными устройствами, соответствующих каналному и сетевому уровням модели взаимодействия открытых сетей *OSI* [2]. Такая архитектура сети *5G* позволяет одновременно с увеличением скорости трафика решить задачу выравнивания спектральной плотности мощности принимаемого сигнала в зоне покрытия каждой соты и между фемто- и пикосотами.

Описанное многообразие и многослойность архитектуры сети при значительном различии в скоростях передачи данных указывает на возможность применения различных технологий доступа в *5G* на выделенных уровнях. В статье публикуются качественные и количественные оценки их сравнительного анализа и формулируются рекомендации для их применения.

Технология многостанционного доступа OFDM

Оценка влияния технологии многостанционного доступа *OFDM* на основные показатели канала при использовании квадратурной модуляции *4QAM* выполнялось в среде *Matlab/Simulink* методом операционной имитации (рис.1)

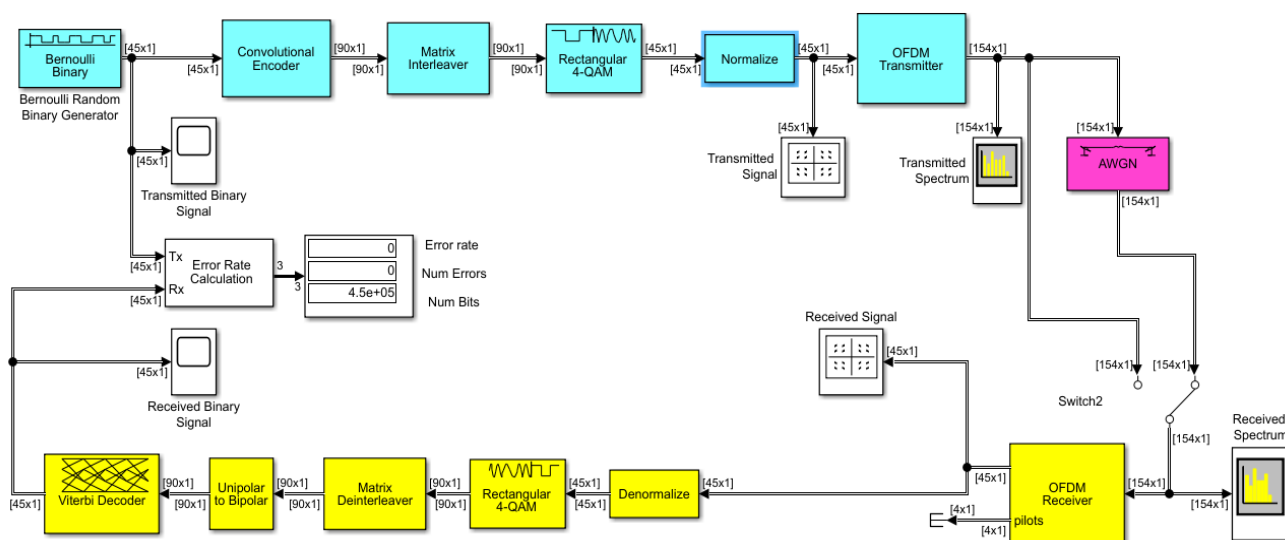


Рис.1 Функциональная схема исследуемой системы.

BernoulliBinary – блок, который генерирует псевдослучайную последовательность символов из алфавита $\{0, 1\}$ с вероятностью появления нуля $\rho_0 = 0.5$ и вероятностью появления единицы $\rho_1 = 0.5$.

Период выдачи одного бита $SampleTime = 1.e-6/45$ при двух битах информации, переносимой одним символом (для *4QAM*).

ConvolutionEncoder – сверточный кодер со скоростью кодирования $R=1/2$ и длиной кодового ограничения равной $Trellisstructure = 7$.

MatrixInterleaver – матричный перемежитель, использующий матрицу размером 15×6 .

Rectangular QAM Modulator Baseband – сигнальный кодер с параметром M -arynumber = 4, определяющим число точек сигнального созвездия. Параметр *Normalizationmethod* определяет способ масштабирования сигнального созвездия. Тип входного сигнала задается с помощью пара-

метра *Inputtype* значение *Bit*. Выходной сигнал задается с помощью параметра *Outputdatatype*, используется значение *double*

Normilize – подсистема, выполняющая функцию нормировочного множителя. для получения одинаковой средней мощности сигнала при различных видах сигнального кодирования и составляет для *4QAM*. Блок **Gain** = $1/\sqrt{2}$, входящий в подсистему задает значение нормировочного множителя; **Multiplication** – задает вид умножения.

OFDM Transmitter – подсистема, в которой моделируется процесс формирования отсчетов комплексной огибающей OFDM символа, включающая блоки: **Selector**, **Matrix Concatenation**, **Zero Pad**, **IFFT**, **Add Cyclic Prefix**, **PN Sequence Generator**, **Unipolar to Bipolar Converter**. Блок **Selector** выбирает из поступающего блока КАМ символов 6 векторов и подает каждый вектор на свой выход. **Matrix Concatenation** выполняет объединение выходных векторов блока **Selector** в матрицу с добавлением символов, которые будут передаваться на пилотных поднесущих и добавление нуля на основной несущей частоте. Блок **ZeroPad** увеличивает длину комплексного вектора, поступающего на его вход, путем добавления к нему нулевых элементов. Нулевые элементы добавляются в конец вектора. IFFT – блок ОБПФ, который формирует отсчеты комплексной огибающей OFDM-символа. Завершающим этапом в получении отсчетов комплексной огибающей OFDM символа является вставка отсчетов защитного интервала. Для этого блок **AddCyclicPrefix** копирует последние элементы комплексного вектора и вставляет их в начало, формируя N – отсчетов комплексной огибающей OFDM символов. Символы, передаваемые на поднесущих пилот сигналов, формируются с помощью блоков псевдослучайных последовательностей с помощью регистра сдвига.

На входе канала может включаться генератор гауссовского шума с нормальным распределением **AWGN**, обладающий параметрами: **Mode** – определяет способ задания уровня шума, **Inputsignalpowe** = 1 задает мощность сигнала.

OFDM Receiver – подсистема, в которой моделируется процесс выделения OFDM - символов и формирования отсчетов комплексной огибающей принятого сигнала. Подсистема состоит из следующих блоков: **RemoveCyclicPrefix** (удаление циклического префикса), **FFT** (процедура прямого дискретного преобразования Фурье), **Removezero-padding and reorder**, **SelectRows** (удаление нулей, сопряжение и формирование символьной последовательности, восстановление последовательной структуры). Данные блоки выполняют операции обратные операциям, выполняемым блоками подсистемы OFDM Transmitter. **Denormilize** - подсистема, выполняющая функцию обратную для подсистемы **Normilize**.

Rectangular QAM Demodulator Baseband – сигнальный декодер, формирующий цифровую последовательность принятого сигнала в основной полосе частот.

MatrixDeinterleaver – матричный деперемежитель, восстанавливающий исходную последовательность элементов во входном массиве, действуя обратным образом по отношению к блоку **MatrixInterleaver**.

Unipolartobipolarconverter – преобразователь униполярной входной последовательности к биполярному виду, для последующей обработки.

ViterbiDecoder – декодер Витерби восстанавливает по принятой на фоне шумов последовательности переданную, используя алгоритм максимального правдоподобия.

Для оценки свойств передаваемого и принимаемого сигналов используются дополнительные блоки (рис.1), включаемые на входе OFDM Transmitter и выходе OFDM Receiver позволяют наблюдать сигнальные созвездия на входе и выходе OFDM блоков. Нормированные спектральные характеристики отображаются на анализаторах спектров, включенных на выходе передатчика (**Transmitted Spectrum**) и входе приемника (**Receiver Spectrum**). Это позволяет оценивать ис-

кажения сигнала, возникающие в канале для различных значений отношения сигнал/шум на входе приемника.

Блок **ErrorRateCalculation** позволяет оценить вероятность ошибки в принятом сигнале (BER). Результаты блока выводятся на дисплей – **Display**. Он показывает общее число принятых бит, число ошибочных битов и оценку вероятности появления ошибочного бита.

Оценка показателей перспективных методов многостанционного доступа

Наиболее распространенная в настоящее время технология *OFDM* обладает рядом недостатков, осложняющих ее применение в системе *5G*: потеря спектральной эффективности, вызванное высоким уровнем боковых составляющих; высокие требования к синхронизации в системе. Если в *LTE* при работе с полосой 20 МГц, используются 100 ресурсных блоков по 12 поднесущих в каждом, а при их разнесении в 15 кГц эффективная полоса частот будет составлять только 18 МГц. Во временном домене системы *LTE* длина циклического префикса, разделяющего *OFDM* символы, составляет 114 или 160 элементов в каждом фрейме. Это снижает скорость информационного потока $\approx 7\%$. Таким образом, общие потери составляют $\approx 17\%$.

F-OFDM. В настоящее время предлагается несколько методов многостанционного доступа, решающих основные задачи, возникшие в процессе работы над стандартом для систем 5 –го поколения. Развитием метода ортогонального мультиплексирования с частотным разделением каналов (*OFDM*) явился метод **F-OFDM (Filtered Orthogonal Division Multiplexing)** фильтрового ортогонального мультиплексирования с частотным разделением каналов. В основе этого метода лежит требование сохранения ортогональности между поднесущими, сгруппированными в некоторой полосе частот, объединенной характером обслуживания. Это позволяет применять общую фильтрацию всех поднесущих сформированной подполосы. Это обеспечивает высокие показатели на разделение между поднесущими внутри группы при снижении требований на разделение между подполосами в частотной области. Благодаря этому уменьшается длина защитных интервалов, что повышает спектральную эффективность системы. Примеры энергетических спектров, полученных программными средствами, для классической *OFDM* и фильтровой *F-OFDM* приведены на (рис. 2 а, б) для одинакового отношения С/Ш = 15 дБ и ширины полосы частот.

Использование фильтровой *F-OFDM* обеспечивает значительное снижение изменения спектральной плотности мощности сигнала в рабочей полосе частот (примерно 20 дБ), который в классической *OFDM* превышает 40 дБ и существенное ослабление боковых излучений. Это указывает на высокую энергетическую эффективность фильтровой технологии.

FBMC. Вариантом совершенствования технологии *OFDM* является **FBMC (Filter Bank Multi Carrier)** – многочастотная передача с гребенчатой фильтрацией, когда каждая поднесущая фильтруется отдельно. Это снижает уровень внеполосных излучений и увеличивает устойчивость к интерференции между поднесущими. Такая технология обладает более высокой спектральной эффективностью по сравнению с *OFDM* (рис.2а и рис.3.а), но необходимость фильтрации каждой поднесущей для всей выделенной полосы частот увеличивает время задержки на обработку принятого сигнала, например, по сравнению с технологией **UFMC (Universal Filtered Multi-Carrier)**. Вероятно, что **FBMC** окажется весьма востребованной с внедрением квадратурной модуляции со сдвигом *OQAM*, особенно при использовании многоантенной технологии пространственного временного кодирования и структуры *MIMO*.

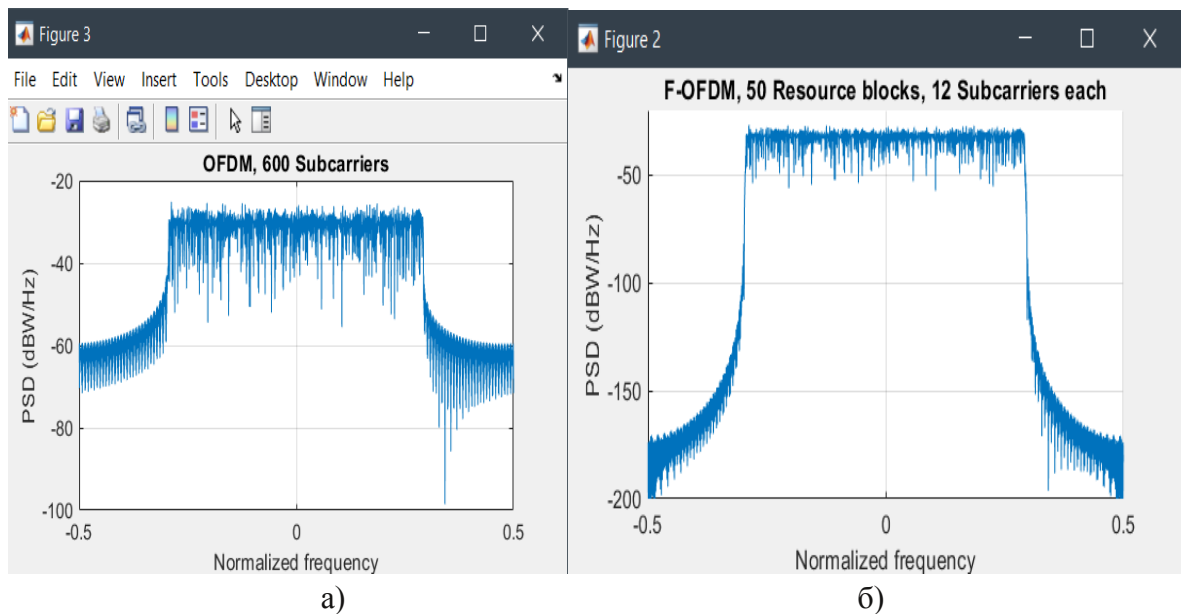


Рис.2 Спектр *F-OFDM* (50 ресурсных блоков, 12 поднесущих в каждом) в сравнении с *OFDM* (600 поднесущих). При отношении сигнал/шум $SNR = 15$ дБ.

Технология *UFMC* является результатом совмещения технологий *OFDM* и *FBMC*. Если в технологии *OFDM* фильтруется выделенная полоса частот, а в *FBMC* – каждая поднесущая, то при использовании *UFMC* – группы подполос, содержащие группы поднесущих. Общее число поднесущих N делится на подполосы, каждая из которой включает фиксированное число поднесущих. Некоторые из них не используются (как и в *OFDM*). Каждая из поднесущих из подполосы фильтруется фильтром длиной L , а отклики на выходе фильтра, выделенного подполосе, суммируются. Это снижает ширину занимаемого спектра на передающей стороне, а соответственно, возрастает спектральная эффективность такой технологии (рис.3а и рис.3б).

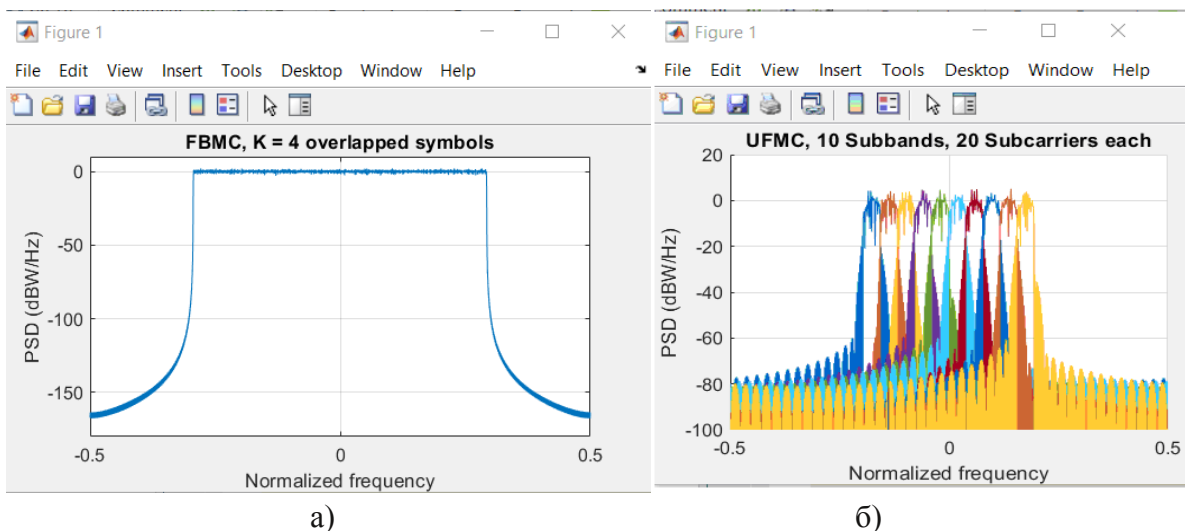


Рис.3 Спектр *FBMC* в сравнении с *UFMC* (10 подполос, 20 поднесущих в каждой). При отношении сигнал/шум $SNR = 15$ дБ.

При одинаковой полосе частот (для *FBMC* и *UFMC*) число точек преобразования Фурье для *FBMC* будет в два раза превышать их число в технологии *UFMC*, что одновременно уменьшит на порядок длину фильтра. Это упрощает его реализацию и сокращает время обработки сиг-

нала. Важным фактором, влияющим на длину КИХ-фильтра, является выбор длины весовой функции окна (обычно Дольфа-Чебышева), примерно равного длине циклического префикса для сигнала *OFDM*.

На рис.4 приведены сигнальные созвездия для рассмотренных ранее технологий радиодоступа к каналу при использовании квадратурной модуляции (*QAM*) с отношением $C/\text{Ш}$ = 15 дБ.

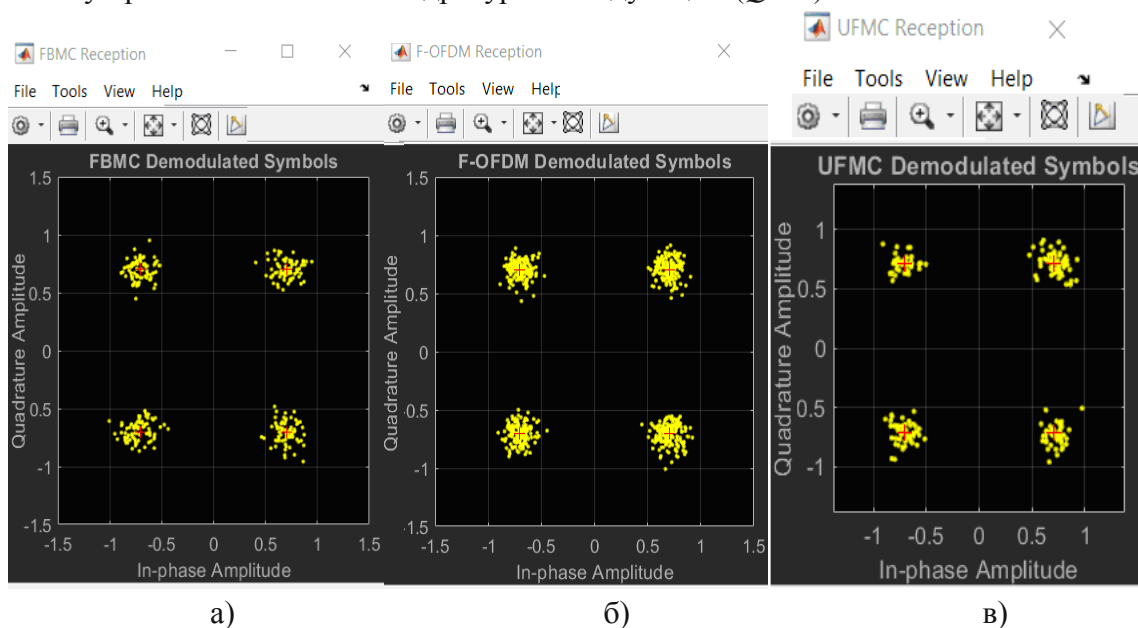


Рис.4 Сигнальные созвездия F-OFDM (BER=0), FBMC (BER=0), UFMC (BER=0).

Уменьшение отношения сигнал/шум до значения $SNR = 5$ дБ показывает, что на сигнальном созвездии (рис.5), при использовании UFMC влияние шума заметно ниже по сравнению с другими технологиями.

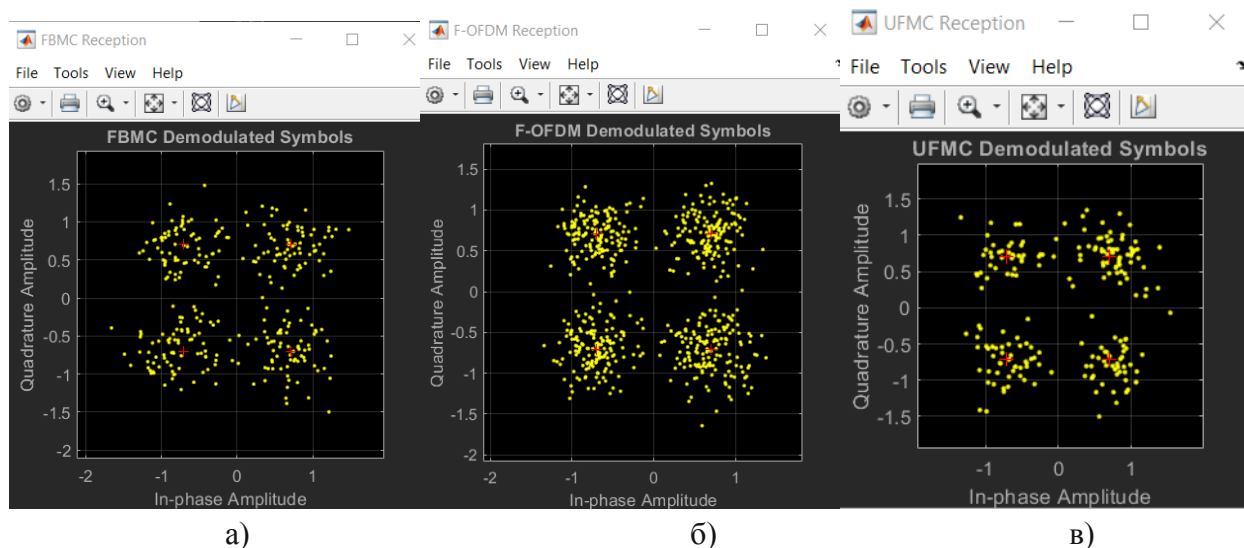


Рис.5 Сигнальные созвездия F-OFDM (BER=0.005), FBMC (BER=0.0099457), UFMC (BER=0.0025)

Для передачи коротких пакетов наилучшим вариантом будет использование фильтра с наименьшей длиной. Это приводит к потере ортогональности между подполосами с сохранением ортогональности между поднесущими внутри подполосы, но позволяет сократить длину поддиапазонных фильтров (значительно меньше CP в *OFDM*), что позволит повысить скорость переда-

чи. Ещё одним преимуществом будет являться применение более технологичной *QAM* модуляции, что вместе с технологией *MIMO* и *FFT* позволит формировать гибкую систему радиодоступа.

Заключение

Из моделирования в среде *Matlab* и последующего анализа видно, что использование технологий *FBMC* и *UFMC* в сети радиодоступа *5G* позволит повысить спектральную эффективность по сравнению с сетями, использующими технологию *OFDM*. При некотором усложнении технологии обработки сигнала в *UFMC* можно заметно повысить скорость передачи данных сокращением длины циклического префикса. Это приведет к потере ортогональности между подполосами, сохранив ортогональность внутри каждой из них. Компактность спектра в *UFMC* (рис.3б) указывает на существенное снижение внеполосных излучений, поэтому снижение взаимной ортогональности подполос практически не скажется на вероятности правильного приема. Одновременно сокращение временных задержек и увеличение скорости передачи данных в *UFMC* окажется предпочтительным в сотах малых размеров для повышения ее энергетической эффективности. Технологии *FBMC* может оказаться весьма эффективным способом борьбы с многолучевостью, особенно при организации сот большого радиуса со слабой застройкой.

Литература

1. 3GPP TS 36.101 v10.0.0; User Equipment (UE) radio transmission and reception.
2. В.В.Логвинов Приемники систем фиксированной и мобильной связи. – М.: СОЛОН-Пресс, 2017. – 816 с.: ил.
3. Wild, T., Schaich, F., Chen Y., "5G air interface design based on Universal Filtered (UF-)OFDM ", Proc. of 19th International Conf. on Digital Signal Processing, pp. 699-704, 2014.

MULTI-STATION ACCESS IN 5G SYSTEM NETWORKS

Vlad I. Ovchinnikov

Student of group MRT1801, MTUCI

vlad0518@yandex.ru

Vasily V. Logvinov

MTUCI, PhD., associate professor of ROS department

adlerbasil@rambler.ru

Key words: *LTE, LTE-Advanced, 5G multiple access networks, radio blocks, resource blocks, OFDM symbols, cyclic prefix, orthogonal subcarriers, multi-frequency transmission, OFDM technologies, F-OFDM, UFMC, FBMC*

Results of a comparative analysis of the probabilistic characteristics of orthogonal frequency division multiplexing (OFDMA) technology used in 4th generation LTE systems, advanced LTE systems and varieties of nonorthogonal frequency division multiplexing technologies proposed for fifth generation 5G system in MATLAB / Simulink The analysis was performed with using modeling methods and software in the frequency and time domain for the identity of the radio resource allocated to the system.

АЛГОРИТМЫ ПЕРЕДАЧИ ДАННЫХ В ИМИТАТОРЕ ОБЪЕДИНЕННОЙ СЕТИ ПД СПЕЦИАЛЬНОГО НАЗНАЧЕНИЯ

Басараб Михаил Алексеевич

*д.ф.-м.н., заведующий кафедрой ИБ МГТУ им. Н.Э.Баумана
bmic@mail.ru*

Бельфер Рувим Абрамович

*к.т.н., доцент кафедры ИБ МГТУ им. Н.Э.Баумана
a.belfer@yandex.ru,*

Кравцов Александр Владимирович

*начальник отдела НИИЦ (Москва) ЦНИИ ВВКО
skyak78@gmail.com*

Орлов Владимир Георгиевич

*к.т.н., доцент, МТУСИ
v.g.orlov@mtuci.ru*

Ключевые слова: имитатор объединенной сети, частная сеть, информационная безопасность, сеть передачи данных.

В статью включены материалы, являющиеся продолжением исследований по разработке алгоритмов функционирования учебного имитатора объединенной сети ПД специального назначения. Эти алгоритмы обеспечивают защищенную передачу пакета данных по установленному коммутируемому виртуальному каналу КВК в каждой из двух изолированных частных сетей ПД. В предлагаемых алгоритмах предусмотрено обеспечение высоких требований по информационной безопасности и надежности.

Введение

Настоящая статья является продолжением публикаций [1-5], в которых рассмотрены вопросы создания имитатора объединенной сети ПД категории специального назначения. Работы по созданию имитатора выполняются в рамках проектирования учебного лабораторного стенда (УЛС) на кафедре «Информационная безопасность» МГТУ им. Н.Э. Баумана совместно с Научно-исследовательским испытательным центром ЦНИИ Войск ВКО МО РФ. В упомянутых публикациях приводится разработанный алгоритм работы имитатора объединенной сети ПД на примере соединений КВК в двух разных частных сетях (ЧС1 и ЧС3) на участках удаленного абонентского доступа и в транспортной части сети. При этом предусмотрена защита от угроз информационной безопасности в соответствии с разработанными в [1] предложениями, а также параллельная передача сообщений по пучку маршрутов из четырех путей маршрутизации с целью обеспечения требований надежности. Данная статья посвящена разработке алгоритмов передачи пакета данных в имитаторе объединенной сети ПД специального назначения на примере установленных соединений КВК по алгоритмам, рассмотренным в [4,5]. Для сокращения объема статьи сформированные таблицы маршрутизации КВК по логическим адресам приводятся только для одного из четырех путей маршрутизации (первого). По этой же причине логические адреса в таблицах маршрутизации для частной сети ЧС3 приведены в скобках.

Конфигурация имитатора объединенной сети ПД

Приведем краткое описание приведенной на рис. 1 конфигурации имитатора объединенной сети ПД имитированной последствием УЛС, в котором предусмотрен обмен пакетами данных между оконечными пунктами КВК одновременно по четырем путям маршрутизации. Каждый путь маршрутизации состоит из трех центров коммутации пакетов (ЦКП). К двум граничным ЦКП абонентского доступа в каждом пути маршрутизации (ЦКП 1.1 – адрес 11, ЦКП 3.1 – адрес 31 и ЦКП

1.2 – адрес 12, ЦКП 3.2 – адрес 32) подключены оконечные пункты, а ЦКП 2.1 – адрес 21 и ЦКП 2.2 – адрес 22 являются транзитными (или транспортными). Доступ оконечных пунктов (a, b, \dots, c и d, e, \dots, f) к граничным центрам коммутации пакетов удаленный.

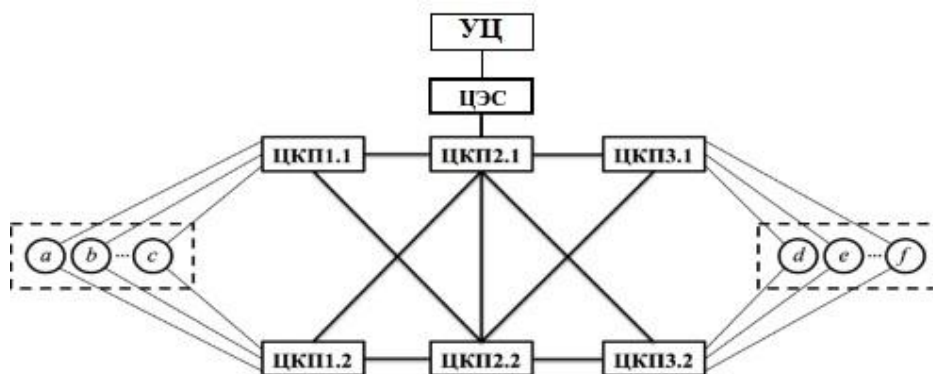


Рис. 1. Конфигурация имитатора объединенной сети ПД с центром эксплуатации сети и единым удостоверяющим центром

Примем, что оконечные пункты a (ОПа) и f (ОП f) принадлежат частной сети 1 (ЧС1); оконечные пункты b (ОП b) и d (ОП d) принадлежат частной сети 3 (ЧС3). Присвоим оконечным пунктам ЧС1 физические адреса от 1 до 999, ЧС3 – от 2001 до 2999. Присвоим физический адрес 101 для ОПа, 601 – для ОП f , 1601 – для ОП e , 2101 – для ОП b , 2601 – для ОП d .

Приведём описание укрупнённого алгоритма передачи пакетов данных в имитаторе объединенной сети ПД специального назначения на примере двух установленных КВК – одного ЧС1, а другого -ЧС3 [4,5]. Для отличия от ЧС1 операции в ЧС3 приведены в скобках. Канальные ключи и сквозные шифрования, алгоритмы шифрования и обеспечения целостности сообщений определены при установлении КВК [1,4,5]. Примем условие отсутствия ошибок в каналах связи при передаче пакетов данных. Алгоритм подтверждения приема пакета данных приводится только для примера КВК ЧС1.

Описание передачи пакета данных от оконечного пункта источника установления КВК в ЧС1 и ЧС3

Маршрутизация при установлении КВК и передача пакета данных от оконечного пункта ОПа источника установления КВК в ЧС1 (ОП b в ЧС3) выполняется по пучку маршрутов из одинаковых для ЧС1 и ЧС3 четырех путей маршрутизации: из ЦКП 1.1 по первому пути (11-21-31) и третьему пути (11-22-31), из ЦКП 1.2 по второму пути (12-22-32) и четвертому пути (12-21-32). В [4,5] приведены полные таблицы маршрутизации для всех этих путей. Для описания алгоритма в настоящей статье в таблицах 1-3 приведены строки не полных таблиц маршрутизации в ЦКП, а только первого пути маршрутизации КВК ЧС1 и КВК ЧС3.

Таблица 1

Таблица маршрутизации КВК ЧС1 и ЧС3 по логическим адресам в ЦКП 1.1

| Входящее сообщение в ЦКП1.1 | | | Исходящее сообщение из ЦКП1.1 | | |
|------------------------------|------------|------|-----------------------------------|------------|------|
| Адр. источн. сообщ. в ЦКП1.1 | LCN | Освн | Адрес поступлен. сообщ. из ЦКП1.1 | LCN | Освн |
| 101(2101) | 809 (2809) | да | 21 | 802 (2802) | да |
| 21 | 802 (2802) | да | 101 | 809 (2809) | да |

Таблица 2

Таблица маршрутизации КВК ЧС1 и ЧС3 по логическим адресам в ЦКП 2.1

| Входящее сообщение в ЦКП2.1 | | | Исходящее сообщение из ЦКП2.1 | | |
|------------------------------|------------|------|---------------------------------------|------------|------|
| Адр. источн. сообщ. в ЦКП2.1 | LCN | Освн | Адрес поступления сообщения из ЦКП2.1 | LCN | Освн |
| 11 | 802 (2802) | нет | 31 | 924 (2924) | да |
| 31 | 924 (2924) | да | 11 | 802 (2802) | нет |

Таблица 3

Таблица маршрутизации КВК ЧС1 и ЧС3 по логическим адресам в ЦКП3.1

| Входящее сообщение в ЦКП3.1 | | | Исходящее сообщение из ЦКП3.1 | | |
|------------------------------|------------|------|--|------------|------|
| Адр. источн. сообщ. в ЦКП3.1 | LCN | Освн | Адрес поступления. сообщения из ЦКП3.1 | LCN | Освн |
| 21 | 924 (2924) | нет | 601 (2601) | 714 (2714) | да |
| 601 (2601) | 714 (2714) | да | 21 | 924 (2924) | нет |

Приведем разработанный алгоритм передачи пакета данных от оконечного пункта ОП_а в ЧС1 (ОП_б в ЧС3), который был источником установления КВК:

1. Сформировать в ОП_а для ЧС1 (в ОП_б для ЧС3) четыре пакета данных для всех четырех путей маршрутизации КВК, включающие одинаковую информационную часть и разные заголовки. Общими в заголовках является тип сообщения и полученный из верхней строки табл.1 LCN=809 для ЧС1 (в ЧС3 LCN=2809). Заголовки этих пакетов данных КВК различаются входящим в него номером пути маршрутизации.

2. Зашифровать в ОП_а для ЧС1 (в ОП_б для ЧС3) сквозным ключом $K_{1_{101601}}$ ($K_{1_{21012601}}$ в ЧС3) информационную часть пакетов данных и выполнить сквозную процедуру обеспечения ее целостности сквозным ключом $K_{1_{101601}}$ ($K_{1_{21012601}}$ в ЧС3).

3. Зашифровать в ОП_а для ЧС1 (в ОП_б для ЧС3) заголовки всех пакетов данных КВК канальным ключом абонентского доступа источника соединения $K_{1_{01}}$ ($K_{2_{101}}$ в ЧС3). Отправить эти пакеты в ЦКП 1.1 и ЦКП 1.2.

4. Дешифровать пакеты данных в ЦКП 1.1 и ЦКП 1.2 ключом $K_{1_{01}}$ ($K_{2_{101}}$ в ЧС3).

5. Заменить в ЦКП 1.1 и ЦКП 1.2 в заголовках всех четырех пакетов данных КВК LCN=809 для ЧС1 (LCN=2809 для ЧС3) на полученные из верхних строк в части "Исходящее сообщение из ЦКП1.1" и "Исходящее сообщение из ЦКП1.2" полных таблиц маршрутизации КВК ЧС1 по логическим адресам в ЦКП 1.1 и в ЦКП 1.2. Как видно из табл.1 для первого пути маршрутизации КВК такой LCN=802 для ЧС1 (LCN=2802 для ЧС3).

6. Определить адреса поступления пакетов данных КВК ЧС1 и ЧС3 из ЦКП 1.1 и ЦКП 1.2. Как видно из табл.1 для первого пути маршрутизации КВК таким адресом из ЦКП 1.1 является 21, т.е. ЦКП 2.1. По остальным путям маршрутизации адреса указаны в полной таблице маршрутизации. В этих таблицах согласно приведенным в начале раздела данным такими адресами из ЦКП 1.1 для третьего пути является 22, для второго пути из ЦКП 1.2 – 22, для четвертого пути из ЦКП 1.2 – 21.

7. Зашифровать в ЦКП 1.1 и ЦКП 1.2 заголовки всех передаваемых пакетов данных КВК канальными ключами ЧС1 между этими ЦКП и транзитными ЦКП 2.1, ЦКП 2.2: $K_{1_{1121}}$, $K_{1_{1222}}$, $K_{1_{1122}}$, $K_{1_{1221}}$ (в ЧС3 канальными ключами – $K_{3_{1121}}$, $K_{3_{1222}}$, $K_{3_{1122}}$, $K_{3_{1221}}$). Эти ключи сформированы при установлении КВК и относятся к соответствующим смежным ЦКП. Ключ

$K1_{1121}$ (в ЧСЗ - $K3_{1121}$) относится к заголовку пакета данных первого пути КВК на участке 11-21.

8. Отправить пакеты данных КВК с зашифрованными заголовками из ЦКП 1.1 и ЦКП 1.2 в соответствующие смежные ЦКП 2.1 и ЦКП 2.2. Произвести дешифрацию заголовков принятых пакетов данных соответствующими канальными ключами.

9. Определить адреса поступления пакетов данных КВК ЧС1 и ЧС3 из ЦКП 2.1 и ЦКП 2.2. Как видно из табл.2 для первого пути маршрутизации КВК таким адресом из ЦКП 2.1 является 31, т.е. ЦКП 3.1. По остальным путям маршрутизации адреса указаны в полной таблице маршрутизации. В этих таблицах согласно приведенным в начале раздела данным такими адресами из ЦКП 2.1 для четвертого пути является 32, для второго пути из ЦКП 2.2 – 32, для третьего пути из ЦКП 2.2 – 31.

10. Зашифровать в ЦКП 2.1 и ЦКП 2.2 заголовки всех передаваемых пакетов данных КВК канальными ключами ЧС1 между этими ЦКП и граничные ЦКП 3.1, ЦКП 3.2: $K1_{2131}$, $K1_{2232}$, $K1_{2231}$, $K1_{2132}$ (в ЧСЗ канальными ключами – $K3_{2131}$, $K3_{2232}$, $K3_{2231}$, $K3_{2132}$). Эти ключи сформированы при установлении КВК и относятся к соответствующим смежным ЦКП. Ключ $K1_{2131}$ (в ЧСЗ – $K3_{2131}$) относится к заголовку пакета данных первого пути КВК на участке 21-31.

11. Отправить пакеты данных КВК с зашифрованными заголовками из ЦКП 2.1 и ЦКП 2.2 в соответствующие смежные ЦКП 3.1 и ЦКП 3.2. Произвести дешифрацию заголовков принятых пакетов данных соответствующими канальными ключами.

12. Определить по верхним строкам таблиц маршрутизации в части "Исходящее сообщение из ЦК 3.1" и "Исходящее сообщение из ЦКП 3.2" полных таблиц маршрутизации КВК ЧС1 по логическим адресам в ЦКП 3.1 и в ЦКП 3.2. адреса поступления пакетов данных КВК из ЦКП 3.1 и ЦКП 3.2. Как видно из табл. 3 для первого пути маршрутизации КВК таким адресом из ЦКП 3.1 является адрес 601, т.е. ОПf (в ЧСЗ, адрес 2601 т.е. ОПd). Согласно приведенному выше пучку маршрутизации такими адресами из ЦКП 3.1 и ЦКП 3.2 для всех четырех путей является 601 (2601 в ЧСЗ).

13. Зашифровать в ЦКП 3.1 и ЦКП 3.2 заголовки всех передаваемых пакетов данных КВК канальным ключом ЧС1 K_{601} между этими ЦКП и ОПf (в ЧСЗ K_{2601} между этими ЦКП и ОПd).

14. Отправить пакеты данных КВК из ЦКП 3.1 и ЦКП 3.2 по адресу 601 в ОПf (в ЧСЗ по адресу 2601 в ОПd). В ОПf (в ЧСЗ в дешифровать ключом ЧС1 K_{601} (в ЧСЗ K_{2601}) заголовки всех принятых с путей маршрутизации пакетов данных КВК. Дешифровать информационную часть этих пакетов данных ключом $K1_{101601}$ (в ЧСЗ $K1_{21012601}$).

15. В ОПf (в ЧСЗ в ОПd) сравнить LCN расшифрованных заголовков пакетов данных с LCN оконечного пункта ОПf (в ЧСЗ в ОПd), записанного при установлении КВК. Сравнить содержание информационной части принятых пакетов данных. По результатам анализа заголовка и информационной части принятых пакетов данных КВК принимается решение о достоверности принятого пакета данных. При успешном результате от ОПf в ОПa (в ЧСЗ от ОПb в ОПd) передается сообщение подтверждения приема пакета данных.

Описание алгоритма передачи сообщения подтверждения приема пакета данных в КВК ЧС1

1. Маршрутизация сообщений подтверждения ПТД приема пакета данных производится на основании нижних строк пути таблицы маршрутизации КВК. Сформировать в ОПf сообщения подтверждения ПТД по всем четырем путям маршрутизации приема пакета данных в КВК ЧС1, включающие тип сообщения, полученный из нижней строки пути табл.3, номер пути маршрутизации LCN=714. Зашифровать канальным ключом абонентского доступа ЧС1 K_{601} сообщения ПТД всех четырех путей маршрутизации. Отправить сообщения ПТД КВК ЧС1 в ЦКП 3.1 и ЦКП 3.2. Эти сообщения дешифруются в ЦКП 3.1 и ЦКП 3.2. канальным ключом ЧС1 K_{601} .

2. На основании части нижних строк таблиц маршрутизации КВК "Исходящее сообщение из ЦКП 3.1" и "Исходящее сообщение из ЦКП 3.2" для всех путей маршрутизации

определяем адрес ЦКП, по которым передаются сообщения ПТД каждого пути маршрутизации. Согласно табл.3 сообщение ПТД первого пути передается по адресу 21 (т.е. в ЦКП 2.1). В ЦКП 3.1 и ЦКП 3.2 на основании части нижних строк таблиц маршрутизации КВК "Исходящее сообщение из ЦКП 3.1" и "Исходящее сообщение из ЦКП 3.2" для всех путей маршрутизации определяем LCN, который заменяет LCN=714 в сообщениях ПТД. Согласно табл. 3 на основании этих же строк маршрутизации LCN=714 в сообщении ПТД первого пути маршрутизации заменяется на LCN=924. Сообщения ПТД шифруются в ЦКП 3.1 и ЦКП 3.2 соответствующим канальным ключом $K_{1_{2131}}$ (первого пути КВК между ЦКП 2.1 и ЦКП 3.1) $K_{1_{2231}}$ (третьего пути ЦКП 2.2 и ЦКП 3.2), $K_{1_{2232}}$ (второго пути между ЦКП 2.2 и ЦКП 3.2.), $K_{1_{2132}}$ (четвертого пути между ЦКП 2.1 и ЦКП 3.2.). Отправить эти сообщения в соответствующие ЦКП 2.1 и ЦКП 2.2. Эти сообщения дешифруются соответствующими канальными ключами ЧС1 между ЦКП 2.1, ЦКП 2.2 и ЦКП 3.1, ЦКП 3.2.

3. На основании части нижних строк таблиц маршрутизации КВК "Исходящее сообщение из ЦКП 2.1" и "Исходящее сообщение из ЦКП 2.2" для всех путей маршрутизации определяем адрес ЦКП, на которые должны передаваться сообщения ПТД каждого пути маршрутизации. Согласно табл.2 сообщение ПТД первого пути должны передаваться по адресу 11 (т.е. в ЦКП 1.1). В ЦКП 2.1 и ЦКП 2.2 на основании части нижних строк таблиц маршрутизации КВК "Исходящее сообщение из ЦКП 2.1" и "Исходящее сообщение из ЦКП 2.2" для всех путей маршрутизации определяем LCN, который заменяет LCN=924 в сообщениях ПТД. Согласно табл. 2 на основании этих же строк маршрутизации LCN=924 в сообщении ПТД первого пути маршрутизации заменяется на LCN=802. Сообщения ПТД шифруются в ЦКП 2.1 и ЦКП 2.2 соответствующим канальным ключом $K_{1_{1121}}$ (первого пути КВК между ЦКП 2.1 и ЦКП 1.1) $K_{1_{1122}}$ (третьего пути ЦКП 2.2 и ЦКП 1.1), $K_{1_{1222}}$ (второго пути между ЦКП 2.2 и ЦКП 1.1.), $K_{1_{1221}}$ (четвертого пути между ЦКП 2.1 и ЦКП 1.2). Отправить эти сообщения в соответствующие ЦКП 1.1 и ЦКП 1.2. Эти сообщения дешифруются соответствующими ключами.

4. На основании части нижних строк таблиц маршрутизации КВК "Исходящее сообщение из ЦКП 1.1" и "Исходящее сообщение из ЦКП 1.2" для всех путей маршрутизации определяем адрес окончного пункта пути маршрутизации, на которые должны передаваться сообщения ПТД каждого пути маршрутизации. Согласно табл.2 сообщение ПТД первого пути должны передаваться по адресу ОПa (т.е. 101). В ЦКП 1.1 и ЦКП 1.2 на основании части нижних строк таблиц маршрутизации КВК "Исходящее сообщение из ЦКП 1.1" и "Исходящее сообщение из ЦКП 1.2" для всех путей маршрутизации определяем LCN, который заменяет LCN=802 в сообщениях ПТД. Согласно табл. 1 на основании этих же строк маршрутизации LCN=802 в сообщении ПТД первого пути маршрутизации заменяется на LCN=809. Сообщения ПТД всех путей маршрутизации шифруются в ЦКП 1.1 и ЦКП 1.2 канальным ключом K_{101} . Отправить сообщения ПТД по всем путям маршрутизации в ОПa (т.е. по адресу 101). Принятые сообщения ПТД дешифруются канальным ключом K_{101} . Расшифрованные значения LCN=809 подтверждают установление КВК.

Заключение

Разработанные алгоритмы защищенной передачи пакета данных по установленному коммутируемому виртуальному каналу КВК в каждой из двух изолированных частных сетей ПД апробированы с использованием имитатора объединенной сети ПД и УЛС. Результаты испытаний показали возможность достижения требуемых показателей информационной безопасности и надежности.

Литература

1. *Басараб М.А., Бельфер Р.А., Кравцов А.В., Никулина Т.П.* Алгоритм аутентификации и формирования разовых ключей в имитаторе лабораторного стенда объединенной сети ПД специального назначения // Первая миля. №2. 2019. С. 62-68.
2. *Басараб М.А., Бельфер Р.А., Кравцов А.В.* Архитектура очередей таблиц маршрутизации объединенной сети ПД категории специального назначения // Электросвязь. № 2. 2019. С. 33-37.
3. *Басараб М.А., Бельфер Р.А., Кравцов А.В.* Маршрутизация в объединенном имитаторе сети ПД специального назначения между оконечными пунктами разных частных сетей // Электросвязь. № 3. 2019. С. 51-53.
4. *Басараб М.А., Бельфер Р.А., Глинская Е.В., Кравцов А.В.* Алгоритм защищенного установления соединения на абонентском доступе имитатора объединенной сети ПД специального назначения // Первая миля. №8. 2019 (в печати).
5. *Басараб М.А., Бельфер Р.А., Глинская Е.В.* Алгоритм защищенного установления соединения в транспортной части имитатора объединенной сети ПД специального назначения // Первая миля. №8. 2019 (в печати).

DATA TRANSMISSION ALGORITHM IN A SIMULATOR OF A COMBINED SPECIAL-PURPOSE DATA NETWORK

Mikhail A. Basarab

*Ph. D., head of the Department of Information Security, MSTU
bmic@mail.ru*

Ruvim. A. Belfer

*candidate of technical Sciences
docent of Information Security, MSTU
a.belfer @ yandex.ru*

Aleksandr. V. Kravtsov

*senior researcher RaTC
skyak78@gmail.com*

Vladimir. G. Orlov

MTUCI, PhD., associate professor of MTUCI

Keywords: *united network simulator, private network, information security, data transmission network.*

The article is a continuation of the development of algorithms for the functioning of a training simulator of a special purpose integrated data network. Algorithms have been developed for secure transmission of a data packet over the installed switched virtual channel of the in each of two isolated private networks. The proposed algorithms provide for ensuring high requirements for information security and reliability.

РАЗРАБОТКА ЛАБОРАТОРНЫХ РАБОТ ПО ИЗУЧЕНИЮ И ИССЛЕДОВАНИЮ МУЛЬТИСЕРВИСНЫХ СЕТЕЙ СВЯЗИ

Канищева Маргарита Геннадьевна

студент группы М61902 МТУСИ

margo.kan@list.ru

Железов Денис Борисович

инженер, ISKRATEL

denis.zhelezka@mail.ru

Маликова Елена Егоровна

МТУСИ, к.т.н., доцент кафедры СС и СК

emalikova@gmail.com

Ключевые слова: мультисервисные сети связи, IP-телефония, лабораторный комплекс, IP-АТС, дополнительные виды обслуживания, Call-центр.

Лабораторный комплекс находится на кафедре Сети связи и системы коммутации (СС и СК). На базе данного комплекса было создано пять лабораторных работ, которые охватывают теоретические вопросы и практические задачи, решаемые инженерами-связистами в процессе эксплуатации телефонных сетей. Эти работы выполняются студентами по направлению подготовки 11.04.02 - Инфокоммуникационные технологии и системы связи. Данная статья посвящена описанию лабораторных работ, в которых изучаются дополнительные виды обслуживания абонентов.

Введение

На сегодняшний день наиболее актуальной темой в инфокоммуникациях является применение технологий виртуализации [1] для удовлетворения запросов клиентов, а также расширения спектра предоставляемых услуг и снижения стоимости телекоммуникационного оборудования. В то же время возрастает потребность в предоставлении различных дополнительных услуг пользователям, а, следовательно, повышаются требования к инженерам, которые должны обслуживать телекоммуникационное оборудование [4-9]. В связи с этим в МТУСИ на базе кафедры СС и СК был организован лабораторный комплекс по изучению мультисервисных сетей, который используется для получения студентами практических навыков работы с различными типами оборудования [10-12].

Лабораторный комплекс

Схема лабораторного комплекса по изучению мультисервисных сетей связи представлена на рисунке 1. В лабораторном комплексе имеется оборудование, которое расположено в нескольких лабораториях кафедры СС и СК. Так, в Л-402 находится лабораторный комплекс по изучению виртуальной АТС типа Asterisk. В настоящее время на рынке представлено несколько моделей виртуальных АТС, таких как Digium Asterisk, 3CX, FreeSwitch и openSIPS. Для организации учебного процесса необходимо, чтобы программная АТС была одновременно универсальной, полнофункциональной и простой в освоении и на эту роль идеально подошла АТС Asterisk [2].

Кроме этого в схему лабораторного комплекса по изучению мультисервисной сети входят: Учрежденческая производственная автоматическая телефонная станция (УПАТС) типа Коралл, которая находится в лаборатории Л-409, оборудование SI3000 компании Iskratel в Л-411, Виртуальный комплекс Сотсби а Л-407, а также УПАТС типа Definity, которая расположена в лаборатории Л-404.

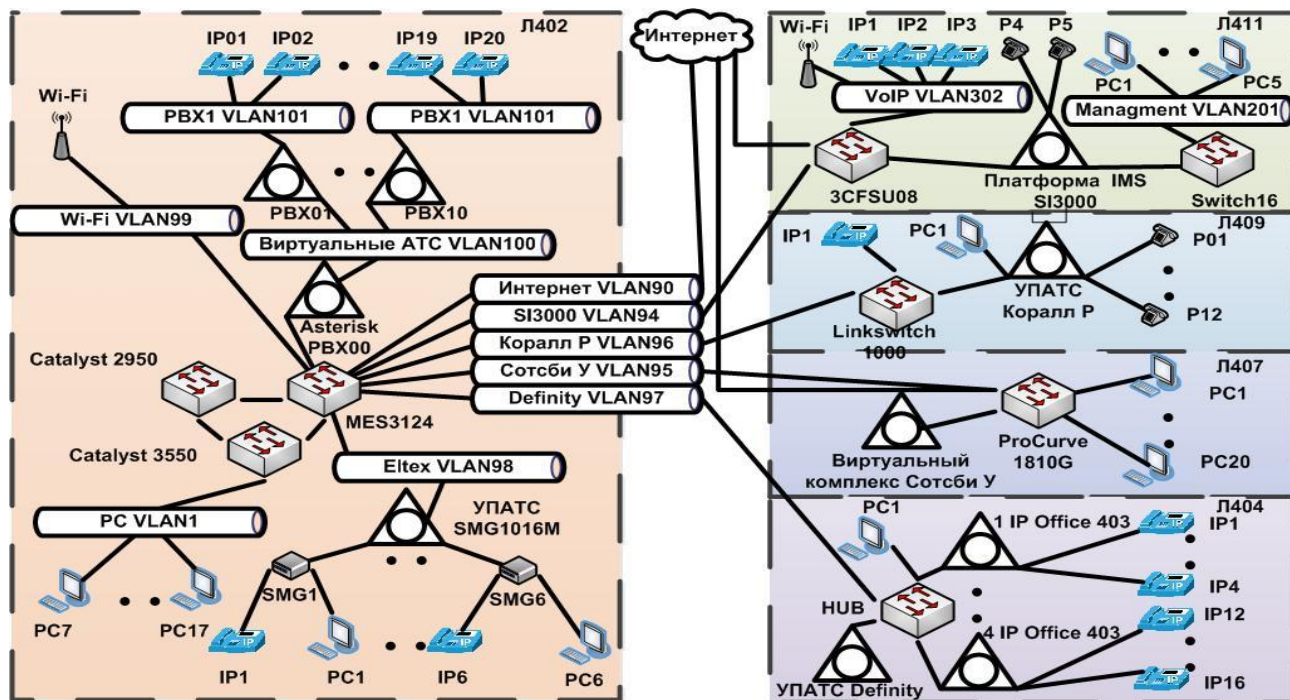


Рис. 1. Схема лабораторного комплекса по изучению мультисервисных сетей связи

Основная работа проводится в лаборатории Л-402, где находится учебный лабораторный комплекс по изучению виртуальной телефонной станции IP-АТС типа Asterisk, схема которого представлена на рис. 2.

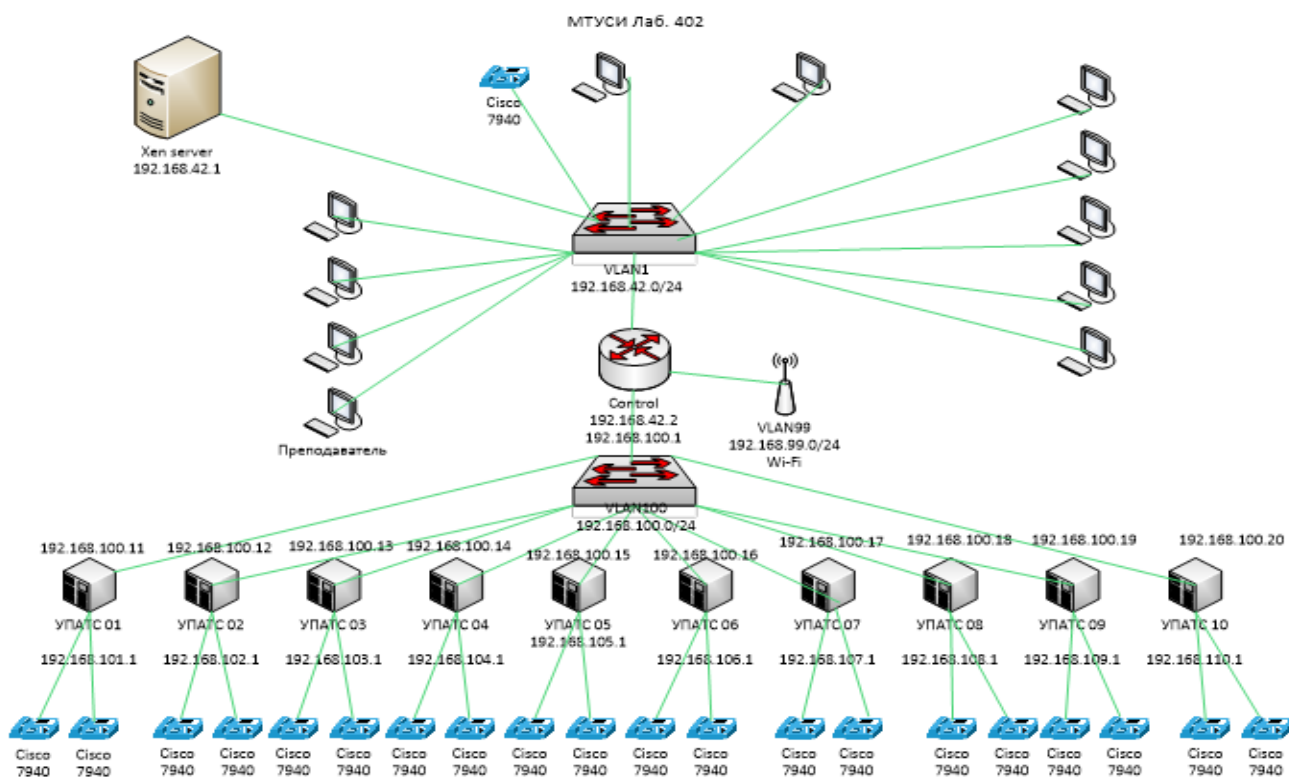


Рис. 2. Схема лабораторного комплекса по изучению виртуальной телефонной станции IP-АТС Asterisk

Здесь имеется 10 рабочих мест, т.е. одновременно 10 бригад студентов могут выполнять лабораторные работы. Каждое рабочее место состоит из рабочей станции, виртуальной машины с

установленной АТС, коммутаторов и двух телефонов. Студентам предоставлены права администратора для доступа к чтению и записи данных в корневых папках на виртуальной машине с АТС. Это необходимо для выполнения привилегированных операций, таких как снятие трафика с сетевого интерфейса или перезапуск сервиса Asterisk. Каждая АТС оснащена двумя сетевыми интерфейсами: Ethernet 0 и Ethernet 1. Интерфейс Ethernet 0 имеет IP-адрес 192.168.100.(10+N). Ethernet 1 имеет IP-адрес 192.168.(100+N).1, где N – номер рабочего места бригады. Данные IP-адреса необходимо вводить каждой бригаде при регистрации абонентов на АТС. Рабочие места полностью независимы и действия одной бригады студентов не влияют на работу остальных бригаад.

В настоящее время на базе лаборатории по изучению Мультисервисных сетей создано уже пять лабораторных работ. Это следующие работы:

- 1) Регистрация абонентов VoIP на АТС типа Asterisk и изучение параметров качества обслуживания;
- 2) Организация связи между двумя АТС типа Asterisk;
- 3) Создание интерактивного голосового меню и настройка голосовой почты на АТС типа Asterisk;
- 4) Предоставление дополнительных видов обслуживания на виртуальной АТС типа Asterisk;
- 5) Организация связи между виртуальной АТС типа Asterisk и оборудованием SI3000 Iskratel.

Первые две работы были уже успешно протестированы студентами, обучающимися на нашей кафедре и описаны в [3]. Поэтому более подробно остановимся на следующих трех работах.

Описание лабораторных работ

Лабораторная работа №3 посвящена изучению интерактивного голосового меню и настройке голосовой почты. При выполнении данной работы студенты используют основные команды Asterisk, которые необходимы для реализации голосового функционала на станции. Так например, при создании интерактивного голосового меню - IVR (Interactive Voice Response), выполняются функции маршрутизации звонков внутри центра обслуживания вызовов (Call-центра). При этом используется информация, которая вводится студентами на клавиатуре телефонного аппарата с помощью тонального набора.

Также в данной работе производится настройка голосовой почты. Для этого студенты производят запись разговора в звуковой файл с заданным именем. При этом можно выбрать звуковой формат, в котором необходимо сохранить запись. Есть возможность записать сообщение сразу нескольким абонентам. Запись может быть произведена на различных языках. Кроме этого, для проверки качества разговорного соединения используется приложение, с помощью которого можно запустить эхо-тест.

На рисунке 3 представлена блок-схема, по которой студенты выполняют задание, по созданию интерактивного голосового меню IVR и настройки голосовой почты.

Лабораторная работа №4 знакомит студентов с другими дополнительными видами обслуживания, такими как: конференцсвязь и перевод вызова на другой номер. Также на данном лабораторном комплексе через точку доступа Wi-Fi студенты могут позвонить со своих смартфонов на телефонные аппараты лабораторного комплекса. Для этого каждой бригаде необходимо создать и настроить абонентов АТС Asterisk и создать учетные записи для подключения мобильных телефонов, установив программу Zoiper или любой другой аналогичный sip-phone.

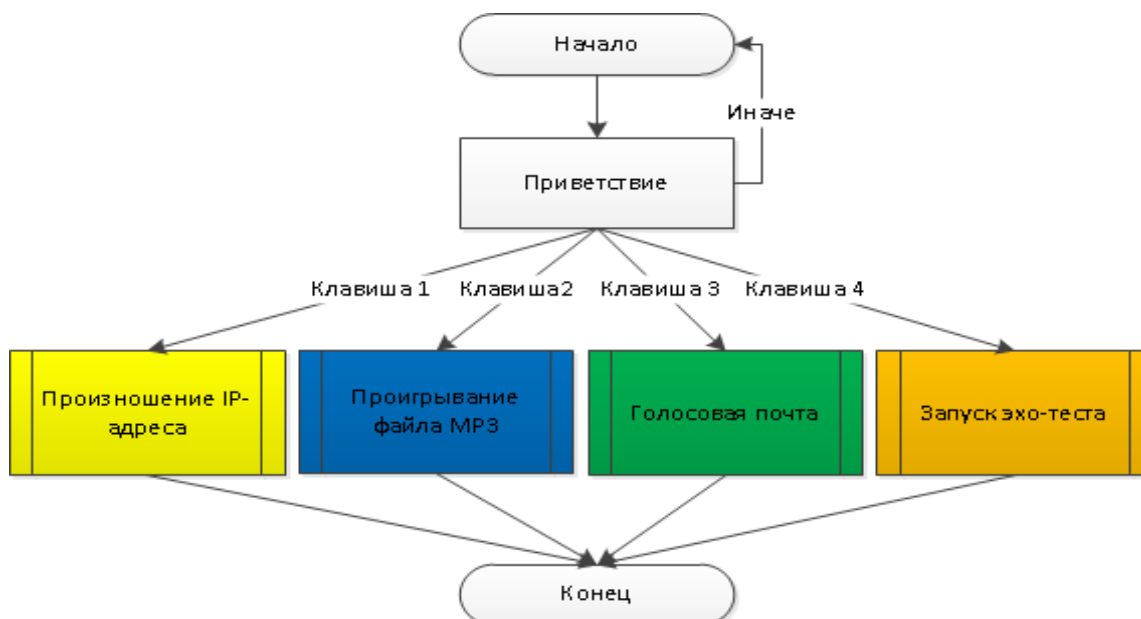


Рис. 3. Блок-схема выполнения лабораторной работы №3

Лабораторная работа №5 посвящена изучению маршрутизации на мультисервисной сети. Для этого было организовано соединение между лабораторным макетом SI3000 компании Iskratel и лабораторным макетом Asterisk. Для выполнения данной работы был разработан единый план нумерации для абонентов всей мультисервисной сети. Все абоненты имеют десятизначную нумерацию, используется префикс 495. На рисунке 4 показан план нумерации для всей мультисервисной сети лаборатории.

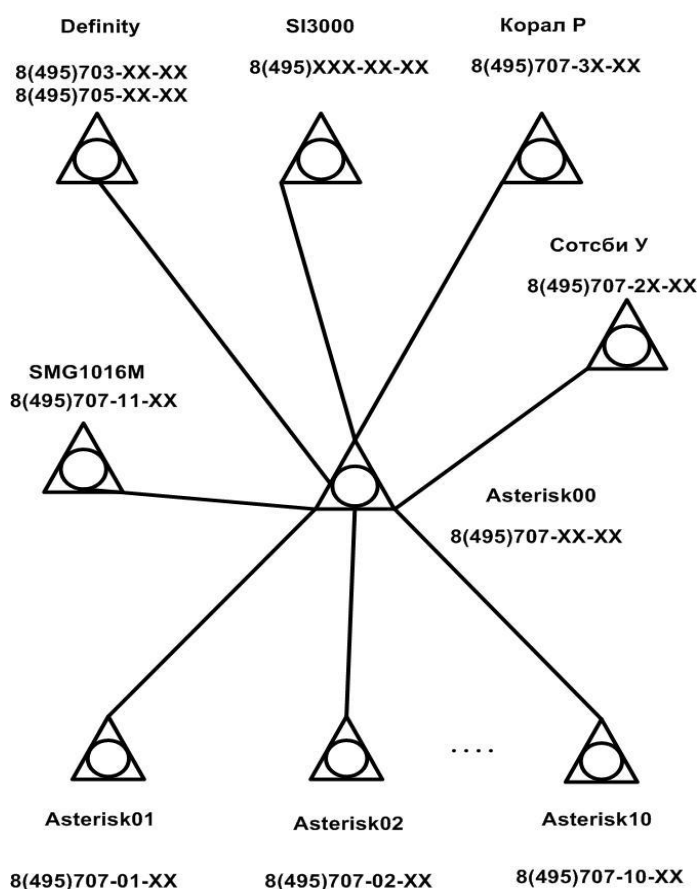


Рис. 4. План нумерации для мультисервисной сети лаборатории

На рисунке 5 показана схема соединения между абонентами станций Астериск и Iskratel. Данная лабораторная работа выполняется одновременно на двух лабораторных стендах. В этой работе студентам предлагается установить связь между двумя телефонными аппаратами станций SI3000 и Asterisk и осуществить звонки с телефонных аппаратов этих станций.

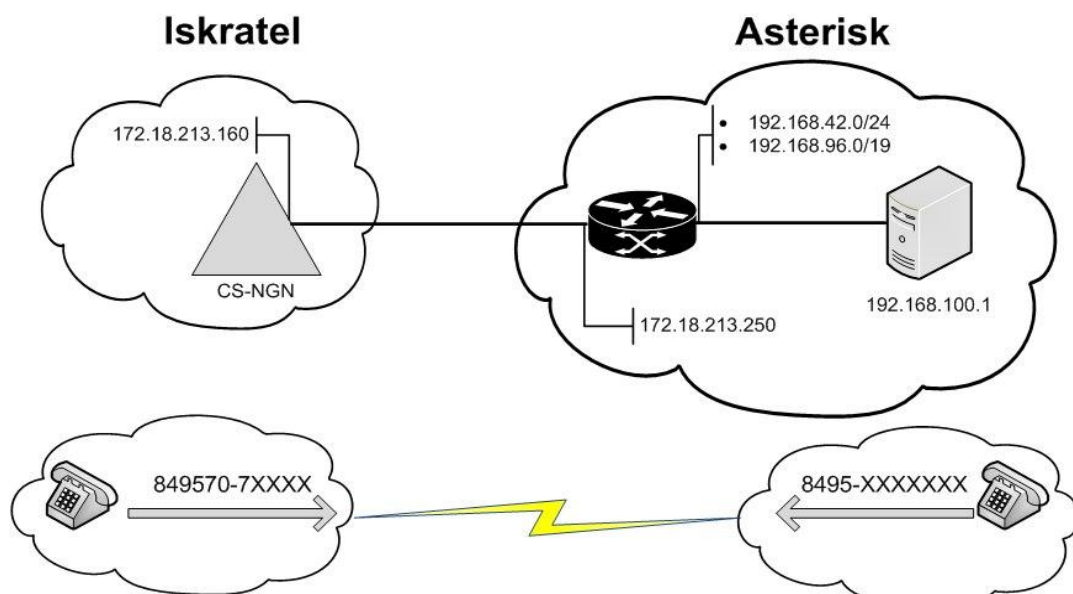


Рис. 5. Установление соединения между абонентами стенда SI3000 Iskratel и абонентами IP-УПАТС Asterisk

На всех компьютерах лабораторного комплекса установлена программа Wireshark, с помощью которой студенты изучают работу различных протоколов IP-телефонии. Эта программа захватывает речевые пакеты (RTP-пакеты) и выводит на экран данные RTP-сессии. Во время разговора значительную его часть составляют паузы. В программе Wireshark имеется возможность представить аудио поток визуально в виде графика, на котором показаны моменты передачи речевых пакетов и пауз между ними. Так на рисунке 6 показана визуализация потока RTP-пакетов, которые передаются между абонентами стенда SI3000 Iskratel и абонентами виртуальной АТС типа Asterisk. На нижней части рисунка приведены IP-адреса источника и получателя голосовых пакетов, номера портов, тип кодека и время отправления. При этом в качестве сигнального протокола для установления соединения используется протокол SIP.



Рис. 6. Визуализация аудиопотока между абонентами стенда SI3000 Iskratel и абонентами АТС типа Asterisk

Дальнейшие перспективы развития лабораторного комплекса

В настоящее время на лабораторном комплексе проводится разработка новых лабораторных работ. Это работы по изучению принципов маршрутизации между различными станциями мультисервисной сети. Так в существующую мультисервисную сеть уже включены виртуальная IP-УПАТС Asterisk, SI3000 Iskratel, IP-УПАТС Coral и оборудование Элтекс. Новые работы помогут студентам освоить принципы установления соединения между различными АТС. Также для приобретения практических навыков работы с оборудованием различных производителей осуществляется постановка новых лабораторных работ, направленных на освоение студентами операционной системы Linux, которая используется в большей части современного инфокоммуникационного оборудования, в частности, на оборудовании SI3000 Iskratel и IP-УПАТС Asterisk.

Выводы

1. Реализованная на данном лабораторном комплексе технология виртуализации является одной из важнейших технологий концепции Будущих сетей.
2. Лабораторные работы знакомят студентов с технологиями пакетной коммутации и принципами IP-телефонии.
3. На лабораторном стенде можно приобрести навыки работы с современным инфокоммуникационным оборудованием, а также изучать принципы маршрутизации в современных сетях.
4. В настоящее время производится постановка новых лабораторных работ, где студенты более подробно будут изучать операционную систему Linux, которая используется на современном инфокоммуникационном оборудовании.

Литература

1. *Росляков А.В., Ваняшин С.В.* Будущие сети (Future Networks). Самара: ПГУТИ, 2015. 274 с.
2. <http://asterisk.ru> (Дата доступа к источнику 04.09.2019).
3. *Пелевин И.И., Маликова Е.Е.* Разработка лабораторного практикума по изучению виртуальной телефонной станции IP-АТС Asterisk // Телекоммуникационные и информационные технологии, №2, 2018. С. 38-42.
4. *Маликова Е.Е.* Задачи группового поллинга для сетей мониторинга и телеметрии большой емкости // Т-Comm: Телекоммуникации и транспорт. 2011. Т. 5. № 7. С. 106-109.
5. *Антонова В.М., Маликова Е.Е.* Исследование взаимного влияния полезного и служебного трафика в сетях LTE // Т-Comm: Телекоммуникации и транспорт. 2014. Т. 8. № 9. С. 17-21.
6. *Антонова В.М., Маликова Е.Е.* Исследование эффективности совместной передачи разнородного трафика в сети LTE // Т-Comm: Телекоммуникации и транспорт. 2017. Т. 11. № 9. С. 22-25.
7. *Степанов С.Н., Степанов М.С., Маликова Е.Е., Цогбадрах А., Ндайикунда Ж.* Построение и анализ обобщенной модели разделения ресурса для LTE технологий с функциональностью NB-IoT // Т-Comm: Телекоммуникации и транспорт. 2018. Т. 12. № 12. С. 71-77.
8. *Маликова Е.Е.* Особенности преподавания дисциплины "Системы коммутации" по направлению инфокоммуникационные технологии и системы связи // Методические вопросы преподавания инфокоммуникаций в высшей школе. 2015. Т. 4. № 2. С. 245-249.
9. *Маликова Е.Е., Пшеничников А.П.* Особенности формирования компетенций по направлению подготовки "Инфокоммуникационные технологии и системы связи" // Методические вопросы преподавания инфокоммуникаций в высшей школе. 2017. Т. 6. № 2. С. 23-25.
10. *Вилков А.Р., Касапов К.В., Маликова Е.Е.* Постановка лабораторных работ на базе оборудования SI3000 компании Iskratel // Телекоммуникации и информационные технологии. 2016. Т. 3. № 1. С. 85-88
11. *Касапов К.В., Оханцев С.С., Маликова Е.Е.* Постановка лабораторных работ по предоставлению дополнительных услуг на базе оборудования SI3000 компании Iskratel // Телекоммуникации и информационные технологии. 2017. Т. 4. № 1. С. 34-39
12. *Антонова В.М., Богомолова Н.Е., Маликова Е.Е.* О новом лабораторном практикуме по изучению виртуальной телефонной станции IP-АТС Asterisk // Методические вопросы преподавания инфокоммуникаций в высшей школе. 2017. Т. 6. № 3. С. 20-22

DEVELOPMENT OF LABORATORY WORKS FOR THE STUDY AND RESEARCH OF MULTISERVICE COMMUNICATION NETWORKS

Margarita G. Kanishcheva

student of the group M61902 MTUCI

margo.kan@list.ru

Denis B. Zhelezov

Engineer, ISKRATEL

denis.zhelezka@mail.ru

Elena E. Malikova

MTUCI, Ph.D., associate professor of of SSaSK department

emalikova@gmail.com

Keywords: *multiservice communication networks, IP-telephony, laboratory complex, IP-PBX, additional types of service, Call center.*

The laboratory complex is located at the Department of Communication Networks and Switching Systems (SSaSK) MTUCI. Five laboratory works were created on the basis of this complex, which cover theoretical issues and practical problems that are solved by communication engineers in the process of operating telephone networks. These works are carried out by students in the direction of training 11.04.02 Infocommunication technologies and communication systems. This article is devoted to the description of laboratory work in which additional types of customer service are studied.

АНАЛИЗ СПЕЦИФИКАЦИИ OPEN FLOW КОНЦЕПЦИИ SDN ДЛЯ СЕТЕЙ БУДУЩЕГО ПОКОЛЕНИЯ

*Кузнецов Петр Геннадьевич,
студент группы 1М61801, МТУСИ
peter.kuznetsov@gmail.com*

*Данилов Алексей Николаевич
МТУСИ, к.т.н., доцент кафедры ССuСК
alexnicdanilov@yandex.ru*

Ключевые слова: программно-конфигурируемая сеть, спецификация Open Flow, маршрутизация, контроллер, приложение, коммутатор

Рассмотрены предпосылки появления концепции программно-конфигурируемых сетей SDN. Рассматривается архитектура сетей SDN. Дано определение спецификации Open Flow. Производится анализ спецификации Open Flow концепции SDN для сетей будущего поколения с точки зрения функционирования сети. Показано, что спецификация Open Flow обеспечивает взаимодействие контроллера со всей сетью SDN посредством организации безопасной сессии и обмена сообщениями.

В современном мире бизнес в сфере информационных технологий предъявляет все большие требования к гибкости и масштабируемости компьютерных сетей.

При всем этом, можно наблюдать, что основными трендами развития корпоративных сетей и сетей центров обработки данных являются:

- стремительный рост объемов трафика и изменение его структуры в сторону передачи видео и унифицированных коммуникаций (UC-C);
- необходимость поддержки мобильных пользователей (BYOD) и социальных сетей;
- высокопроизводительные кластеры для обработки больших данных (BIG DATA);

При этом сеть в классическом ее виде (управление через командную строку и конфигурационные файлы) становится ограничивающим фактором развития вычислительной инфраструктуры. Классические подходы к решению проблем, к примеру, на основе виртуализации сетей (VLAN, VRF), не соответствует уровню развития виртуализации серверов и систем хранения данных. Традиционные сети прежде всего статичны и не соответствуют быстрой динамике развития современного IT бизнеса. Возможности масштабирования традиционных сетей не соответствуют требованиям крупного бизнеса и сервис провайдеров (Deutsche Telekom, Facebook, Google, Microsoft, Verizon и Yahoo), а распределенное управление устройствами традиционных сетей слишком сложное и неэффективное. Привязка же к выбранному сетевому производителю не гарантирует поддержку будущих приложений и сервисов. Как результат наблюдается картина, что традиционные архитектуры/дизайны сетей становятся неэффективны в динамических средах.

Необходима новая технология или подход к построению информационных сетей позволяющая решить перечисленные выше проблемы. Такая технология есть и носит название - Software Defined Networking или сокращенно SDN (программно-конфигурируемая сеть) - сеть передачи данных, в которой уровень управления сетью отделён от устройств передачи данных и реализуется программно, одна из форм виртуализации вычислительных ресурсов.

Предпосылки появления SDN

До Open Flow, и, разумеется, до появления термина Software Network, существовали различные идеи и технологии (табл. 1), создание и развитие которых привело к тому, что теперь называется SDN [1, 6-8]. Некоторые из самых ранних работ в программируемых сетях начались не с Интернет-маршрутизаторов и коммутаторов, а с АТМ-коммутаторов. Двумя примечательными примерами были Devolved Control of ATM Networks (DCAN) и открытая сигнализация. Как видно из названия, DCAN предписывает разделение управления коммутаторами АТМ от самих коммута-

торов. Этот контроль будет приниматься внешним устройством, которое аналогично роли контроллера в сетях SDN.

Открытая сигнализация предложила набор открытых программируемых интерфейсов для аппаратных средств коммутации ATM. Основная концепция заключалась в том, чтобы отделить программное обеспечение управления от коммутационного оборудования. Эта работа при поддержке IETF, привела к созданию общего протокола управления коммутаторами (GSMP).

В GSMP централизованный контроллер может устанавливать и освобождать соединения на коммутаторе ATM, а также множество других функций, которые в противном случае могут быть достигнуты посредством распределенных протоколов на традиционном маршрутизаторе.

Multiprotocol Label Switching (MPLS) и связанные с ними технологии являются отклонением от автономных распределенных решений о пересылке, характерных для традиционного Интернет-маршрутизатора, и в этом смысле они были небольшим шагом на пути к более подобной SDN парадигме.

Активные сети также включали очень новое предложение для небольших загружаемых программ, называемых капсулами, которые будут перемещаться с пакетами на маршрутизаторы и могут перепрограммировать поведение маршрутизатора на лету.

Таблица 1

Идеи и технологии, повлиявшие на создание SDN

| Название | Описание |
|-------------------------|--|
| Открытая сигнализация | Разделение плоскостей пересылки и управления в ATM-коммутации (1999 г.) |
| Активная сеть | Разделение элементов управления и программируемых переключателей (конец 1990-х годов) |
| DCAN | Разделение плоскостей пересылки и управления в ATM коммутации (1997 г.) |
| IP-коммутация | Управление переключателями уровня два в качестве три слоя маршрутизации (конец 1990-х годов) |
| MPLS | Разделительное программное обеспечение для управления (установления полустатических путей пересылки потоков) традиционных маршрутизаторов (конец 1990-х годов) |
| RADIUS, COPS | Контроль доступа к сети на основе политик, установленных сетевым администрированием. Определение, следует ли вводить пользователя в сеть (2010 г.) (Remote Authentication Dial-In User Service (RADIUS), Common Open Policy Service) |
| Оркестрация | Использование SNMP и CLI для автоматизации настройки сетевого оборудования (2008 г.) |
| Диспетчер виртуализации | Использование плагинов для выполнения перенастройки сети для поддержки виртуализации серверов (2011 г.) |
| For CES | Разделение плоскостей пересылки и управления (2003 г.) |
| 4D | Распознавание интеллекта контрольной плоскости в централизованной системе (2005 г.) |
| Ethane | Достижение полного доступа к предприятиям и сетям и управление ими с использованием отдельных плоскостей переадресации и управления и использования централизованного контроллера (2007 г.) |

Эти программы могут быть настолько мелкомасштабными, чтобы предписывать правила пересылки для одного пакета, даже, возможно, полезную нагрузку пакета, содержащего программу.

RADIUS был использован для обеспечения автоматической реконфигурации сети. Это решение было несколько перспективным, поскольку RADIUS можно рассматривать как предше-

ственник SDN. Идея заключается в том, что через RADIUS сетевые атрибуты будут меняться на основе идентичности только что появившегося вычислительного ресурса и требуемого подключения к сети.

Ранние попытки автоматизации включали приложения, которые обычно назывались оркестрами. Подобно тому, как дирижер может сделать гармоничное целое из разных инструментов оркестра, такие приложения могут принимать обобщенную команду или цель и применять ее в широком диапазоне. Эти приложения для оркестровки обычно используют общие интерфейсы приложений (API), такие как интерфейс командной строки (CLI) или простой протокол сетевого управления (SNMP).

Работа по пересылке и контролю элементов (Forwarding and Control Element Separation – For CES), созданная в IETF, началась примерно в 2003 году. For CES было одним из оригинальных предложений, рекомендующих развязку плоскостей пересылки и управления. Общая идея For CES заключалась в том, чтобы предоставить простые аппаратные средства пересылки в основу сетевого устройства и элементов управления на основе программного обеспечения выше.

Эти простые аппаратные форвардеры были построены с использованием технологии коммутации ячеек или переключения меток. Контроллер с программным обеспечением отвечал за более широкие задачи, часто включающие координацию между несколькими сетевыми устройствами (например, обновления маршрутизации BGP). For CES предлагает разделение плоскости пересылки от плоскости управления и предлагает два разных варианта осуществления этой архитектуры.

Концепция 4D, названная в честь четырех плоскостей – определения, распространения, обнаружения и данных - предлагает идею сосредоточения работы плоскости управления в отдельной и независимой системе, предназначенной для этой цели.

Концепция 4D утверждает, что состояние сетей сегодня является хрупким и, следовательно, часто течет на краю отказа из-за его текущей конструкции, основанной на распределенных автономных системах. Небольшое локальное событие, такое как некорректная конфигурация протокола маршрутизации, может иметь серьезное глобальное воздействие.

For CES и 4D внесли большой вклад в эволюцию концепций, лежащих в основе SDN: разделение плоскостей переадресации и управления (For CES) и централизованный контроллер, отвечающий за общие решения о маршрутизации и пересылке (4D). Однако оба этих предложения страдают от отсутствия фактических реализаций.

Концепция Ethane была представлена в документе под названием «Переосмысление корпоративного сетевого контроля» в 2007 году. Ethane - это решение на основе политик, которое позволяет сетевым администраторам определять политики, касающиеся доступа к сетевым уровням для пользователей.

Концепция Ethane построена на трех основных принципах:

- Сеть должна регулироваться политиками высокого уровня.
- Маршрутизация для сети должна быть осведомлена об этих политиках.
- Сеть должна обеспечивать привязку между пакетами и их происхождением.

Поведение переключателей Ethane, как правило, совпадает с сегодняшними коммутаторами Open Flow, которые передают и фильтруют пакеты на основе таблиц потоков, которые были настроены на устройстве. Если коммутатор не знает, что делать с пакетом, он перенаправляет его на контроллер и ожидает дальнейших инструкций. Короче говоря, Ethane - это, в основном, технология Software Defined Networking, а ее компоненты - antecedents Open Flow.

Архитектура SDN

Архитектура программно-конфигурируемых сетей SDN (см. рис. 1) подразделяется на три уровня:

- Уровень данных.
- Уровень управления.
- Уровень приложений.

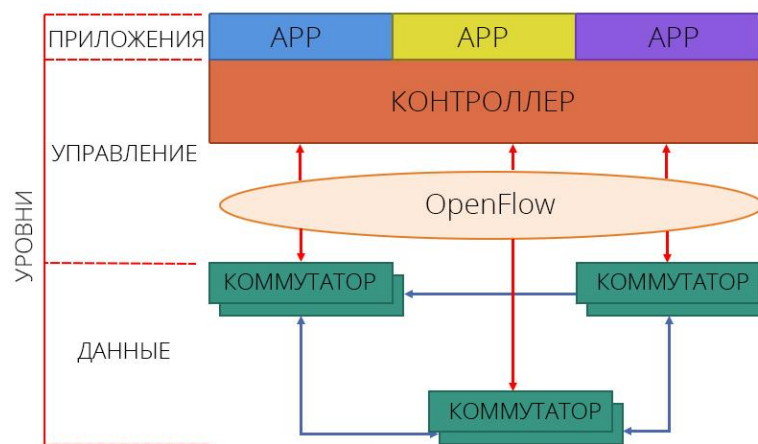


Рис. 1. Архитектура программно-конфигурируемых сетей

Опишем каждый из уровней подробнее.

Инфраструктурный уровень (физический, уровень данных) предоставляет набор сетевых устройств, коммутаторов и каналов передачи. На этом уровне в концепции SDN находятся простые устройства, обеспечивающие только пересылку пакетов, за счет таблиц потоков. Вынесение уровня управления из этих устройств значительно уменьшает их стоимость и скорость обработки входящего трафика, а также упрощает настройку.

Уровень управления, включающий в себя сетевую операционную систему, которая обеспечивает приложениям сетевые сервисы и программный интерфейс для управления сетевыми устройствами и сетью. Данный уровень выведен в контроллер – устройство с большими вычислительными мощностями и интерфейсами для связи с коммутаторами и приложениями. Контроллер – центр SDN, он видит всю топологию сети и управляет всеми изменениями в сетевых устройствах через протокол Open Flow автоматически или по средствам команд от администратора.

Уровень сетевых приложений необходим для гибкого и эффективного управления сетью. Он обеспечивает программный интерфейс для приложений управления сетью и реализует механизмы управления таблицами коммутаторов: добавление, удаление, модификацию правил и сбор разнообразной статистики. Таким образом, фактически решение задач управления сетью выполняется с помощью приложений, реализованных на основе API сетевой операционной системы, позволяющих создавать приложения [2].

Определение спецификации Open Flow

Open Flow – стандартный протокол, является основным элементом концепции SDN который обеспечивает взаимодействие контроллера с сетевыми устройствами [3]. Контроллер используется для управления таблицами потоков коммутаторов, на основании которых принимается решение о передаче принятого пакета на конкретный порт коммутатора. Таким образом, в сети формируются прямые сетевые соединения с минимальными задержками передачи данных и необходимыми параметрами. Соответственно коммутатор Open Flow состоит, как минимум, из двух компонент, т.е. таблицы потоков (flow table) и безопасного канала (secure channel).

| MAC src | MAC dst | IP Src | IP Dst | TCP dport | ... | Action | Count |
|---------|---------|--------|---------|-----------|-----|------------|-------|
| • | 10:20:. | • | • | • | • | port 1 | 250 |
| • | • | • | 5.6.7.8 | • | • | port 2 | 300 |
| • | • | • | • | 25 | • | drop | 892 |
| • | • | • | 192. | • | • | local | 120 |
| • | • | • | • | • | • | controller | 11 |

Рис. 2. Пример таблицы потоков Open Flow

Коммутаторы с поддержкой Open Flow уже доступны на рынке, так в портфолио лидера в разработке концепции SDN – компании Hewlett-Packard, уже более 40 моделей коммутаторов поддерживают Open Flow версии 1.3, соответственно готовы выступать «кирпичиками» построения реальной сети SDN. Кроме коммутаторов Hewlett-Packard предлагает несколько моделей готовых контроллеров SDN и бесплатно предоставляет несколько готовых приложений SDN для конкретных бизнес программ, к примеру, Microsoft Lync. Компания HP также поддерживает активное сообщество разработчиков SDN (sdndevcenter.hp.com), где пользователи могут делиться своими идеями, а также онлайн-магазин приложений SDN App Store, откуда пользователи могут скачивать различные приложения на контроллер HP VAN SDN всего в несколько кликов.

Появление концепции Open Flow

Представленные выше стандарты и предложения, были предшественниками появления новой парадигмы, но точкой, на которой фактически родился SDN является появление Open Flow. Open Flow был разработан, чтобы позволить экспериментировать и внедрять новые протоколы в повседневных сетях. Спецификация Open Flow определяется как протокол, который используется между контроллером и коммутатором [4]. Его простую архитектуру представим на рис. 3.

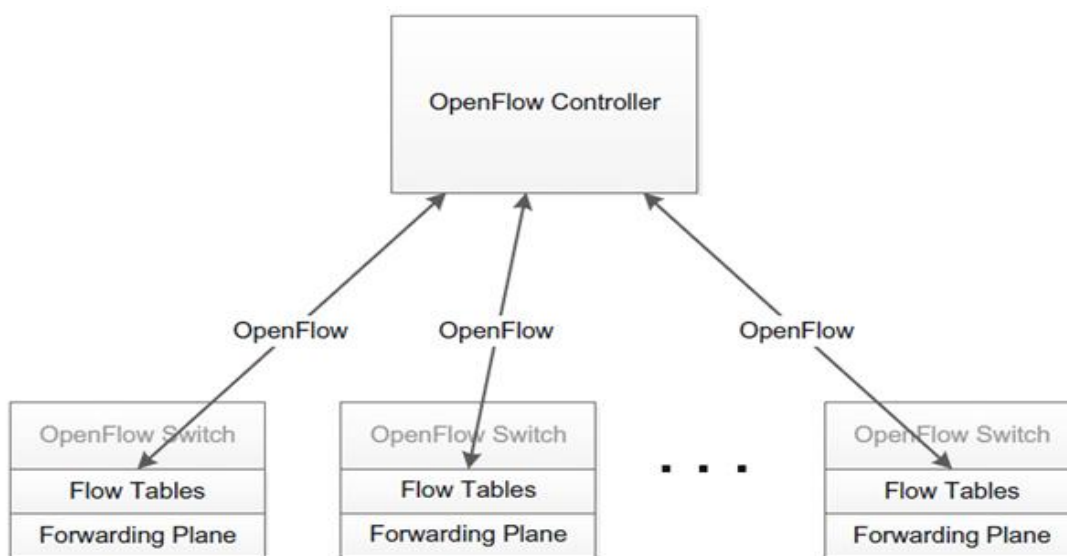


Рис. 3. Архитектура Open Flow

Open Flow собрал достаточный импульс, чтобы ответственность за сам стандарт перешла в Open Networking Foundation (ONF). Одним из аспектов ONF является то, что его совет директоров полностью состоит из крупных сетевых операторов, а не самих сетевых производителей.

Перемещение управления с устройства

Как говорилось ранее, управляющее программное обеспечение можно считать интеллектом сетевого устройства, которое определяет оптимальные пути и реагирует на свои и новые сетевые требования.

В концепции SDN подразумевается перемещение этого интеллекта в некое централизованное устройство – контроллер, который способен видеть всю сеть и принимать оптимальные решения с учетом полного понимания ситуации и топологии сети [5].

В SDN сетевая деятельность организуется следующим образом:

- Функции маршрутизации, реализованные в аппаратных таблицах, остаются на устройстве. Кроме того, такие функции, как фильтрация на основе ACL и приоритезации трафика, также применяются локально на устройстве.
- Сложное программное обеспечение управления удаляется с устройства и помещается в централизованный контроллер, который имеет полное представление о сети и возможность оптимальных решений по переадресации и маршрутизации.

Заключение

Благодаря использованию Open Flow ресурсы коммутатора можно использовать для более быстрой переадресации пакетов и для решения более сложных задач. При помощи контроллеров сетевые администраторы могут более эффективно управлять своими сетями, а над контроллером запускаются сетевые приложения, реализующие функции более высокого уровня и, кроме того, участвующие в принятии решений о том, как лучше всего управлять и контролировать пересылку и распространение пакетов.

Литература

1. Галич С. В., Сердюкова О. Е., Сафонова И. К. Обзор архитектуры SDN-контроллера Open Day light // Проблемы передачи информации в инфокоммуникационных системах: Сб. докладов и тезисов VI Всероссийской научно-практической конференции. Волгоград: Изд-во ВолГУ, 2015. С. 23-28.
2. Данилов А. Н., Максимов С.П., Кочетков Н.А. Перспективы развития программно-конфигурируемых сетей связи// Международный форум информатизации. Труды международной научно-технической конференции "Телекоммуникационные и вычислительные системы -2017". М.: Горячая линия – Телеком, 2017. С. 93-94.
3. Росляков А.В., Ваняшин С.В. Будущие сети (Future Networks). Самара: ПГУТИ, 2015. 274 с.
4. Paul Göransson and Chuck Black. Software defined networks a comprehensive approach. USA. 2014. 315 p.
5. Thomas D. Nadeau and Ken Gray. SDN: Software Defined Networks. 2013. 353 p.
6. Данилов А.Н., Максимов С.П., Крылов Р.Х. Исследование процессов обмена сигнальными сообщениями протокола SIP в сетях NGN // Т-Comm: Телекоммуникации и транспорт. 2014. Т. 8. № 8. С. 32-36
7. Данилов А.Н., Максимов С.П., Куприянов Д.О. Анализ вариантов использования беспроводных технологий при построении мультисервисных офисных сетей связи // Т-Comm: Телекоммуникации и транспорт. 2017. Т. 11. № 11. С. 26-29.
8. Деревягин П.Б., Данилов А.Н. Методы локализации сбоев в сети IP/MPLS // Телекоммуникации и информационные технологии. 2018. Т. 5. № 1. С. 72-75.

OPEN FLOW SPECIFICATION ANALYSIS SDN CONCEPTS FOR FUTURE GENERATION NETWORKS

Peter G. Kuznetsov

student of group 1M61801, MTUCI

peter.kyznetsov@gmail.com

Alexey N. Danilov

MTUCI, PhD., associate professor of CN and SS department

alexnicdanilov@yandex.ru

Keywords: *software-configured network, Open Flow specification, routing, controller, application, switch*

The prerequisites for the emergence of the concept of software-configurable SDN networks are considered. The architecture of SDN networks is considered. The definition of the Open Flow specification is given. An analysis is made of the Open Flow specification of the SDN concept for future generation networks in terms of network operation. It has been shown that the Open Flow specification allows the controller to interact with the entire SDN network by organizing a secure session and messaging.

РАЗРАБОТКА И АНАЛИЗ МАТЕМАТИЧЕСКОЙ МОДЕЛИ ОБСЛУЖИВАНИЯ ВЫЗОВОВ В ПЕРСПЕКТИВНЫХ КОНТАКТ-ЦЕНТРАХ

Шишкин Максим Олегович

студент группы М091801(72), МТУСИ

mackschischkin1@yandex.ru

Степанов Михаил Сергеевич

МТУСИ, к.т.н., доцент кафедры ССuСК

mihstep@yandex.ru

Ключевые слова: контакт-центр, математическая модель, оценка качества работы контакт-центра, расчет количества операторов, расчет доли потерянных заявок.

Поставлены задачи для реализации математической модели контакт-центра. Представлена математическая модель перспективного контакт-центра. Анализ модели и реализация одним из алгоритмических языков программирования. Демонстрация и проверка правильной работы данной модели. Оценка качества работы контакт-центра на численном примере. Расчет среднего количества занятых операторов и доли потерянных заявок в процентном соотношении.

Введение

Контакт-центры широко используются во многих областях, важных для жизни современного человека, а именно в банковских услугах, здравоохранении, ситуационных центрах, ритейл. 60% компаний-лидеров считают, что у них правильное соотношение «живых» и автоматизированных каналов коммуникации с клиентами.

Согласно статистике, от 60% до 70% процентов операционных затрат (ОРЕХ) на содержание контакт-центра составляет заработная плата операторов. По этой причине оптимизация численности персонала является одной из основных задач, стоящих сегодня перед менеджментом компаний.

Формулировка цели и задач

Целью работы является создание программного продукта для оценки числа операторов, необходимого для обслуживания потока вызовов с заданным качеством. Калькулятор оценки качества обслуживания должен учитывать основные тенденции развития современных справочно-информационных служб.

Необходимо провести исследование предметной области, а также построение функциональных и математических моделей, разработку на их основе алгоритмов оценки показателей качества обслуживания клиентов.

Для создания калькулятора оценки численности персонала планируется использовать методы теории телетрафика, теории вероятностей и вычислительной математики. Таким образом, техническое решение задачи разбивается на два этапа: построение математической модели контакт-центра, вывод выражений для показателей качества обслуживания (доля потерянных вызовов, доля обслуженных вызовов и т.д.) и разработка калькулятора оценки численности персонала с применением одного из алгоритмических языков программирования.

Построение модели и тестирование

Рассмотрим модель поступления запросов на информационное обслуживание в контакт-центре. Имеется v операторов, которые обслуживают поступающий поток заявок λ , поток заявок μ , которых обслужил контакт-центр и σ – поток заявок не получивших обслуживание, а так же ω – количество мест ожидания в очереди на обслуживание, в данной рассматриваемой модели это

значение равняется числу операторов контакт-центра, но его можно лимитировать и любыми другими значениями (рис. 1) [1, 5-15].

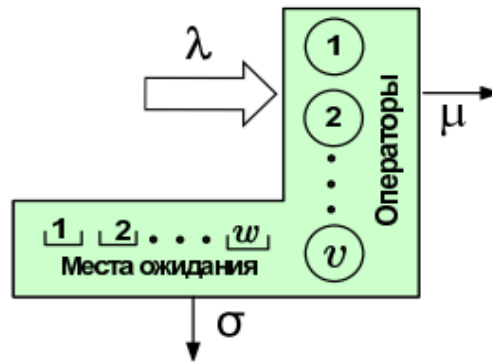


Рис. 1. Математическая модель контакт-центра

Появление заявок задаётся либо моментами времени t_k , $k = 1, 2, \dots$, либо интервалами времени между их поступлениями $x_k = t_k - t_{k-1}$, $k = 1, 2, \dots$ (Рис.2.).

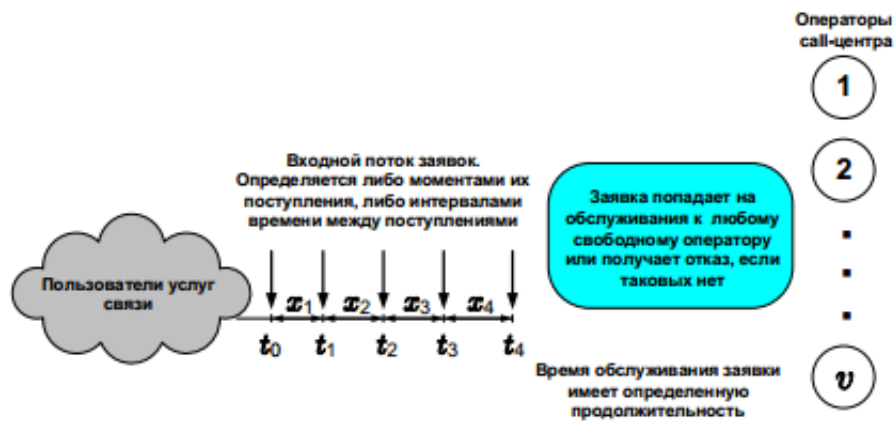


Рис. 2. Модель поступления запросов в контакт-центре

Заявка попадает к любому свободному оператору. Если таковых нет, то она получает отказ и теряется.

Обозначим через $i(t)$ число операторов занятых в момент времени t . Функция $r(t) = i(t)$ – случайный процесс с непрерывным временем и конечным пространством состояний $S = \{0, 1, \dots, v\}$ (рис. 2).

Наличие траектории, отражающей динамику измерений контакт-центра, даёт возможность оценить долю потерянных заявок π_c и среднее число занятых операторов m .

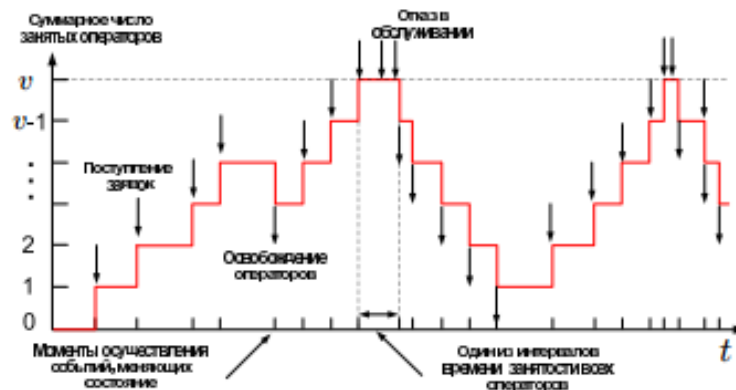


Рис. 3. Функция изменений состояний модели во времени, одна из возможных траекторий процесса

Обозначим через $a(T)$ число заявок, поступивших в интервале $(0, T]$, а через $b(T)$ обозначим число заявок, поступивших в интервале $(0, T]$ и получивших отказ из-за нехватки операторов или отсутствия мест в очереди [2]. То π_c и m рассчитываются из выражений:

$$\pi_c = \lim_{T \rightarrow \infty} \frac{b(T)}{a(T)}; \quad m = \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T i(t) dt. \quad (1)$$

Так как функция изменений состояний имеет ступенчатый характер, то формулу для оценки среднего числа занятых операторов (1) будет выглядеть так:

$$m = \lim_{T \rightarrow \infty} \sum_{i=0}^v \frac{r_i(T)}{T} i = \sum_{i=0}^v p(i) i, \quad (2)$$

Так как случайный процесс нашей математической модели описывает изменение системы в зависимости от параметра t и при этом вероятность любого будущего состояния системы зависит только от её состояний в настоящий момент и не зависит от того, каким оно было в прошлом, то наш процесс называется марковским. Построение марковских процессов основано на применении случайных величин имеющих экспоненциальное распределение. То есть случайная величина ξ имеет экспоненциальное распределение с параметром λ и функцией распределения:

$$F(x) = p\{\xi \leq x\} = \begin{cases} 1 - e^{-\lambda x}, & x \geq 0; \\ 0, & x < 0. \end{cases} \quad (3)$$

В соответствии с этим, вероятности переходов функции состояний из состояния i , будет определяться так:

$$p_{i,i+1} = \frac{\lambda}{\lambda + i\mu}; \quad p_{i,i-1} = \frac{i\mu}{\lambda + i\mu}; \quad (4)$$

Неприводимый марковский процесс тогда и только тогда имеет предельные вероятности, когда система уравнений равновесия (СУР) имеет единственное решение:

$$\alpha_x P(x) = \sum_{(y) \in S \setminus (x)} P(y) \alpha_y p_{y,x} = \sum_{(y) \in S \setminus (x)} P(y) \alpha_{y,x}, \quad (x) \in S \quad (5)$$

Значения стационарных вероятностей находятся из условия нормировки:

$$p(x) = \frac{P(x)}{\sum_{(y) \in S} P(y)}. \quad (6)$$

С помощью индикаторной функции СУР (6) можно записать в виде одного соотношения:

$$\begin{aligned} P(i) \{ \lambda I(i < v) + i \mu I(i > 0) \} = \\ = P(i-1) \lambda I(i > 0) + P(i+1) (i+1) \mu I(i < v), \quad i = 0, 1, \dots, v. \end{aligned} \quad (7)$$

Для каждого конкретного состояния (7) получаем соответствующее выражение СУР. Их полный перечень выглядит так:

$$\begin{aligned} P(0) \lambda &= P(1) \mu, \quad i = 0; \\ P(1) (\lambda + \mu) &= P(0) \lambda + P(2) 2 \mu, \quad i = 1; \\ P(i) (\lambda + i \mu) &= P(i-1) \lambda + P(i+1) (i+1) \mu, \quad i = 2, \dots, v-2; \\ P(v-1) (\lambda + (v-1) \mu) &= P(v-2) \lambda + P(v) v \mu, \quad i = v-1; \\ P(v) v \mu &= P(v-1) \lambda, \quad i = v. \end{aligned} \quad (8)$$

Нельзя забыть про условие нормировки:

$$P(0) + P(1) + \dots + P(v) = 1. \quad (9)$$

Теперь значение всех состояний можно выразить через любую из рассматриваемых вероятностей, например через $P(0)$:

$$P(i) = P(0) \left(\frac{\lambda}{\mu}\right)^i \frac{1}{i!}, \quad i = 0, 1, \dots, v. \quad (10)$$

Из условия нормировки (9) находим:

$$p(0) = \frac{1}{1 + \frac{\lambda}{\mu} + \left(\frac{\lambda}{\mu}\right)^2 \frac{1}{2!} + \dots + \left(\frac{\lambda}{\mu}\right)^v \frac{1}{v!}}, \quad p(i) = \frac{\left(\frac{\lambda}{\mu}\right)^i \frac{1}{i!}}{1 + \frac{\lambda}{\mu} + \left(\frac{\lambda}{\mu}\right)^2 \frac{1}{2!} + \dots + \left(\frac{\lambda}{\mu}\right)^v \frac{1}{v!}}. \quad (11)$$

Положим $\lambda/\mu = a$. a – среднее число потенциальных соединений.

Доля потерянных заявок совпадает с долей времени занятости всех операторов. Для модели Эрланга эта характеристика обозначается через $E(v, a)$. Для оценки m используем определение из формулы Литтла [3]:

$$\pi_c = p(v) = E(v, a) = \frac{\frac{a^v}{v!}}{1 + a + \frac{a^2}{2!} + \dots + \frac{a^v}{v!}}, \quad m = \sum_{i=0}^v p(i) i = a(1 - E(v, a)). \quad (12)$$

После всего описанного выше можно перейти к тестированию [4].

Выберем количество операторов равное 5 ($v=5$), количество мест ожидания выберем равно количеству операторов, а значит тоже равно 5 ($w=5$), входной поток заявок равный 5 заявок за единицу времени и выходной поток заявок равный 2 заявкам за единицу времени. С учётом входных параметров и формулы (10) реализуем итерационную схему решения СУР:

$$\begin{aligned} P(0)\lambda &= P(1)\mu; \\ P(1)(\lambda + \mu) &= P(0)\lambda + P(2)2\mu; \\ P(2)(\lambda + 2\mu) &= P(1)\lambda + P(3)3\mu; \\ P(3)(\lambda + 3\mu) &= P(2)\lambda + P(4)4\mu; \\ P(4)(\lambda + 4\mu) &= P(3)\lambda + P(5)5\mu; \\ P(5)5\mu &= P(4)\lambda. \end{aligned} \quad (13)$$

Воспользуемся положением итерационного алгоритма Гаусса-Зейделя и составим систему рекурсивных соотношений для СУР (13):

$$\begin{aligned} P^{(s+1)}(0) &= (P^{(s)}(1)\mu)/\lambda; \\ P^{(s+1)}(1) &= (P^{(s+1)}(0)\lambda + P^{(s)}(2)2\mu)/(\lambda + \mu); \\ P^{(s+1)}(2) &= (P^{(s+1)}(1)\lambda + P^{(s)}(3)3\mu)/(\lambda + 2\mu); \\ P^{(s+1)}(3) &= (P^{(s+1)}(2)\lambda + P^{(s)}(4)4\mu)/(\lambda + 3\mu); \\ P^{(s+1)}(4) &= (P^{(s+1)}(3)\lambda + P^{(s)}(5)5\mu)/(\lambda + 4\mu); \\ P^{(s+1)}(5) &= (P^{(s+1)}(4)\lambda)/(5\mu). \end{aligned} \quad (14)$$

Зададим начальные приближения вероятностей состояний нулевым вектором и, подставив входные параметры в выражения (14), сделаем 20 итераций (Табл.1).

Таблица 1

Таблица с прохождением 20 итераций.

| Состояние | Значения $P^{(s)}(i)$ в зависимости от номера итерации s | | | | | | | | |
|-----------|--|-------|-------|-------|-------|-------|-------|-------|-------|
| | 0 | 1 | 2 | 3 | 4 | 5 | 10 | 15 | 20 |
| (0) | 1 | 0,200 | 0,100 | 0,069 | 0,056 | 0,051 | 0,046 | 0,046 | 0,046 |
| (1) | 1 | 0,500 | 0,345 | 0,281 | 0,253 | 0,240 | 0,230 | 0,229 | 0,229 |
| (2) | 1 | 0,786 | 0,671 | 0,618 | 0,594 | 0,583 | 0,574 | 0,574 | 0,574 |
| (3) | 1 | 0,991 | 0,973 | 0,964 | 0,959 | 0,958 | 0,956 | 0,956 | 0,956 |
| (4) | 1 | 1,106 | 1,155 | 1,177 | 1,187 | 1,191 | 1,195 | 1,195 | 1,195 |
| (5) | 1 | 1,106 | 1,155 | 1,177 | 1,187 | 1,191 | 1,195 | 1,195 | 1,195 |

После проделанных действий нужно провести нормировку, используя формулу (9), полученных вероятностей, в результате чего получим: $p(0)=0,011$; $p(1)=0,055$; $p(2)=0,137$; $p(3)=0,228$; $p(4)=0,285$; $p(5)=0,285$.

Далее находим долю потерянных заявок и среднее число занятых операторов, используя формулу (12):

$$\pi_c = p(v) = 0,285; \quad m = 3,576. \quad (15)$$

Имея результаты полученный вручную проверим на правильность работы полученный калькулятор для оценки качества обслуживания контакт-центра (рис. 4).

Сравнивая результаты работы программы и ручной расчёт можно сделать вывод, что программа работает правильно и её можно использовать по назначению.

Далее, попробуем сравнить результаты при тех же значениях входного и выходного потоков, но при разном количестве операторов (рис.5 и 6).

При сравнении двух результатов видно, что чем больше операторов, тем меньше потеря заявок, если быть точнее. то в модели с 8-ю операторами доля потерянных заявок равняется приблизительно 7%, а с 2-мя практически 68%, но при этом следует заметить, что при таких входных параметрах, а именно 8 операторов, весь персонал не используется полностью.

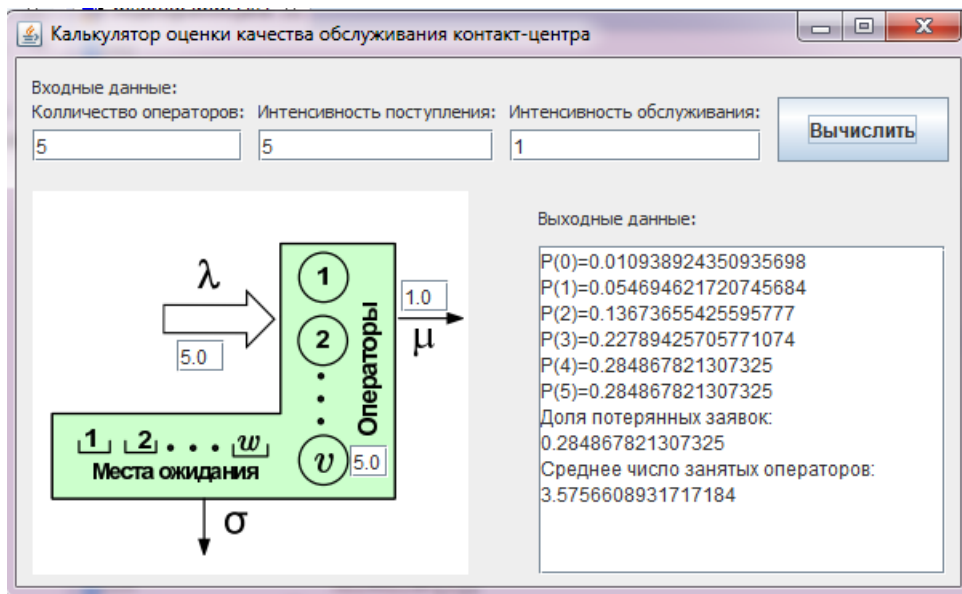


Рис. 4. Результат работы калькулятора

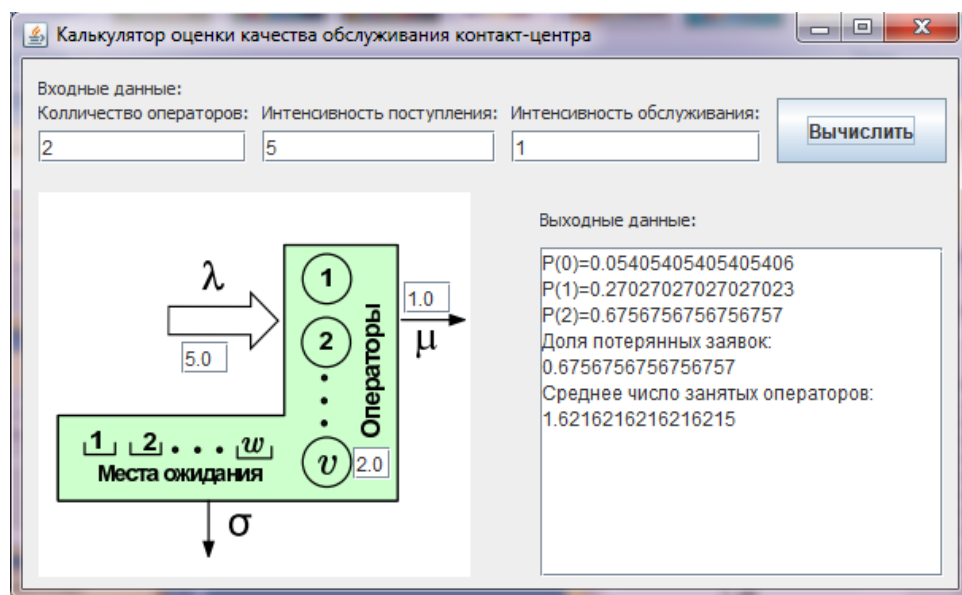


Рис. 5. Результат работы калькулятора с количеством операторов равным 2

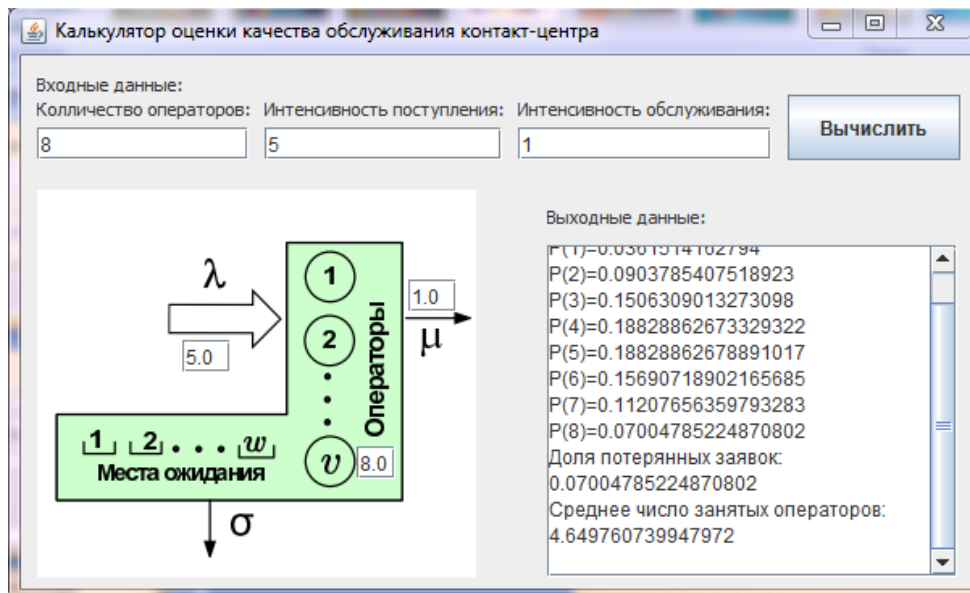


Рис. 6. Результат работы калькулятора с количеством операторов равным 8

Заключение

В данной статье была рассмотрена модель контакт-центра только с группой операторов, в дальнейшем планируется добавить 2 группы консультантов и систему IVR (англ. Interactive Voice Response - система предварительно записанных голосовых сообщений, выполняющая функцию маршрутизации звонков внутри call-центра с использованием информации, вводимой клиентом на клавиатуре телефона с помощью тонального набора) (рис. 7).

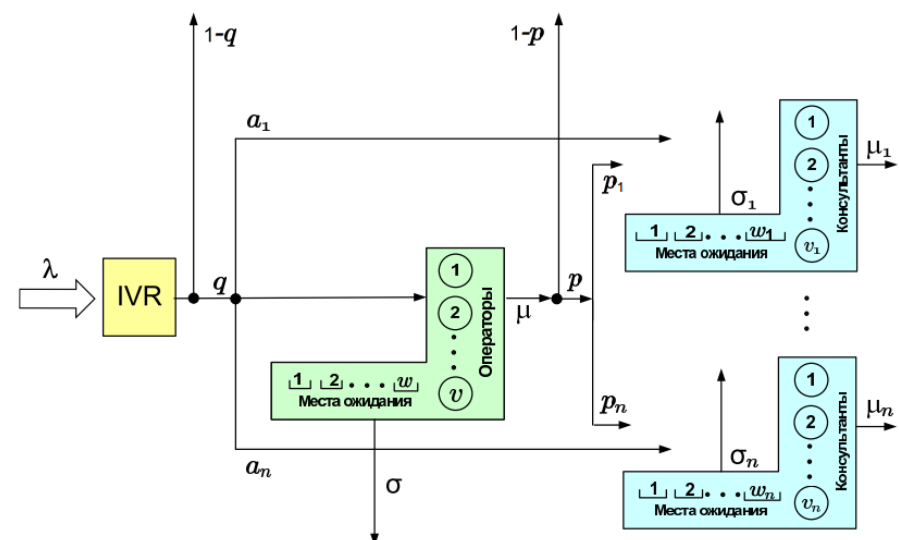


Рис. 7. Математическая модель планируемого контакт-центра

Литература

1. Степанов С.Н. Теория телетрафика: концепции, модели, приложения / Серия "Теория и практика инфокоммуникаций". М.: Горячая линия – Телеком. 2015. 868 с.
2. Степанов М.С. Определение и свойства входных параметров обобщенной модели контакт-центра // Т-Сотт: Телекоммуникации и транспорт. 2015. Том 9. №7. С. 25-30.
3. Степанов М.С. Обобщенная модель контакт-центра и частные случаи ее использования // Т-Сотт. Телекоммуникации и транспорт. 2013. Том 7. №7. С. 126-130.
4. Степанов М.С. Оценка характеристик работы контакт-центра с использованием итерационных методов // Т-Сотт. Телекоммуникации и транспорт. 2012. Том 6. №7. С. 188-192.

5. Степанов М.С. Планирование числа операторов и линий доступа в современных контакт-центрах // Т-Сотт: Телекоммуникации и транспорт. 2014. Т. 8. № 8. С. 89-92.
6. Дудина В.А., Журко А.М., Степанов М.С. Модель контакт-центра с учетом навыков операторов и нетерпеливости абонентов // Т-Сотт: Телекоммуникации и транспорт. 2017. Т. 11. № 12. С. 43-48.
7. Пшеничников А.П., Степанов М.С. Моделирование процесса обслуживания вызовов в современных контакт-центрах // Труды Северо-Кавказского филиала Московского технического университета связи и информатики. 2015. № 1. С. 271-273.
8. Степанов М.С., Степанов С.Н., Журко А.М. Построение математической модели контакт-центра с учетом системы IVR // REDS: Телекоммуникационные устройства и системы. 2017. Т. 7. № 2. С. 253-255.
9. Степанов С.Н., Осия Д.Л. Приближённый метод оценки показателей обслуживания заявок в иерархических сетях доступа с учётом влияния поведения пользователя // Т-Сотт: Телекоммуникации и транспорт. 2014. Т. 8. № 8. С. 93-96.
10. Степанов С.Н., Осия Д.Л. Асимптотические методы оценки характеристик иерархической сети доступа с учётом повторных вызовов // Т-Сотт: Телекоммуникации и транспорт. 2015. Т. 9. № 3. С. 44-49.
11. Васильев А.П., Степанов С.Н. Построение и анализ математической модели с ожиданием и динамическим распределением канального ресурса при групповом поступлении запросов на передачу данных // Научные технологии в космических исследованиях Земли. 2017. Т. 9. № 4. С. 6-12.
12. Степанов С.Н., Саламех Н. Построение и анализ обобщенной модели передачи мультисервисного трафика реального времени в сетях связи // Научные технологии в космических исследованиях Земли. 2016. Т. 8. № 4. С. 37-44.
13. Степанов С.Н., Саламех Н. Построение и анализ двухпоточковой модели звена с конечным числом абонентов и возможностью внутренних блокировок // Т-Сотт: Телекоммуникации и транспорт. 2016. Т. 10. № 9. С. 30-37.
14. Васильев А.П., Степанов С.Н. Построение и анализ математической модели с динамическим распределением канального ресурса при групповом поступлении запросов на передачу данных // Т-Сотт: Телекоммуникации и транспорт. 2016. Т. 10. № 11. С. 55-59.
15. Степанов С.Н., Романов А.М., Осия Д.Л. Построение и анализ модели передачи данных на линии доступа от конечной группы абонентов // Т-Сотт: Телекоммуникации и транспорт. 2015. Т. 9. № 9. С. 29-34.

DEVELOPMENT AND ANALYSIS OF THE MATHEMATICAL MODEL OF MAINTENANCE OF CHALLENGES IN PERSPECTIVE CONTACT CENTERS

Maxim O. Shishkin

*Student of group M091801(72), MTUCI
mackschischkin1@yandex.ru*

Mikhail S. Stepanov

*MTUCI, PhD., associate professor of SSiSK department
mihstep@yandex.ru*

Keywords: *contact center, mathematical model, assessment of the quality of work of the contact center, calculation of the number of operators, calculation of the share of lost requests.*

Tasks for the implementation of the mathematical model of the contact center. A mathematical model of a promising contact center is presented. Model analysis and implementation of one of the algorithmic programming languages. Demonstration and verification of the correct operation of this model. Evaluation of the quality of the contact center by a numerical example. Calculation of the average number of employed operators and the percentage of lost requests as a percentage.

ОБЗОР МИКРОСХЕМ ASIC И FPGA И ИХ РОЛЬ В СИСТЕМЕ МОНИТОРИНГА КОМПЬЮТЕРНЫХ СЕТЕЙ И АНАЛИЗЕ ПАКЕТОВ

Акопян Ваник Артакович
студент группы M091801(72), МТУСИ
ivanivan9668@gmail.com

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

Ключевые слова: ASIC, FPGA, sFlow, DPI, глубокий анализ пакетов, Allot Communication, NBAR, система мониторинга.

Рассмотрены микросхемы ASIC и FPGA и их отличие. Исследуется роль данных микросхем в системе мониторинга компьютерных сетей. Исследуется понятие «глубокого анализа пакетов» и его реализация через микросхемы.

Введение

В настоящее время в сети Интернет передаётся огромное количество данных. Возникает необходимость анализировать трафик и производить мониторинг компьютерной сети той или иной организации. Чаще всего данные действия предпринимаются с целью выявления нежелательного трафика и несанкционированного доступа. Анализ пакетов и мониторинг сетей можно производить с помощью специальных микросхем, которые выполняют определённую программу, и, следовательно, позволяют производить вычисления намного быстрее. Наиболее известные микросхемы это ASIC и FPGA.

О микросхемах ASIC и FPGA

ASIC (аббревиатура от англ. application-specific integrated circuit, «интегральная схема специального назначения») — интегральная схема, специализированная для решения конкретной задачи. В отличие от обычных интегральных схем общего назначения, специализированные интегральные схемы применяются в конкретном устройстве и выполняют строго ограниченные функции, характерные только для данного устройства, следовательно, выполнение функций происходит быстрее. Примером ASIC может являться микросхема, разработанная исключительно для управления мобильным телефоном, микросхемы аппаратного кодирования/декодирования аудио- и видеосигналов (сигнальные процессоры). [1]

Преимущества:

- высокая производительность;
- низкое энергопотребление;
- небольшой размер;
- низкая потребляемая мощность.

Недостатки:

- не перепрограммируемая;
- узкоспециализированная;
- проблемы с масштабируемостью.

На рисунке 1 изображена общая структура ASIC.

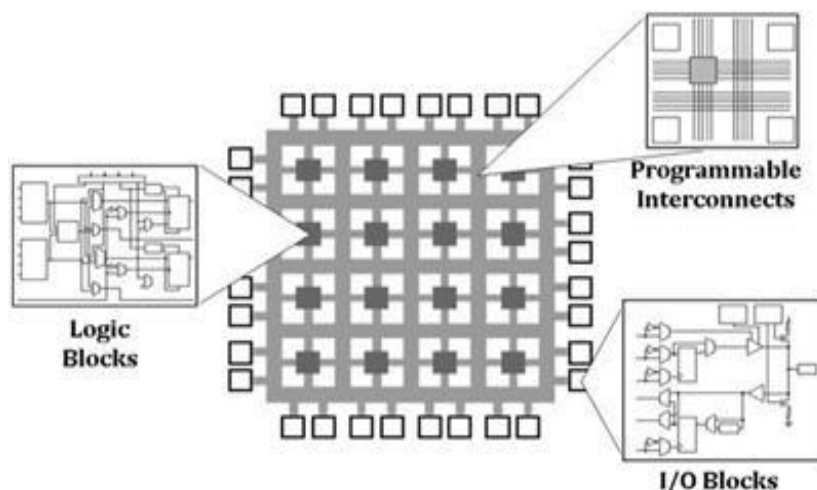


Рис. 1. Общая структура интегральной схемы специального назначения (ASIC)

ASIC разделяется на несколько видов:

- микросхемы со стандартной конструкцией ячеек (Standard Cell) – суть заключается в том, каждый производитель ASIC может создавать функциональные блоки с известными электрическими характеристиками, такими как задержка распространения, емкость и индуктивность. Конструкция стандартной ячейки – это использование этих функциональных блоков для достижения хороших электрических характеристик;
- gate-матрицы (Gate Array) – это способ изготовления, при котором слои, каждый из которых состоит из транзисторов и других активных устройств, являются заранее определенными, не подключаются до стадии процесса изготовления. Процесс физического проектирования определяет взаимосвязи этих слоев для конечного устройства. Для большинства производителей ASIC он состоит из двух-девяти слоев металла. Единовременные затраты на проектирование намного ниже, чем у полностью нестандартных конструкций. Достигается компромисс между быстрым проектированием и производительностью;
- полностью настраиваемый (Full-Custom). Преимущества – уменьшенная площадь, повышение производительности, а также возможность интеграции аналоговых компонентов и других предварительно разработанных компонентов, таких как ядра микропроцессора, которые образуют систему на кристалле (электронная схема, выполняющая функции целого устройства (например, компьютера) и размещенная на одной интегральной схеме). Недостатки – увеличение времени проектирования и изготовления.

Этапы производства микросхем ASIC:

1. спецификация – документ, который описывает работу будущей микросхемы на бумаге;
2. проектирование – реализация замысла проекта путем создания его описания на уровне регистровой передачи (RTL);
3. верификация – подтверждение того, что RTL-код соответствует первоначальным намерениям;
4. имплементация – синтезирование RTL-кода в стандартные 32-нм ячейки. Размещение и трассировка;
5. производство – изготовление набора масок. Формирование каждого слоя на поверхности кремния;
6. отбраковка – проверка микросхем перед использованием;
7. монтаж – монтаж микросхемы в корпус.

Применение микросхем ASIC:

- устройства интернета вещей (IoT);
- сетевое оборудование (роутеры, коммутаторы, сетевые карты);
- робототехника;
- майнинг криптовалют и др.

В настоящее время производителями микросхем ASIC являются Broadcom, Vicor, Microchip, Phoenix Contact, TE Connectivity, OKW, Intel PSG (бывшая Altera), Optoway Technology и др.

Программируемая логическая интегральная схема (Field-Programmable Gate Array, FPGA) – разновидность микросхем, главным достоинством которых является возможность перенастройки внутренней архитектуры в процессе эксплуатации. Это позволяет при необходимости модифицировать логику работы устройства для оптимизации его целевых характеристик или же полностью переориентировать всю цифровую вычислительную систему на решение новых задач. При этом время, стоимость и сложность разработки, отладки и тестирования FPGA значительно ниже, чем для ASIC. В основе данного типа микросхем лежит двумерный массив конфигурируемых логических блоков (configurable logic block, CLB), каждый из которых реализует последовательную или комбинационную логику, например, простейшие логические вентили AND, OR, NOT, XOR или мультиплексор. При проектировании схемы между блоками устанавливаются связи – соединительные маршруты. Наряду с CLB, чип, как правило, содержит блоки ввода/вывода, блоки памяти с произв. доступом, синхронизирующие блоки и т. д. На рисунке 2 изображена общая структура FPGA.

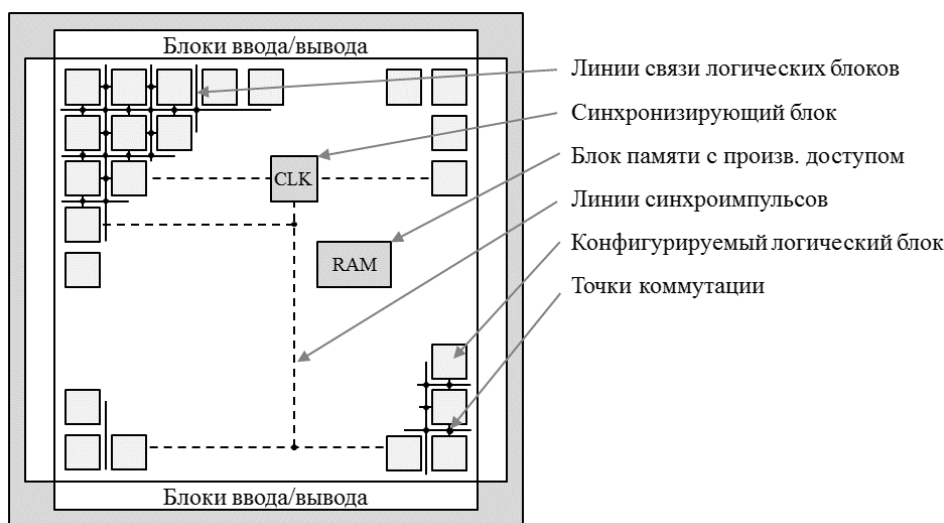


Рис. 2. Общая структура программируемой логической интегральной схемы (FPGA)

Процесс разработки FPGA включает в себя следующие шаги:

1. создание проекта интегральной схемы на языке описания аппаратуры (hardware description language, HDL; рассмотрен ниже) в виде текстовых HDL-файлов с изложением всех логических блоков, связей между ними, возможных ограничений;
2. выполнение тестовой RTL-симуляции, подразумевающей отладку HDL-кода логики микросхемы на так называемом уровне регистровых передач (register transfer level, RTL);
3. процесс логического синтеза, выполняемый средой разработки и заключающийся в преобразовании конструкций языка HDL в компоненты уровня логических вентилях и триггеров;
4. процесс реализации, подразделяющийся на три операции: трансляции (translate), отображения (technology mapping) и размещения и маршрутизации (placement and routing). Первая выполняет компиляцию нескольких файлов с описанием дизайна схемы в единый файл со списком всех блоков и соединений – netlist. Далее все вентили из списка отображаются на конфигурируемые логические блоки и блоки ввода/вывода интегральной схемы. Последняя операция определяет компоновку чипа, последовательно выполняя физическое размещение логических блоков и создание маршрутов для обеспечения прохождения различных сигналов между ними. Дополнительной операцией процесса реализации может служить задание статических временных параметров схемы, например, максимальной тактовой частоты;
5. создание конфигурационного файла на основе списка всех соединений, загружаемого в FPGA.

Роль микросхем в анализе пакетов

Анализ сетевого трафика – это очень обширная тема. Обычно под этим понимают совокупное название технологий и их реализаций по накоплению, обработке, классификации контролю модификации пакетов в зависимости от их содержимого в реальном времени.

Глубокий анализ пакетов DPI (Deep Packet Inspection или Deep Packet Processing) – это действия над пакетами, включающие их модификацию, фильтрацию и перенаправление. При этом системы DPI могут принимать решения не только по содержимому пакета, но и по косвенным признакам. Для этого может быть использован статистический анализ, расширенный список применений технологии классификации, приоритезации, маркировки и кэширования. DPI-технология разрабатывалась изначально для каналов связи со скоростью 100 Гбит/сек и для большого (порядка нескольких тысяч) идентифицированных приложений прежде всего седьмого уровня модели OSI [3].

Одна из тенденций последнего времени – централизация анализа, то есть реализация “DPI как сервис”. Суть концепции заключается в том, что при использовании большого количества различных средств анализа трафика (протокольные и кабельные анализаторы, оптимизаторы трафика, межсетевые экраны), имеет смысл вынести весь анализ на отдельное устройство. Оно будет выполнять полный разбор сетевых данных и рассылать результаты всем устройствам согласно их потребностей. Существуют программные реализации (Ipoque PACE, Widriver Contention Integration Engine), аппаратно-программные комплексы, привязанные к аппаратуре (Cisco NBAR), аппаратно – программные реализации отдельных производителей (Allot Communication).

Устройства компании Allot Communications серии NetEnforcer отслеживают сетевой трафик отдельных приложений и пользователей и управляют им, обеспечивая интеллектуальную оптимизацию услуг доступа [2].

Данные программно-аппаратные средства обладают следующими функциями:

- перенаправление трафика: трафик перенаправляется в сервисы, работающие на блейд-модулях (также блейд-сервер, компьютерный сервер с компонентами, которые помещаются в «корзину» для уменьшения занимаемого пространства) в платформе с поддержкой «горячей» замены, в реальном времени или развёртывание на внешних системах;
- сбор данных: происходит сбор и экспорт данных, сессий и отчётов использования сети в системы анализа и биллинга;
- обнаружение тетеринга (совместного доступа к мобильному интернету): выявление тетеринга, и применение политик оператора в отношении тетеринга в реальном времени.

В NetEnforcer реализованы ряд дополнительных функций:

- реализация технологии DART (Dynamic Actionable Recognition Technology – технология распознавания трафика не только по протоколам прикладного уровня или по приложениям, но и по принадлежности трафика тому или иному пользователю, по производителю устройства, по контексту использования приложений и т.д.);
- обработка асимметричных потоков (передаваемые с разными скоростями в прямом и обратном направлениях);
- поддержка протокола OpenFlow и т.д.

Procera Networks предлагает решения, основанные на их версии глубокой проверки пакетов (DPI), называемой Datastream Recognition Definition Language (языком определения распознавания потока данных) (DRDL). DRDL использует свойства приложений для целей идентификации, и это позволяет операторам управлять абонентским трафиком на основе приложения, подобно многим продуктам DPI. Procera предлагает более 100 вариантов использования в аналитике, управлении трафиком, применении политик и соблюдении поставщиков услуг.

Cisco Networks. NBAR (Network Based Application Recognition) – механизм используемый в сетях передачи данных для распознавания потока данных (dataflow). Расширенные механизмы классификации NBAR2 с глубоким анализом потоков трафика (на уровнях L4-L7) позволяют идентифицировать более 1500 приложений, использующих как статические, так и динамические порты TCP/UDP с отслеживанием состояния. В отличие от NBAR версии 1, NBAR2 также поддерживает классификацию IPv6.

Кроме этого, NBAR2 позволяет администраторам создавать собственные сигнатуры для приложений, которые не включены в стандартную библиотеку. NBAR2 может применяться не только для обнаружения приложений и сбора статистики (количество пакетов, байт, bit rate и т.д.) по каждому идентифицированному приложению. Классифицировав приложения, можно применить политику качества обслуживания QoS для них, например, ограничить полосу пропускания для трафика bittorrent. На рисунке 3 изображена реализация NBAR.

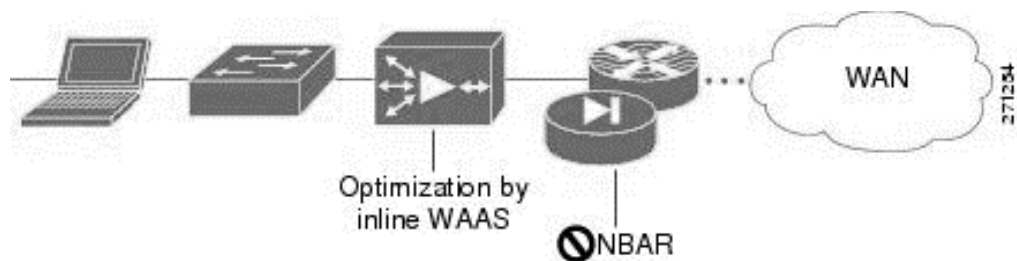


Рис. 3. Реализация NBAR

Аппаратное обеспечение, являющееся частью выше описываемых решений, использует специальные сетевые карты (NIC) для захвата трафика. Это интегральные схемы специального назначения – ASIC (Application Specific Integrated Circuit). Вся обработка буферов сетевых карт закладывается производителем в микросхемную базу (включая сигнатуры). Используются также специальные виды ассоциативной памяти (CAM) для параллельного сравнения своего содержимого с поступившем на вход префикса (n первых байт пакета), специального вида процессоры для обработки сетевого трафика (NP) с поиском по шаблонам, ключам, управлению очередями.

Роль микросхем в системе мониторинга

sFlow – стандарт для мониторинга компьютерных сетей, беспроводных сетей и сетевых устройств. [4]

Примеры применения sFlow:

- обнаружение, диагностика и устранение сетевых проблем;
- управление перегрузками в режиме реального времени;
- анализ для выявления несанкционированной сетевой активности и отслеживания источников атак "отказ в обслуживании".

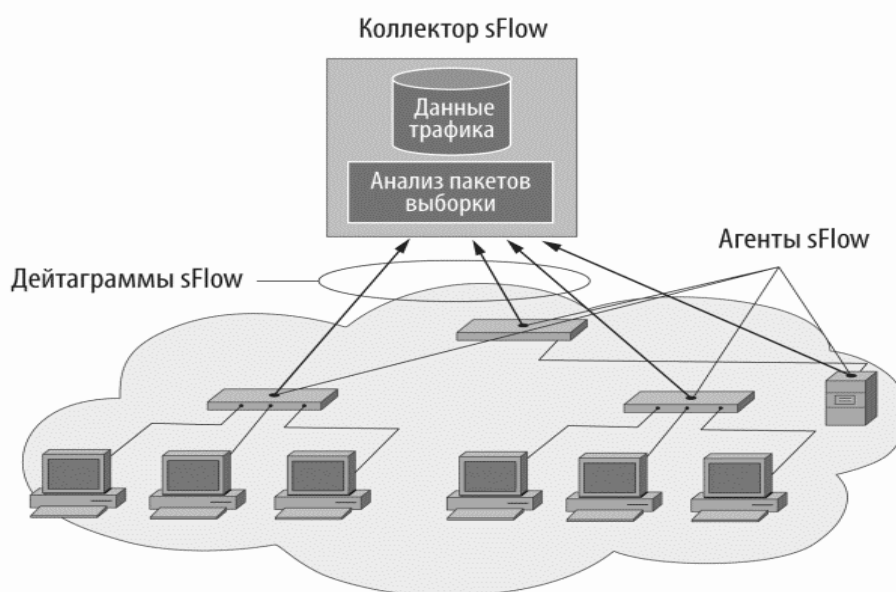


Рис. 4. Реализация протокола sFlow

Программный модуль, реализующий функции агента sFlow, работает как часть управляющего ПО, располагающегося на сетевом устройстве (рисунок 3). Когда выбирается пакет, его заголовок извлекается и помещается в дейтаграмму, которая включает заголовок, адреса и порты отправителя и получателя, статистику интерфейса и другую информацию, необходимую для анализа трафика на уровнях OSI от второго до седьмого. Сформированная дейтаграмма сразу же направляется по сети в центральный сборщик данных sFlow Collector. Один коллектор может обрабатывать данные более чем от 20 тыс. портов. Выборка пакетов в типичном случае выполняется с помощью коммутирующих или маршрутизирующих заказных специализированных микросхем (ASIC) со скоростью, которую позволяют физические соединения. Записывается также состояние элементов коммутационных и маршрутных таблиц, связанных с каждым пакетом выборки. На рисунке 4 показана реализация протокола sFlow.

Выводы

Микросхемы ASIC и FPGA играют важную роль в системе мониторинга компьютерных сетей и анализе пакетов. И если сравнивать данные микросхемы, то FPGA имеет большое количество портов ввода и вывода и возможность параллельной работы по нескольким задачам, что позволяет обработать большое количество информации в компьютерной сети. Но в отличие от ASIC у него есть такие недостатки, как высокая стоимость производства микросхемы (соответственно и конечная стоимость самой схемы), размеры чипа, меньшая производительность. Поэтому большинство аппаратных средств, которые применяются для анализа пакетов и мониторинга компьютерных сетей, используют микросхему ASIC.

Литература

1. Allot NetEnforcer & Service Gateway Documentation // Allot Communications, 2015, pp. 1-10.
2. *Jakub Svoboda*. Network Traffic Analysis with Deep Packet Inspection Method. Masatyk University Faculty of Informatics. Brno, Spring 2014, pp. 250-267.
3. *John Klein*. Digging Deeper Into DPI Network Visibility & Service Management // Allot Communications, 05.2007
4. *Прохоров Д.О., Креймер А.В., Трофлянин В.В.* Вопросы анализа производительности в инфокоммуникационных сетях. Шестая межвузовская студенческая конференция, 2017. С. 1-3.
5. *Беленькая М.Н., Малиновский С.Т., Яковенко Н.В.* Администрирование в информационных системах. Учебное пособие для вузов. 2-е издание, испр. и доп. М.: Горячая линия – Телеком, 2018. 408 с.

ASIC AND FPGA MICROCHIPS OVERVIEW AND THEIR ROLE IN COMPUTER NETWORK MONITORING SYSTEM AND PACKAGE ANALYSIS

Vanik A. Akopyan

student of group M091801(72), MTUCI

ivanivan9668@gmail.com

Marina N. Belenkaya

Associated professor of the department of multimedia networks and communication services, MTUCI

mn.belenkaya@mail.ru

Keywords: ASIC, FPGA, sFlow, DPI, deep packet inspection, Allot Communication, NBAR, monitoring system.

ASIC and FPGA chips and their difference are considered. The role of these chips in the monitoring system is investigated. The concept of "deep packet analysis" and its implementation through chips is investigated.

ОБЗОР ФУНКЦИОНАЛЬНЫХ ВОЗМОЖНОСТЕЙ ПРОГРАММНОГО ПРОДУКТА RIVERBED® STEELCENTRAL™ NETPROFILER

*Кошечаров Дмитрий Яковлевич,
студент группы М091801(72) МГУСИ*

kosheparovdy@yandex.ru

*Беленькая Марина Наумовна,
МГУСИ, доцент кафедры МСuУС
mn.belenkaya@mail.ru*

Ключевые слова: DPI, QoS, интернет-трафик, анализ трафика, Riverbed Technologies.

Рассмотрено определение понятия DPI, представлены основные методы анализа интернет-трафика с помощью систем DPI, рассмотрены методы реализации механизма углублённой проверки пакетов, приведены основные направления использования технологии углублённой обработки пакетов, рассмотрено понятие QoS, рассмотрена реализация технологии DPI на программном уровне компании Riverbed Technologies.

На данный момент анализ сетевого трафика является одним из актуальных направлений исследования и разработки. Под анализом сетевого трафика принято понимать технологии и их реализации по накоплению, обработке, классификации, контролю и модификации пакетов в зависимости от их содержимого в реальном времени. Одной из технологий анализа сетевого трафика является DPI.

DPI (Deep Packet Inspection, углублённая проверка пакетов) – это технология накопления статистических данных о потоках и проверки и фильтрации сетевых пакетов по полному их содержимому, а также последующих действий: модификации и перенаправления сетевых пакетов.[1, 5-7] При этом решение, принимаемое системой DPI, может основываться не только на прямом анализе содержимого пакетов, но и с помощью таких методов, как проверка на соответствие заранее определённым правилам, анализ сигнатур, анализ поведения, статистический анализ. Классификация проанализированных потоков может осуществляться как по типу приложения, так и по типу конкретного протокола прикладного уровня модели OSI или, в отдельных случаях, по конкретному приложению.[2]

На данный момент системы DPI являются стандартом де-факто на анализ трафика и защиту от несанкционированного доступа. Соблюдение стандартов DPI обеспечивается рекомендациями Международного Союза Электросвязи Y.2770 и Y.2771 и рядом других рекомендаций.

Решения DPI могут быть представлены в разной форме: это могут быть аппаратно-программные решения отдельных производителей, аппаратно-программные решения, привязанные к оборудованию конкретного производителя, и программные решения. Рассмотрим последний пункт более подробно.

Riverbed® SteelCentral™ NetProfiler представляет собой программный продукт, предназначенный для работы в среде корпоративной сети. Он предоставляет глобальную аналитику потоков в сети, которую можно использовать для быстрого просмотра производительности приложений и устранения неполадок, и позволяет обеспечить активный мониторинг сети, анализ и формирование отчётов на основе полученных данных.

ПО NetProfiler запускается на выделенном сервере под управлением гипервизора VMWare EXSi версии не менее 6.0 и, в частности, под управлением ОС семейства *NIX (например, FreeBSD) либо семейства ОС Windows Server (например, Windows Server 2012 R2 Standard). Полный список требований представлен в табл. 1.

Таблица 1

Системные требования NetProfiler

| Конфигурация | F1 | F2 | F3 | F4 | F5 | F6 | F7 | F8 | F9 | F10 | F11 |
|--------------|---|----|----|----|--|----|--|----|---|--|--|
| Процессор | Минимум 4 CPU Рекомендуемая тактовая частота - 2.6 GHz | | | | Минимум 4 CPU / Рекомендация - 8 CPU Рекомендуемая тактовая частота - 2.6 GHz | | Рекомендация - 8 CPU Рекомендуемая тактовая частота - 2.6 GHz | | Минимум 8 CPU / Рекомендация - 12 CPU Рекомендуемая тактовая частота - 2.6 GHz | | Минимум 12 CPU / Рекомендация — 16 CPU Рекомендуемая тактовая частота - 2.6 GHz |
| RAM | 8 GB - минимум / 12 GB - рекомендация | | | | 12 GB - минимум / 16 GB - рекомендация | | 16 GB - минимум / 32 GB - рекомендация | | 32 GB - минимум / 48 GB - рекомендация | 48 GB - минимум / 64 GB - рекомендация | 64 GB - минимум / 96 GB - рекомендация |

Сервер с установленным ПО NetProfiler включается в сегмент серверов корпоративной сети. Сервер NetProfiler должен быть включен в схему сети после сетевого оборудования, осуществляющего подключение к другим сегментам сети. Также в сегменте сети должны присутствовать резервный сервер NetProfiler, выделенный сервер для размещения модуля App Response (требования к серверу совпадают с требованиями для сервера NetProfiler) и сервер БД, предназначенный для хранения данных NetProfiler (ОС сервера — MS Windows Server, СУБД — MS SQL Server). Пример схемы сети представлен на рис. 1.

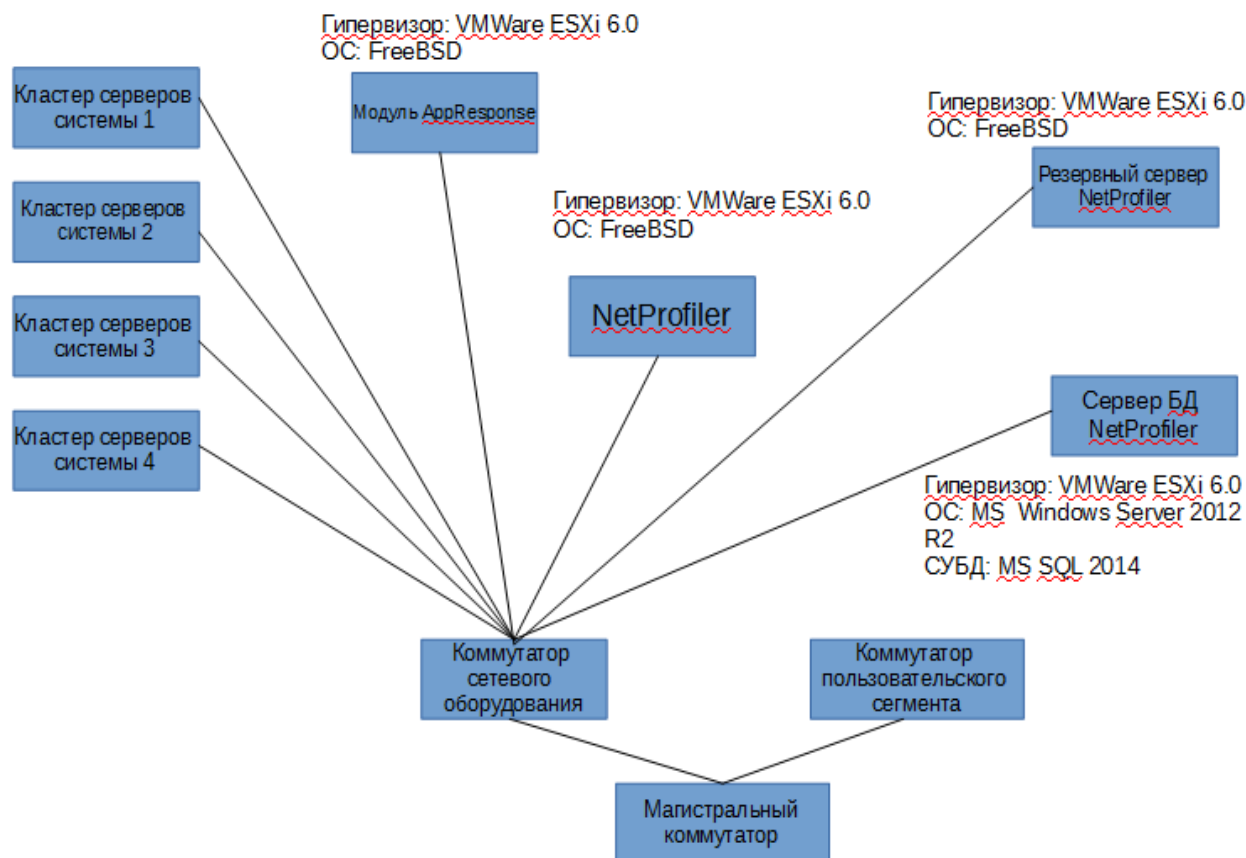


Рис. 1. Схема сети при включении ПО NetProfiler

NetProfiler собирает данные трафика, приложений и пользователей из нескольких источников, получая данные различных сетевых протоколов (например, NetFlow), и объединяет их, чтобы сообщать как подробную, так и агрегированную информацию о хостах, портах, интерфейсах, приложениях и пользователях. Также эти данные используются для идентификации и предупреждения изменений в поведении сети. В зависимости от конфигурации, NetProfiler может обрабатывать от 15000 до 3000000 потоков. Данные о потоках хранятся в дисковом пространстве в виде файлов захвата пакетов в формате *.pcap.

Обработка и анализ файлов захвата пакетов производится с помощью модуля App Response.

Методы анализа NetProfiler включают в себя и сигнатурный анализ, и сопоставления на четвёртом уровне модели OSI.

Сигнатурный анализ производится следующим образом: модуль App Response анализирует файлы захвата пакетов, сопоставляя захваченные потоки с образцами из базы сигнатур, содержащей более 1300 определений протоколов и приложений. Средства модуля App Response также позволяют в некоторых случаях производить диссекцию пакетов, защищенных с помощью SSL (путём подмены сертификатов) и инкапсуляции трафика (L2TP, GRE). Примеры распознаваемых протоколов приведены в табл. 2.

Таблица 2

Примеры распознаваемых приложений и протоколов 7-ого уровня модели OSI

| | |
|--------------------------|--|
| P2P (Peer-to-Peer) | BitTorrent, eDonkey, Gnutella, eMule |
| VoIP | SIP, Skype, WhatsApp, Viber |
| Web | HTTP, мобильные приложения и приложения социальных сетей |
| Потоковые | YouTube, RTMP |
| Обмен файлами | FTP, HTTP |
| Инкапсуляция трафика | L2TP, MPLS |
| Корпоративные приложения | Citrix, Lotus Notes, SAP |

Также существует возможность самостоятельного создания и добавления сигнатур.

Сопоставление на четвёртом уровне модели OSI. Приложение может быть определено путем сопоставления узлов и портов с именем приложения. Когда NetProfiler обнаруживает поток, который соответствует определённым хостам или портам, он классифицирует этот поток как трафик конкретного приложения.

Программное обеспечение NetProfiler также позволяет контролировать соблюдение принципов QoS. QoS (Quality of Service, качество обслуживания) описывает сеть в терминах задержки и ее вариации и полосы сигнала.[3]

NetProfiler обеспечивает единый мониторинг и отчетность об эффективности формирования политики качества, развернутой на устройствах по всей сети. Для этого производится сбор двух типов информации с устройств:

- атрибуты трафика – идентификатор классификации QoS, идентификатор приложения и маркировка DSCP. Устройства должны быть настроены таким образом, чтобы включать эту информацию в данные SteelFlow Net или NetFlow v9, которые они отправляют в NetProfiler.
- конфигурационная информация – конфигурационные параметры политики настройки QoS. Для опроса устройств для информации о конфигурации QoS NetProfiler использует запросы REST API один раз каждые 24 часа, чтобы синхронизироваться с фактическими настройками.

На основе полученных данных NetProfiler позволяет производить аналитику и мониторинг таких показателей, как:

- Активные подключения;
- Пропускная способность;
- Новые подключения;
- Количество сбросов TCP;
- Полоса пропускания;

- Средняя пропускная способность приложения на соединение;
- Средняя продолжительность соединения;
- Время отклика;
- Джиттер;
- MOS;
- Пакетная потеря;
- Процент потери пакетов.[4]

Выводы

Решение SteelCentral NetProfiler компании Riverbed Technologies соответствует стандартам требований к системам DPI и является крайне перспективным решением для корпоративных сетей. Изучение программного обеспечения NetProfiler компании Riverbed Technologies является необходимым для понимания принципов программной реализации технологии DPI и последующей разработки собственных систем DPI.

Литература

1. Riverbed SteelCentral NetProfiler Documentation // Riverbed Technologies.
2. *Jakub Svoboda*. Network Traffic Analysis with Deep Packet Inspection Method. Masatyk University Faculty of Informatics. Brno. Spring, 2014.
3. *Гетьман А. И., Евстропов Е. Ф., Маркин Ю. В.* Анализ сетевого трафика в режиме реального времени: обзор прикладных задач, подходов и решений. Препринт ИСП РАН 28, 2015 г.
4. *Беленькая М.Н., Малиновский С.Т., Яковенко Н.В.* Администрирование в информационных системах. Учебное пособие для вузов. М.: Горячая линия – Телеком, 2018.
5. *Беленькая М.Н., Зайцев Е.С., Акоюн В.А., Кошепаров Д.Я.* Обзор методов анализа сетевого трафика средствами DPI // Труды Северо-Кавказского филиала Московского технического университета связи и информатики. 2018. № 1. С. 15-20.
6. *Прохоров Д.О., Беленькая М.Н.* Исследование технологии глубокого анализа пакетов DPI для применения в корпоративных сетях // Телекоммуникации и информационные технологии. 2017. Т. 4. № 2. С. 83-88.
7. *Кошепаров Д.Я., Зайцев Е.С., Акоюн В.А., Беленькая М.Н.* Обзор технологии DPI // Телекоммуникации и информационные технологии. 2018. Т. 5. № 1. С. 136-140.

DPI SYSTEM FUNCTIONALITY OF THE RIVERBED® STEELCENTRAL™ NETPROFILER OVERVIEW

Dmitry Ya. Kosheparov

student of group M091801(72), MTUCI

kosheparovdy@yandex.ru

Marina N. Belenkaya

MTUCI, associated professor of the department of multimedia networks and communication services

mn.belenkaya@mail.ru

Key words: *DPI, QoS, internet traffic, traffic analysis, Riverbed Technologies.*

Abstract.

Here is given the definition of the DPI concept, the main methods of analyzing Internet traffic using DPI systems are presented, the methods for implementing the mechanism of in-depth packet inspection are presented, the main directions of using deep packet inspection technology are considered, the concept of QoS is reviewed, and overview of the program realization of DPI on software level by Riverbed Tech.

ОПТИМИЗАЦИЯ ПРОЦЕССА ОПЛАТЫ В РЕСТОРАННОМ БИЗНЕСЕ ПРИ ПОМОЩИ СОВРЕМЕННЫХ РАСПРЕДЕЛЕННЫХ ИТ-РЕШЕНИЙ

Ефименков Демид Дмитриевич
студент группы ЗМАС1701 МТУСИ
efimenkovdd@gmail.com

Депутатов Евгений Алексеевич
студент группы ЗМАС1701 МТУСИ
ev.dep@yandex.ru

Воронцов Юрий Алексеевич
МТУСИ, д.т.н., профессор кафедры ИС
yvorontsov.1943@mail.ru

Ключевые слова: процесс оплаты, POS-терминалы, распределенный кассовый сервис, распределённые системы, системы автоматизации.

В статье рассмотрена оптимизация системы автоматизации оплаты товаров и услуг в ресторанном бизнесе. Технология распределенного кассового сервиса как современное решение реализации системы оплаты.

В ресторанном бизнесе существует множество точек обслуживания клиента. Одним из процессов является прием оплаты за указанную услугу или проданный товар. В большинстве систем оплаты используется прием оплаты банковскими картами при помощи переносных POS-терминалов. Текущая технология имеет множество недостатков как для клиента, так и для бизнеса такие как: долгое время обслуживания, большие затраты на обслуживающий персонал и на обслуживание аппаратной части системы оплаты, низкое качество статистических данных по работе предприятия. Рассмотренный в данной статье новый подход к реализации системы оплаты позволит устранить большинство недостатков используемых в данный момент систем.

При реализации многих точек оплаты товаров и услуг клиентами в стандартных архитектурах применяется количество объектов обработки равным количеству терминалов.

При применении технологии распределенного кассового сервиса (далее технология РКС) можно сократить затраты на использование оборудования обработки оплаты и увеличить уровень клиентского сервиса за счет увеличения количества точек оплаты, при этом, не увеличивая затрат на закупку оборудования обработки оплаты. Большим преимуществом использования технологии РКС является централизованной сбор статистики по обороту компании и удобства ведения налоговой отчетности.

Данная архитектура [1,2] (рисунок 1) позволит сократить затраты на аппаратную часть реализации системы автоматизации оплаты товаров и услуг.

Затраты [4,5] на аппаратную часть реализации системы РКС приведены в таблице 1.

Стандартные системы кассового обслуживания не позволяют принимать online оплату, тем самым, не позволяя бизнесу принимать платежи с ресурсов общего доступа. Технология РКС включает в себя API для безопасной интеграции с любой необходимой для бизнеса точкой оплаты, таких как: Мобильное приложение, ресурс общего доступа в сети интернет и терминалы оплаты.

У современной тенденции к оплате в сети интернет есть большое количество преимуществ:

1. Отсутствие затрат на покупку и обслуживание POS – терминала;
2. Более низкие затраты на оплату комиссий за online эквайринг;
3. Online оплата позволяет привлечь бизнесу большее количество клиентов (покупателей).

Таблица 1

Затраты на реализацию аппаратной части системы РКС

| Вариант систем оплаты | Сервер обработки оплаты | | Сервер приложения | | POS-терминал | | Контрольно-кассовая техника | | Общие затраты на аппаратную структуру |
|-------------------------|-------------------------|--------|-------------------|--------|--------------|--------|-----------------------------|--------|---------------------------------------|
| | Цена | Кол-во | Цена | Кол-во | Цена | Кол-во | Цена | Кол-во | |
| Текущие системы оплаты | 140000 | 1 | 30000 | 7 | 15000 | 7 | 17000 | 7 | 574000 |
| Технология оплаты РКС | | 1 | | 2 | | | | 7 | |
| Экономия – 235000 (41%) | | | | | | | | | |

В случае выхода из строя аппаратного обеспечения технологии РКС, увеличивается скорость, и простота замены оборудования. Время простоя системы стремится к минимуму, тем самым обеспечивая безотказную работу в случае непредвиденных ошибок оборудования. Для замены узлов оборудования требуется меньшая квалификация технического сотрудника.

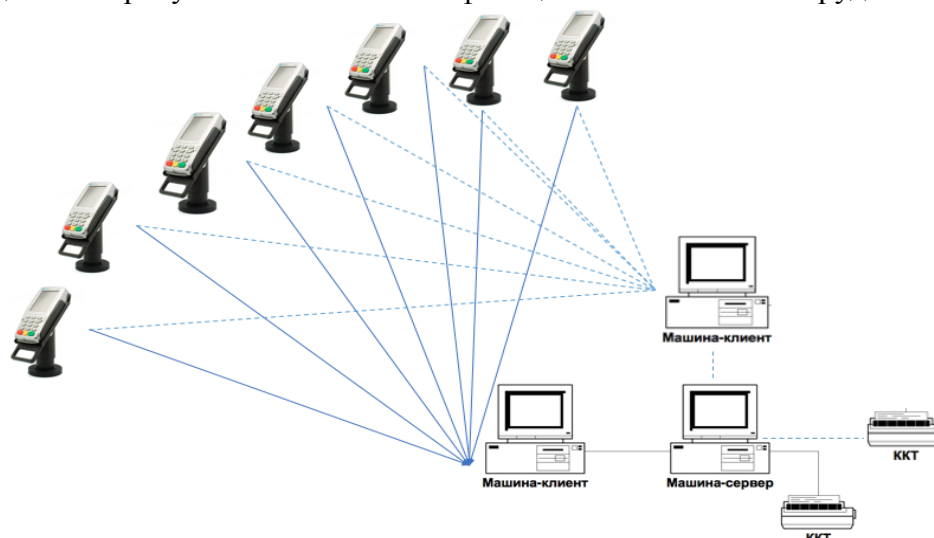


Рис. 1. Схема реализации распределенного кассового сервиса

Использование данной технологии построения сети позволяет экономить время выдачи и ресурсы создания бумажной версии фискального документа. Чеки оплаты могут быть отправлены при помощи СМС или электронной почты на контактные данные клиента. Данный вид предоставления информации долговечнее в хранении и навсегда сохраняется в online кабинете клиента (не «выцветает» со временем и исключает возможность потери клиентом).

Технология РКС совместима с онлайн сервисами отчетности [3] и позволяет получить статистику данных со всех подключенных филиалов компаний. Информация может быть заранее агрегирована и доступна в online времени по любому месту расчетов внутри компании. Использование локальных точек оплат позволяет сэкономить рабочее время персонала необходимое для обслуживания клиентов во время приема оплаты товаров и услуг. Экономия на заработной плате может достигать более 200%, не теряя качество клиентского сервиса.

Заключение

Рассмотренная в статье технология РКС имея небольшое количество точек оплаты, позволяет сэкономить на этапе внедрения системы более 40% затрат на реализацию системы оплаты. Повышается качество обслуживания клиентов за счет уменьшения времени на оплату. После

внедрения данной системы появляется возможность интеграции с современными методами приема оплаты из общих ресурсов оплаты сети интернет. Решение РКС позволяет уменьшить затраты бизнеса на прием оплаты и повышает качество клиентского сервиса[6].

Литература

1. Драйвер ФР версии А4.1. Руководство программиста. Штрих-М [Электронный ресурс} URL: ftp://ftp.shtrih-m.ru/download/FR/shtrih-fr-k/doc/Driver4_6.pdf.
2. Архитектура распределённых приложений [Электронный ресурс} URL: <https://www.itweek.ru/infrastructure/article/detail.php?ID=66147>.
3. *Воронцов Ю.А., Лопусов Б.А., Сергейчук С.К.* Информационные системы в административном управлении предприятиями связи. М.: Радио и связь, 2004.
4. *Воронцов Ю.А.* Техничко-экономическое обоснование эффективности проектов информационных систем. М.: Инсвязьиздат, 2008.
5. *Воронцов Ю.А.* Практическая реализация проектов информационных систем: Монография. М.: Инсвязьиздат, 2009.
6. *Еремичев В.И.* Проектирование информационных систем для предприятий связи, 2005.

OPTIMIZATION OF PAYMENT PROCESS IN RESTAURANT BUSINESS WITH THE HELP OF MODERN DISTRIBUTED IT-SOLUTIONS.

Demid D. Efimenkov

*Student of group ZMAS1701, MTUCI
efimenkovdd@gmail.com*

Evgeniy A. Deputatov

*Student of group ZMAS1701, MTUCI
ev.dep@yandex.ru*

Yuri A. Voroncov

*MTUCI, d.t.s., professor of ISdepartment
yvorontsov.1943@mail.ru*

Keywords: *payment process, POS-terminals, distributed cash service distributed systems, automation systems.*

Optimization of the automation system of payment for goods and services in the restaurant business. Technology of distributed cash service as a modern solution for the implementation of the payment system.

БИНАРНАЯ КЛАССИФИКАЦИЯ КОМПЬЮТЕРНЫХ АТАК НА ИНФОРМАЦИОННЫЕ РЕСУРСЫ ПРИ ПОМОЩИ НЕЧЁТКОЙ ЛОГИКИ

Груздев Сергей Павлович
магистрант группы М091801(71) МТУСИ
House.HD@yandex.ru
Шелухин Олег Иванович
МТУСИ, д.т.н., профессор кафедры ИБ
sheluhin@mail.ru

Ключевые слова: KDD, NSL-KDD, система обнаружений вторжений, нечёткая логика, система нечёткого вывода Такаги-Сугено, бинарная классификация, нормальные данные, атака, признаки сетевого трафика.

Целью работы являлся поиск минимального количества признаков сетевого трафика, для создания нечёткой системы, при помощи которой будет проводиться бинарная классификация сетевого трафика. Эксперименты с созданной системой обнаружения вторжений с использованием нечеткой логики выполнялись с помощью набора данных компьютерных атак NSL-KDD. Экспериментальные результаты показали, что созданная система бинарной классификации с использованием нечёткой логики обеспечивает достоверность правильной классификации компьютерных атак на уровне 86,6%.

Постановка задачи.

В связи с быстрым развитием глобальной сети интернет в повседневной жизни, безопасность компьютерных систем стала одной из важных проблем для защиты данных и информации от злоумышленников. Система обнаружения вторжений (СОВ) отвечает за мониторинг сетевого трафика на любые подозрительные события [6,7] и поднимает тревогу, чтобы предпринять надлежащие действия против вторжения. Часто СОВ представляет собой бинарный классификатор, реализуемый методами машинного обучения. Подобные СОВ классифицируют данные на два класса, на “Нормальные данные” и “Атака”. Одним из способов создания классификатора является использование нечёткой логики [1].

Анализ результатов проведенных исследований показал, что обычно для создания СОВ на основе нечёткой логики используют систему нечёткого вывода Маддани и от 20 до 34 признаков сетевых атак. Однако использование такого большого количества признаков для проведения бинарной классификации существенно усложняет процедуру обработки. Для упрощения алгоритмов обработки при обеспечении заданного качества классификации предлагается минимизировать количество используемых наиболее значимых признаков сетевых атак. Используя отобранные признаки, предлагается построить нечёткую СОВ на основе системы нечёткого вывода Такаги-Сугено имеющей преимущества перед системой Маддани. Снижение числа признаков позволяет снизить необходимые вычислительные мощности и увеличить скорость проведения бинарной классификации.

Анализ баз данных аномальных вторжений.

Во многих работах для тестирования разрабатываемых и уже существующих СОВ используется набор данных компьютерных атак KDD (Knowledge Discovery in Databases). Однако поскольку KDD содержит повторяющиеся записи, в работе использовалась версия набора KDD, содержащая в себе только уникальные записи NSL-KDD. Данный набор содержит в обучающем наборе данных 22 типа атак, а в тестирующем наборе - 37 типов атак. Набор данных NSL-KDD состоит из записей, которые являются образами сетевых соединений, т.е. образ последовательности пакетов данных за некоторое время, в течение которого они передаются от IP-адреса источника к

IP-адресу получателя по определенному протоколу. Каждая запись содержит в себе 41 признак сетевого трафика (табл. 1) и помечена либо как «Нормальные данные» или как «Атака», с определением её типа, а также имеет метку сложности правильной принадлежности данной записи к правильному классу, и которая представляет собой число от 0 до 21.

Таблица 1

Список признаков набора данных KDD.

| № | Признаки | Описание | Тип |
|---|-----------------------------|--|-------------|
| Характеристики TCP-соединения | | | |
| 1 | duration | Продолжительность соединения (секунды) | непрерывный |
| 2 | protocol_type | Протокол транспортного уровня | символьный |
| 3 | service | Сервис прикладного уровня | символьный |
| 4 | flag | Статус соединения | символьный |
| 5 | src_bytes | Количество байтов данных от источника к получателю | непрерывный |
| 6 | dst_bytes | Количество байтов данных от получателя к источнику | непрерывный |
| 7 | land | 1, если соединение от / к тому же хосту / порту; 0 иначе | символьный |
| 8 | wrong_fragment | Число неправильных фрагментов | непрерывный |
| 9 | urgent | Число срочных пакетов | непрерывный |
| Характеристики сеанса | | | |
| 10 | hot | Число «горячих» индикаторов | непрерывный |
| 11 | num_failed_logins | Число неудачных попыток входа | непрерывный |
| 12 | logged_in | 1 Успешный вход, иначе 0 | символьный |
| 13 | num_compromized | Число «скомпрометированных» условий | непрерывный |
| 14 | root_shell | Доступ с административными полномочиями | непрерывный |
| 15 | Su_attempted | 1 если попытка ``su root`` предпринята; 0 в противном случае | непрерывный |
| 16 | num_root | Число попыток доступа с правами администратора | непрерывный |
| 17 | num_file_creations | Количество операций создания файла | непрерывный |
| 18 | num_shells | Число попыток использования командной строки | непрерывный |
| 19 | num_access_files | Число операций с файлами контроля доступа | непрерывный |
| 20 | num_outbound_cmd | Количество исходящих команд в сеансе ftp | непрерывный |
| 21 | is_hot_login | 1, если логин принадлежит списку ``hot``; 0 в противном случае | символьный |
| 22 | is_guest_login | 1, если логин является «гостевым»; 0 в противном случае | символьный |
| Статистика за 2 секунды / за 100 соединений | | | |
| 23 | count | Число соединений с совпадающим хостом | непрерывный |
| 24 | srv_count | Число соединений с совпадающим сервисом | непрерывный |
| 25 | error_rate | % соединения с ошибкой ``SYN`` | непрерывный |
| 26 | srv_error_rate | % соединения с ошибкой ``SYN`` | непрерывный |
| 27 | rerror_rate | % соединений с ошибкой ``REJ`` / % соединений с одинаковым исходным портом | непрерывный |
| 28 | srv_rerror_rate | % соединений с ошибкой ``REJ`` | непрерывный |
| 29 | same_srv_rate | % соединений с одинаковым сервисом | непрерывный |
| 30 | diff_srv_rate | % соединений с различным сервисом | непрерывный |
| 31 | srv_diff_host_rate | % соединений с различающимися хостами | непрерывный |
| 32 | dst_host_count | Число соединений с совпадающим хостом | непрерывный |
| 33 | dst_host_srv_count | Число соединений с совпадающим сервисом | непрерывный |
| 34 | dst_host_same_srv_rate | % соединений с одинаковым сервисом | непрерывный |
| 35 | dst_host_diff_srv_rate | % соединений с различным сервисом | непрерывный |
| 36 | dst_host_same_src_port_rate | % соединений с ошибкой ``REJ`` / % соединений с одинаковым исходным портом | непрерывный |

| | | | |
|----|-----------------------------|--|-------------|
| 37 | dst_host_srv_diff_host_rate | % соединений с различающимися хостами | непрерывный |
| 38 | dst_host_serror_rate | % соединения с ошибкой ``SYN`` | непрерывный |
| 39 | dst_host_srv_serror_rate | % соединений с ошибкой ``REJ`` | непрерывный |
| 40 | dst_host_rerror_rate | % соединений с ошибкой ``REJ`` / % соединений с одинаковым исходным портом | непрерывный |
| 41 | dst_host_srv_rerror_rate | % соединений с ошибкой ``REJ`` | непрерывный |

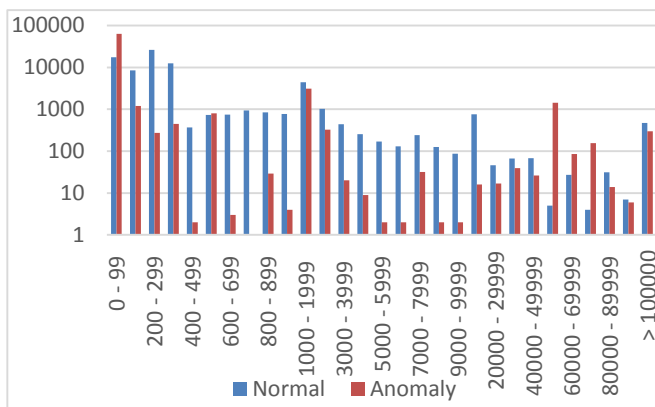
При помощи признаков, приведённых в табл. 1, создаётся образ соединения, который отражает основную информацию соединения, необходимую для проведения классификации. Образ соединения структурирован и занимает намного меньше места на диске по сравнению с самим набором пакетов соединения, благодаря этому с образом соединения удобнее работать и хранить его. Поскольку данный набор данных используется для проведения бинарной классификации при помощи нечёткой логики, производить обучение на конкретных записях не требуется. Модель обучается при помощи экспертов, которые создают правила на этапе создания системы нечёткой логики, основываясь на распределении значений признаков сетевого трафика. Поэтому для проведения бинарной классификации использовался объединённый набор данных NSL-KDD, состоящий из записей обучающего и тестирующего наборов данных NSL-KDD. Объединённый набор данных NSL-KDD состоит из 148517 записей.

Отбор признаков данных NSL-KDD

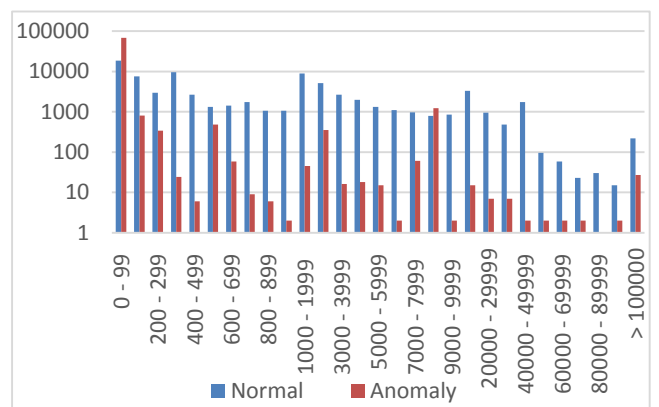
Основной целью выбора количества признаков является отбор наиболее значимых признаков для бинарной классификации. Для выбора признаков был использован алгоритм ранжирования Gain Ratio [2]. Результат ранжирования признаков сетевого трафика полученный с помощью алгоритма Gain Ratio следующий : 12, 26, 4, 25, 39, 6, 30, 38, 5, 29, 3, 37, 34, 33, 8, 35, 23, 31, 41, 32, 28, 27, 36, 16, 15, 2, 10, 13, 19, 1, 40, 18, 17, 24, 14, 22, 11, 20, 7, 9, 21. Номера признаков соответствуют номерам в таблице 1. Вычисленный ранг признаков следует в порядке убывания. Так, например номер признака 12 (logged_in) имеет самый высокий ранг, а номер признака 21 (hot_login) является наименее релевантным. Далее для 17 наиболее значимых признаков сетевого трафика были построены гистограммы функций распределения значений, на основе которых отображены наиболее подходящие сетевые признаки для создания СОВ на основе нечёткой логики.

На основе анализа результатов полученных в [3], видно, что, используя всего пять информативных признаков, можно получить достаточно эффективную систему обнаружения вторжений. Гистограммы отобранных признаков приведены на рис. 1.

Для разработанной системы нечёткой логики было создано 1024 набора правил, которые были составлены экспертами для обучения системы, основываясь на распределении значений признаков и зависимостями между значениями признаков. Анализируемые правила созданы таким образом, что используют все пять входных лингвистических переменных признаков сетевого трафика, так что каждому уникальному набору значений входных переменных ставится в соответствие единственное подходящее правило.



(a)



(б)

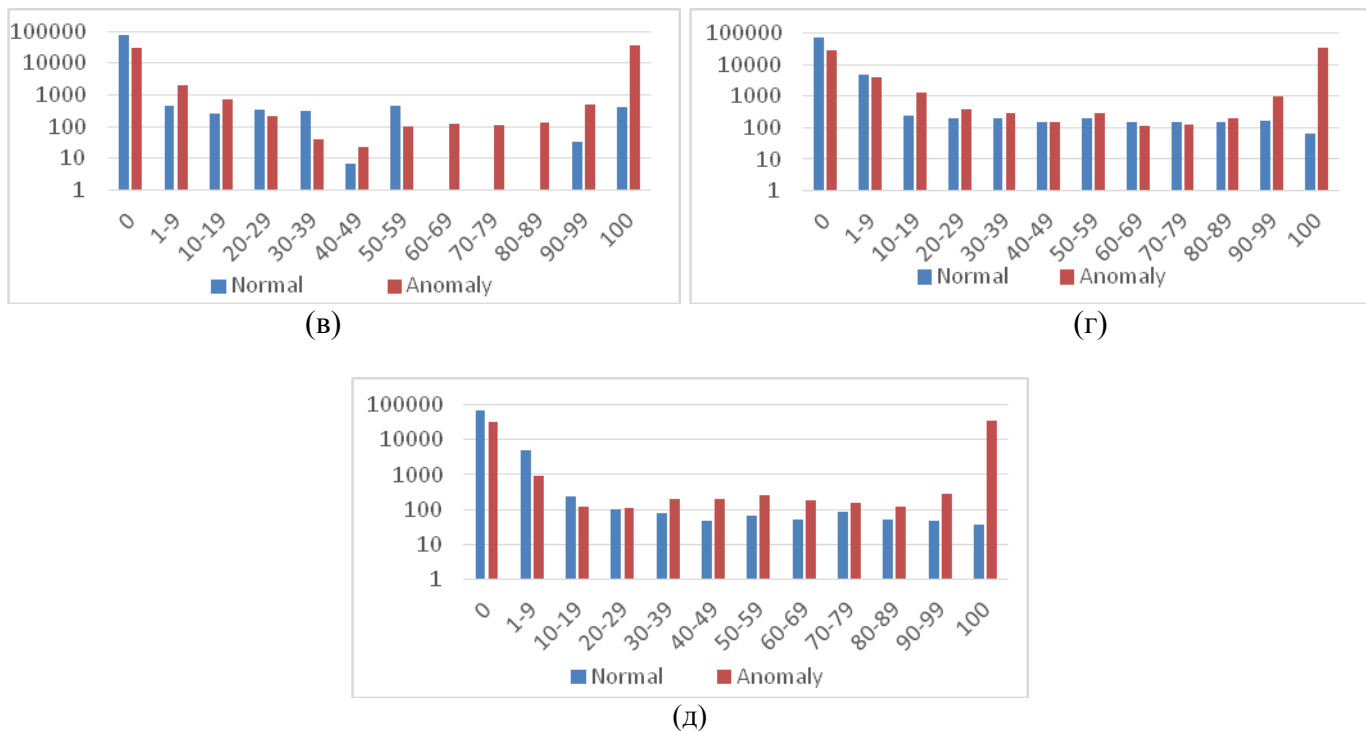


Рис. 1. Гистограммы функций распределения значений сетевых признаков: (а)Src_bytes, (б)Dst_bytes, (в) Error_rate, (г) Dst_host_error_rate, (д) Dst_host_srv_error_rate

Нечеткая логика

Для определения нечеткого множества вводятся понятия нечеткой и лингвистической переменной. Нечеткая переменная описывается кортежем: $\langle \alpha, X, A \rangle$, где α – название нечеткой переменной; X – область определения нечеткой переменной; A – нечеткое множество на множестве X . Лингвистическая переменная представляет собой кортеж из следующих элементов: $\langle \beta, T, X, G, M \rangle$, где β – название лингвистической переменной; T – множество её значений (терм-множество), элементы которого представляют собой названия нечетких переменных; X – область определения нечетких переменных (универсум); G – синтаксическое правило генерации новых термов; M – семантическое правило, ставящее в соответствие значению лингвистической переменной, образованной процедурой G , нечеткое подмножество множества X .

Нечетким высказыванием называют высказывания следующего вида: $\langle \beta \text{ есть } \alpha \rangle$, где β – имя лингвистической переменной; α – её значение, т.е. один из термов этой переменной, которому соответствует нечеткое подмножество множества X .

Для реализации алгоритма принятия решений с помощью нечёткого вывода необходимо описать правила, по которым осуществляется связь между различными входными характеристиками и выходным значением. Принятие решений осуществляется при помощи продукционных правил, которые состоят из нескольких посылок и заключения. Продукционное правило имеет следующий вид: **ЕСЛИ посылка [И посылка], ТО заключение**, где посылка и заключение являются нечёткими высказываниями.

Кроме логического оператора “И”, используется логическая связка “ИЛИ”. Подусловием называется каждое из нечетких высказываний в условии любого правила. Аналогичным образом, любое из высказываний в заключении правила называется подзаключением. Характеристикой нечеткого множества является функция принадлежности, которая указывает степень принадлежности элемента x к подмножеству A . Множество M называют множеством принадлежностей. Если $M = \{0, 1\}$, тогда нечеткое подмножество A может рассматриваться как обычное или четкое множество. Функции принадлежности могут быть определены разными способами. На практике чаще всего используются треугольные, трапециевидные, гауссовы и другие функции принадлежности. На рисунке 2 приведен пример представления четырёх значений лингвистической переменной VL, L, M, H с треугольными функциями распределения.

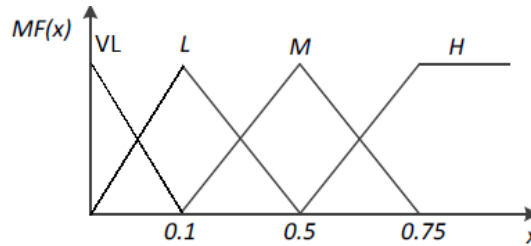


Рис. 2. Разбиение интервала с помощью нечетких множеств с треугольной функцией принадлежности

Метод нечеткого вывода Такаги-Сугено

Такаги и Сугено в работе [5] предложили алгоритм нечёткого вывода, в котором применяемые правила $R^{(k)}$, $k=1,2,3,\dots,N$, нечетки только в части ЕСЛИ, тогда как ТО имеют функциональные зависимости:

$R^{(1)}$: ЕСЛИ (x_1 это A_1^1 И x_2 это A_2^1 ... И x_n это A_n^1), ТО $y_1 = f^{(1)}(x_1, x_2, \dots, x_n), \dots$

$R^{(N)}$: ЕСЛИ (x_1 это A_1^N И x_2 это A_2^N ... И x_n это A_n^N), ТО $y_1 = f^{(N)}(x_1, x_2, \dots, x_n), \dots$

Допустим, что на вход модуля нечеткого управления подается сигнал $\bar{x} = (\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n)$, который может интерпретироваться как вектор состояния объекта. Определим выходной сигнал \bar{y} этого модуля. Для правила $R^{(1)}$ рассчитываем $\mu_{A_1^1}(\bar{x}_1), \mu_{A_2^1}(\bar{x}_2), \dots, \mu_{A_n^1}(\bar{x}_n)$

$$w^1 = \begin{cases} \min\{\mu_{A_1^1}(\bar{x}_1), \mu_{A_2^1}(\bar{x}_2), \dots, \mu_{A_n^1}(\bar{x}_n)\} \\ \mu_{A_1^1}(\bar{x}_1), \mu_{A_2^1}(\bar{x}_2), \dots, \mu_{A_n^1}(\bar{x}_n) \end{cases}$$

На следующем шаге рассчитываем $\bar{y}_1 = f^{(1)}(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n)$.

Продолжаем расчеты для остальных правил $R^{(k)}$, $k = 2, 3, \dots, N$. Для правила $R^{(N)}$ получаем рассчитываем $\mu_{A_1^N}(\bar{x}_1), \mu_{A_2^N}(\bar{x}_2), \dots, \mu_{A_n^N}(\bar{x}_n)$

$$w^N = \begin{cases} \min\{\mu_{A_1^N}(\bar{x}_1), \mu_{A_2^N}(\bar{x}_2), \dots, \mu_{A_n^N}(\bar{x}_n)\} \\ \mu_{A_1^N}(\bar{x}_1), \mu_{A_2^N}(\bar{x}_2), \dots, \mu_{A_n^N}(\bar{x}_n) \end{cases}$$

$$\bar{y}_N = f^{(N)}(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n).$$

Выходной сигнал модуля нечеткого управления Такаги-Сугено представляет собой нормализованную взвешенную сумму отдельных выходов $\bar{y}_1, \dots, \bar{y}_N$ т.е.

$$\bar{y} = \frac{\sum_{k=1}^N w^k \bar{y}_k}{\sum_{k=1}^N w^k}.$$

В линейном случае базу правил модуля можно записать в виде

$R^{(k)}$: ЕСЛИ (x_1 это A_1^k И x_2 это A_2^k ... И x_n это A_n^k),

ТО $y_k = c_0^{(k)} + c_1^{(k)} x_1 + \dots + c_n^{(k)} x_n$, для $k = 1, 2, 3, \dots, N$.

Алгоритм классификации при помощи системы нечёткой логики.

На рисунке 3 приведена блок схема алгоритма классификации записей набора данных NSL-KDD при помощи системы нечёткой логики.

Алгоритм классификации сетевого трафика включает в себя следующие этапы:

1. Из базы данных трафика берётся одна запись, из которой берутся значения следующих ранее отобранных признаков: Src_bytes, Dst_bytes, Serror_rate, Dst_host_serror_rate, Dst_host_srv_serror_rate и метка записи эксперта (“normal” или “anomaly”).

2. Фаззификация – процесс преобразования значений признаков сетевого трафика в лингвистические переменные.

3. Значения 5 лингвистических переменных отправляются в систему нечёткого вывода Такаги-Сугено, внутри которого происходит поиск правила, подходящего значениям входных лингвистических переменных и ранее описанные вычисления.

4. Вывод результатов происходит следующим образом:

Первый столбец — это номер записи в базе данных.

Со второго по шестой столбец включительно – значения наших 5 входных признаков сетевого трафика соответственно: Src_bytes, Dst_bytes, Serror_rate, Dst_host_serror_rate, Dst_host_srv_serror_rate.

Седьмой столбец — это метка в базе данных, которую установили эксперты.

Восьмой столбец — это результат, полученный из разработанной ранее нечёткой системы после обработки значений входных признаков.

5. Проверка наличия в базе данных не классифицированных записей. Если есть, то перейти к пункту 1, иначе перейти к пункту 6.

6. После прохода по всей базе данных рассчитывается количество правильно классифицированных записей (табл. 2).

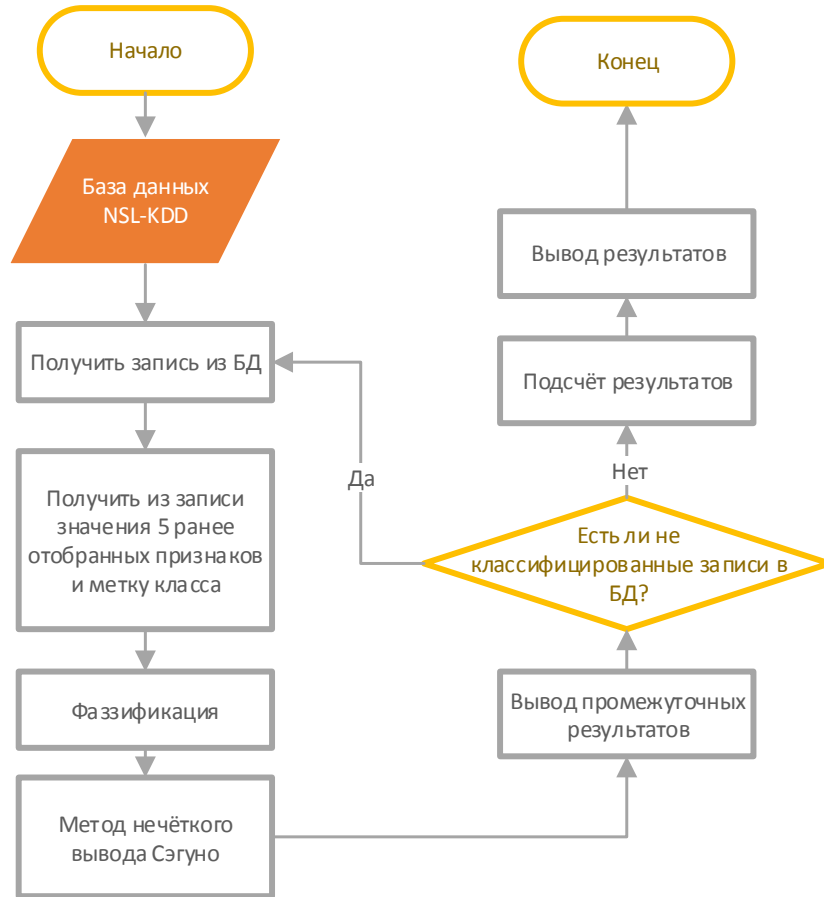


Рис. 3. Алгоритм классификации сетевого трафика при помощи нечёткой логики

Результаты эксперимента

После прохождения записей через систему нечёткой логики наши записи разделены на 4 класса:

- Нормальные данные (normal);
- Скорее всего нормальные данные (mb_normal);
- Скорее всего данные атаки (mb_anomaly);
- Атака (anomaly).

В таблице 2 приведена матрица ошибок бинарной классификации.

Таблица 2

Результаты бинарной классификации.

| | Нормальные данные | Данные атаки | Скорее всего нормальные данные | Скорее всего данные атаки |
|------------------------------|-------------------|--------------|--------------------------------|---------------------------|
| Отнесено к нормальным данным | 71932 | 16571 | 286 | 23 |
| Отнесено к атаке | 5122 | 54892 | 1825 | 4 |

Поскольку в данной работе рассматривается бинарная классификация, необходимо обобщить данные, например способами, представленными в табл. 3.

Обобщение результатов классификации.

| Всего записей | Правильно классифицированных записей | % | Вид обобщения результатов | Описание |
|---------------|--------------------------------------|----------------|--------------------------------|--|
| 148517 | 126824 | 85,39359 | Абсолютный | Учитывается только абсолютное совпадение с меткой в БД |
| | 127115 | 85,58953 | 50 на 50 | mb_normal относится к нормальным данным, а mb_anomaly к аномалии |
| | 128653 | 86,6251 | Оба значения к аномалии | mb_normal и mb_anomaly относятся к аномалии |
| | 127133 | 85,60165 | Оба значения к аномалии | mb_normal и mb_anomaly относятся к нормальным данным |

Заключение

В работе предложено использовать при разработке СОВ, основанных на нечёткой логике, систему Такаги-Сугено, вместо обычно используемой системы Мамдани. Достоинством системы Такаги-Сугено является то, что для неё не требуется проводить дефuzziфикацию результатов, что ускоряет классификацию и снижает количество необходимых вычислительных ресурсов системы. Разработанная СОВ достигла максимальной точности классификации в 86,6%, при использовании только 5 признаков сетевого трафика, что позволило значительно упростить систему обработки, ускорить процесс классификации, а также снизить необходимые вычислительные ресурсы системы. Однако это повлекло за собой некоторое снижение качества классификации примерно на 5% относительно результатов работы [5].

Литература

1. J. Luo, and S. M. Bridges, "Mining fuzzy association rules and fuzzy frequency episodes for intrusion detection", International Journal of Intelligent Systems, Vol. 15, No. 8, pp. 687-704, 2000.
2. Han J., Kamber M., Data Mining Concepts and Techniques, 3rd edition, 2012.
3. Shrivastava A. K., Singhai S. K., Hota H. S. An Efficient Decision Tree Model for Classification of Attacks with Feature Selection. International Journal of Computer Applications, Volume 84 – No 14, 2013
4. Рутковская Д., Пилиньский М., Рутковский Л. Нейронные сети, генетические алгоритмы и нечеткие системы: Пер. с польск. И. Д. Рудинского. М.: Горячая линия – Телеком, 2006. 452 с.
5. R. Shanmugavadivu, Network intrusion detection system using fuzzy logic, Indian Journal of Computer Science and Engineering (IJCSE), Vol. 2 No. 1 pp: 101-111, 2011.
6. Шелухин О.И., Симонян А.Г., Ванюшина А.В. Влияние структуры обучающей выборки на эффективность классификации приложений трафика методами машинного обучения // Т-Comm: Телекоммуникации и транспорт. 2017. Т. 11. № 2. С. 25-31.
7. Шелухин О.И., Смычэк М.А., Симонян А.Г. Фильтрация нежелательных приложений интернет-ресурсов в целях информационной безопасности // Научно-технические исследования в космических исследованиях Земли. 2018. Т. 10. № 2. С. 87-98.

BINARY CLASSIFICATION OF COMPUTER ATTACKS TO INFORMATION RESOURCES USING FUZZY LOGIC

Sergey P. Gruzdev

master student of the group M091801(71) MTUCI

House.HD@yandex.ru

Oleg I. Sheluhin

MTUCI, Doctor of Technical Sciences, Professor of Information Security department

sheluhin@mail.ru

Keywords: *KDD, NSL-KDD, intrusion detection system, fuzzy logic, Takagi-Sugeno fuzzy inference system, binary classification, normal data, attack, signs of network traffic.*

The aim of the work was to find the minimum number of signs of network traffic to create an optimal fuzzy system, and with it's help the binary classification of network traffic will be carried out. The experiments of the created intrusion detection system with the use of fuzzy logic were performed using the NSL-KDD computer attack data set. Experimental results showed that the created binary classification system with the use of fuzzy logic provides the accuracy of the correct classification of computer attacks at the level of 86.6%.

W-CODE ПЛАТФОРМЫ, КАК ИНСТРУМЕНТ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Летунов Алексей Сергеевич
Студент группы М091801 (72) МТУСИ
alexletunov@mail.ru

Докучаев Владимир Анатольевич
МТУСИ, д.т.н., профессор, заведующий кафедрой МСuУС
v.dok@tlsf.ru

Ключевые слова: low-code платформа, low-code разработка, кодирование, 4GL, IoT, AI, no-code платформа, AD&D, бизнес-приложение, RAD.

Проведен анализ развития разработки приложений при помощи использования low-code платформ. Проведено сравнение общих положений создания приложений с помощью кодирования и использования low-code платформ. Рассмотрены особенности и достоинства данных платформ. Проанализирован рынок low-code платформ и спрогнозировано их будущее развитие.

Для понятия «*low-code* платформа» не существует научного определения, несмотря на то что данная технология не является новой. В соответствии с источником [1] понятие «*low-code* платформа» определяется как платформа, «обеспечивающая быструю разработку приложений с минимальным кодированием, быстрой настройкой и развертыванием для систем взаимодействия». В соответствии с [2] определение было дополнено следующим: «*low-code* платформа – это продукты и (или) облачные сервисы для разработки приложений, в которых вместо программирования используются визуальные декларативные методы и которые доступны клиентам по низким ценам, а также не требуют больших затрат по времени на обучение. При этом стоимость приложения растет пропорционально коммерческой ценности платформы». Также можно утверждать, что данные платформы являются следующей ступенью развития после *CMS*, за тем лишь важным исключением, что платформы позволяют создавать полноценные веб-приложения, а не просто связанные между собой *web*-страницы [3]. Первоначально *low-code* платформы возникли как инструменты автоматического генерирования кода, но в настоящее превратились в платформы разработки приложений корпоративного уровня. Инструментарий *low-code* платформ позволяет разрабатывать крупномасштабные приложения.

Low-code платформы опираются на декларативный инструментарий разработки вместо использования всеми привычных языков программирования. Инструменты визуальной и декларативной разработки были доступны в различных формах на протяжении десятилетий, и поставщики применяли эти методы для разработки приложений узкой и широкой направленности. В число инструментов могут входить: предметно-ориентированные языки, графический редактор пользовательского интерфейса, понятие *UX* (пользовательского опыта), построение диаграмм и визуальное моделирование данных. Однако создание новых приложений с помощью кодирования продолжает оставаться трудоемким процессом. С помощью вышеперечисленных инструментов команда разработчиков сможет разрабатывать информационные системы значительно быстрее, чем с помощью языков программирования. Они также открывают возможность быстро проверить бизнес идеи в рабочем приложении. Визуальная настройка новых приложений с использованием этих платформ позволила ускорить разработку их программного обеспечения в 5-10 раз по сравнению со стандартной разработкой программного обеспечения, как показано на рис. 1, и позволила бизнесу предоставлять обратную связь в реальном времени о результирующей функциональности приложений [2, 4]. Также можно утверждать, что сегодня на рынках «быстрое создание и запуск новых цифровых продуктов и услуг, обеспечивающих удобное, простое и интересное обслуживание клиентов, является ключевым рычагом» [2, 4]. Сравнение различных программных проектов [4], созданных с помощью кодирования и использования *low-code* платформ, представлено в табл. 1.

Таблица 1

Сравнение различных программных проектов, созданных с помощью кодирования и использования *low-code* платформ

| Предприятие | Результат | Кодирование | Low-code платформа |
|---|---------------------------------------|---------------------|--|
| Правительство США (Закон о доступной медицинской помощи) | Модуль соответствия документа | 100 человеко-месяца | 5 человеко-месяца |
| Британская страховая компания | Агентский портал | Неизвестно* | 10 дней для создания минимально жизнеспособного продукта |
| Колл-центр | Клиентское приложение | 4 месяца | 3 недели |
| Испанская страховая компания | Веб-канал и система администрирования | 2,7 лет (примерно) | 13 недель |

*В течение многих лет проект находился в списке отставания, и его запуск стал вовсе маловероятен

Сегодня, когда на рынке программного обеспечения существует неограниченный выбор продуктов, создание нового пользовательского приложения для клиента стало довольно сложным. При гибкой разработке программного обеспечения, часто возникает ситуация, когда видение конечного продукта может изменяться после того, как разработка продукта закончена и его функции можно увидеть в действии. Приложение может работать не так, как представлялось, либо из-за ограничений в технологии, либо из-за деталей общего рабочего процесса. Поэтому зачастую лучше быстро создать минимально жизнеспособные продукты для проверки идей и требований клиентов, прежде чем тратить ресурсы на функции и функциональные возможности, которые клиенты могут даже не оценить. *Low-code* платформы позволяют компаниям быстро воплощать идеи в недорогие рабочие прототипы, которые они могут развернуть и протестировать на рынке. Прототипы могут быть построены путем кодирования, но эти прототипы, возможно, придется перестраивать по мере увеличения объема и разнообразия пользователей. Благодаря *low-code* разработке сама платформа обеспечивает быстрое развертывание, масштабирование и настройку производительности приложений.

Особенности и преимущества *low-code* платформ

Инструменты декларативной разработки программного обеспечения существуют практически со времен языков *4GL* (четвёртое поколение языков программирования). *Low-code* платформы характеризуются следующими особенностями: они более открыты и легче интегрируются с другими системами. До нынешнего поколения эти инструменты и платформы были полностью проприетарными и, следовательно, были изолированными и трудными для подключения в более широкие портфели приложений. *Low-code* платформы также используют многие из этих элементов, но они также предоставляют такие функции, как *API*-интерфейсы платформы (интерфейс прикладного программирования). Большинство из них совместимы со средами разработки с открытым исходным кодом, в частности с *AngularJS* и *React*. Можно сказать, что текущее поколение *low-code* платформ более соответствует термину «платформы», потому что они не являются простыми инструментами, а они являются законченными платформами. Такие функции, как развертывание приложений и управление жизненным циклом приложений, отличают текущие продукты от предыдущих [4].

Разработка приложений с использованием *low-code* платформ не требует больших временных затрат. Они позволяют команде разработчиков добиться успеха с менее опытными разработчиками, сводя к минимуму кодирование и, следовательно, ускоряя разработку. Они также могут

устранить барьеры для участия клиентов в проектах. Легко получить быструю обратную связь от клиента, а также от других сотрудников и партнеров, благодаря так называемому подходу «тестируй и учись», когда приложение может быть показано клиенту на ранней стадии разработки. Это устраняет догадки о требованиях, потому что требования могут быть проверены в действии на очень ранней стадии разработки. Устойчивая цифровая трансформация требует адаптивного многократно повторяемого подхода к разработке приложений, что значительно упрощает саму разработку и обслуживание программного обеспечения. Это позволяет легко масштабировать программное обеспечение по всему предприятию и адаптироваться к изменениям бизнеса. Большинство, если не все *low-code* платформы, предоставляют адаптивный дизайн, мобильную функциональность и автономную поддержку по умолчанию. Все находится в унифицированной и централизованной среде, от аутентификации пользователя и *UI* (пользовательского интерфейса) до управления хранилищем данных и доступа на основе ролей [1]. Можно выделить пять наиболее важных особенностей хорошей *low-code* платформы [3]: легкая визуальная конфигурация, готовые варианты интеграции, мобильная совместимость, масштабируемость и поддержка полного жизненного цикла, которые, согласно отчетам [5], являются частями наиболее успешных платформ.

Разработка с использованием *low-code* платформ не идет по традиционному пути «очередного повторения запрограммированного черного ящика», потому что они позволяют вести сотрудничество и обмениваться идеями практически в реальном времени. Приложения – это больше, чем просто код, они управляются самой логикой, поскольку бизнес-правила «внедряются» в приложение на уровне ядра. Кроме того, производительность труда разработчиков может увеличиться, поскольку они могут тратить меньше времени на написание кода и больше времени на создание объектов, представляющих функциональность многих строк кода. Модификации и другие задачи обслуживания могут выполняться в одной части системы – изменения распространяются на всю систему. Часть нововведений были внедрены в *low-code* платформы, чтобы ослабить ограничения на модификацию и обслуживание, затрагивающие изменения схемы модели данных. Теперь изменения в модель данных могут быть внесены, не беспокоя пользователей информационной системы. Одной из проблем, но также и преимуществ *low-code* платформ, является безопасность приложений. Пользовательские права могут быть установлены администратором системы без каких-либо необходимых знаний языка программирования. Оптимизированный и свободный от кода подход к этой функции позволяет пользователям не видеть данные, которые они не должны видеть.

Было несколько мифов о *low-code* платформах и *low-code* разработке в целом. Одним из них было то, что *low-code* платформы предназначены только для начинающих малоопытных разработчиков, а не для профессиональных. Согласно исследованию [4], разработчики используют *low-code* платформы для создания инструментов, предназначенных для малоопытных специалистов, которые в свою очередь занимаются самой разработкой приложений для клиентов. Другой миф состоит в том, что данные платформы направлены на устранение необходимости в любом программировании. Несомненно, есть вероятность создать некоторые относительно простые приложения без необходимости писать какой-либо код, но, все же, разработка более сложного приложения не обойдется без применения кодирования. К кодированию относят такие функции, как интеграции и пользовательские алгоритмы. В настоящее время на рынке программных обеспечений начинают появляться *no-code* платформы, которые могут рассматриваться как эволюция *low-code* платформ или, скорее, как часть платформ, когда уже изготовленные и кастомизированные компоненты могут использоваться при разработке, например, инструменты *drag-n-drop* («тяни-бросай»). В целом *low-code* платформы следует рассматривать как новые возможности, а не то, что могло бы поставить под угрозу работу профессиональных разработчиков. В опросе о *RAD* (быстрой разработке приложений) 74% ИТ-специалистов рассматривали отсутствие кода или низкое его содержание как ключ при оценке платформ *RAD* [5]. В том же опросе 56% ожидали, что количество приложений без кода и с низким его содержанием увеличится. Однако, стоит отметить, что в конечном итоге данные платформы не являются решением всех проблем программирования, и всегда найдется место для более традиционных фреймворков. Следующие новые большие инновации, ожидаемые для интеграции в платформы, – это датчики и исполнительные механизмы *IoT* (интернет вещей), а также *AI* (искусственный интеллект) [2, 4].

Недостатки *low-code* платформ

Low-code платформы не идеальны, несмотря на положительные отзывы. Можно утверждать, что современное поколение *low-code* платформ является лишь альтернативой *4GL*, а также, что компании не считают создание программного обеспечения дорогим и сложным предприятием. *Low-code* платформы используют подход к разработке приложения, при котором приложение постоянно модифицируется, подвергается исправлению. Но такой подход может увеличить конечную стоимость продукта, что дает преимущество процессу создания приложения при помощи языков программирования. Разработчики вправе использовать любые удобные инструменты, но проект разработки программного обеспечения не будет иметь успех, если разработчики не будут достаточно опытными. За каждым успешным проектом по разработке программного обеспечения должен быть разработчик, знающий внутреннюю архитектуру *low-code* платформы. Должен быть тот, кто имеет представление об основах разработки программного обеспечения.

Одна из самых больших проблем многих *low-code* платформ заключается в том, что они на самом деле не ограничивают код; они просто скрывают его от конечного пользователя. Создание визуальных инструментов должно быть преобразовано, скомпилировано и выполнено так же, как и обычное программное обеспечение, написанное путем кодирования. Можно считать, что *low-code* платформы — это инструменты, которые могут решать некоторые бизнес-задачи, но они никак не устраняют необходимость в использовании кодирования из-за всех ограничений. *Low-code* платформы всегда предлагают возможность вернуться к кодированию, когда необходимо выполнить более сложную логику. Этот факт показывает недостаток данных платформ, поскольку компании по-прежнему не могут обойтись без разработчиков, которые понимают основы разработки программного обеспечения. Несмотря на то, что *low-code* платформы могут и не избавлять от потребности в опытных разработчиках, но, все же, делают код узким местом, влияющим на скорость разработки.

Рынок *low-code* платформ

Изначально *low-code* платформы были сегментированы следующим образом [4]: платформы общего назначения, платформы приложений процессов, платформы приложений баз данных, платформы обработки запросов и платформы приложений для мобильных устройств. Год спустя ситуация изменилась. Большинство платформ, если не все платформы на рынке, стали платформами общего назначения [2].

Таблица 2

Сравнение сегментов *low-code* платформ

| | <i>AD&D</i> | Бизнес-приложения |
|---|---|--|
| Цель | Ускорить разработку программного обеспечения, сделав профессиональных разработчиков более продуктивными | Ускорить разработку программного обеспечения, с помощью управления разработкой бизнес-экспертами |
| Целевой покупатель | Лидеры <i>AD&D</i> | Бизнес-лидеры |
| Предназначен для | Разработчиков, которым важен контроль и гибкость | Разработчиков, которым важна простота расширения возможностей |
| Количество use cases | Множество | Узкий диапазон |
| Масштаб целевого приложения | От средних до больших | Средний |
| Обеспечивает инструментами | Бизнес-эксперты | <i>AD&D</i> и ИТ-специалисты |
| Предпочитаемый способ разработки | <i>Low-code</i> | <i>No-code</i> |
| Поставщики | <i>Appian, Mendix, OutSystems</i> и <i>Salesforce</i> | <i>Caspio, Kintone, MatsSoft</i> и <i>Nintex</i> |

Были признаны два новых сегмента, определенных целевым покупателем, а также стремление поставщика влиять на то, кто и как предоставляет приложения: платформы для *AD&D* (разработка и поставка приложений) и платформы для разработчиков бизнес-приложений. Первый сегмент является продуктивной альтернативой программированию или, другими словами, стратегической альтернативой *Java* и *.NET*. Второй сегмент предназначен для разработчиков, которые могут создавать приложения с помощью простых инструментов. Сравнение описанных выше двух сегментов представлено в табл. 2.

В настоящее время, рынок *low-code* платформ растет, поскольку крупные корпоративные компании начали ценить данные платформы как инструменты для решения более сложных проблем [2]. Доход отслеживаемых 67 поставщиков достиг 2,6 млрд долларов в 2016 году и 3,8 млрд долларов в 2017 году [2]. Рост будет происходить до 2020 года, когда этот показатель составит 55% (рис. 1). Отметим, что рынок все еще фрагментирован, так как большинство поставщиков имеют годовой доход менее 100 миллионов долларов, причем *Salesforce* является единственным поставщиком с доходом в 1 миллиард долларов. Возможно, что тенденция новых платформ нацеливается на использование узких *use cases* (сценарии использования). Традиционные ИТ-предприятия, такие как *Dell*, *Microsoft*, *Oracle*, *IBM* и *SAP*, также присоединятся к рынку, скорее всего, покупая более мелких поставщиков вместо создания новых платформ, поскольку создание *low-code* платформы с нуля может превзойти возможности даже самых крупных поставщиков [2].



Рис. 1. Предполагаемый рост рынка *low-code* платформ

Заключение

Low-code платформы предоставляют большие возможности для разработки программного обеспечения. Несмотря на все их возможности, данные платформы не предназначены и не преследуют цель заменить разработку, использующую кодирования с помощью языков программирования. Также было отмечено, что *low-code* платформы пользуются спросом на рынке, что говорит о том, что крупные корпоративные компании начали ценить данные платформы как инструменты для решения сложных проблем.

Литература

1. Richardson, C., Rymer, J. New Development Platforms Emerge For Customer-Facing Applications, 2014.
2. Rymer, J. A Fork In The Road For Low-Code Development Platforms, 2017.
3. Nepal, M. Launchpads to Build Powerful Apps Easily, 2018.
4. Richardson, C., Rymer, J. Vendor Landscape: The Fractured, Fertile Terrain Of Low-Code Application Platforms, 2016.
5. PMG. IT and the Rise of the RAD Enterprise, 2017.

LOW-CODE DEVELOPMENT PLATFORM AS A TOOL FOR SOFTWARE DEVELOPMENT

Alexey S. Letunov

Graduate student M091801 (72), MTUCI

alexletunov@mail.ru

Vladimir A. Dokuchaev

MTUCI, professor, Head of Department MCSaS

v.dok@tlsf.ru

Key words: *low-code development platform, low-code development, hand-coding, 4GL, IoT, AI, no-code platform, AD&D, business application, RAD.*

The development of applications using low-code development platform is analyzed. Comparison of the general provisions for creating applications using low-code platforms. Features and advantages of these platforms are considered. The market with low-code development platforms is analyzed and their future development is predicted.

СРАВНИТЕЛЬНЫЙ АНАЛИЗ СОВРЕМЕННЫХ МЕТОДОВ СЖАТИЯ ИЗОБРАЖЕНИЙ БЕЗ ПОТЕРЬ

Нуралин Дмитрий Германович
студент группы М091801(72) МТУСИ
nurmity@yandex.ru

Шевелев Сергей Владимирович
МТУСИ, к.т.н., доцент кафедры МСuУС
shevelev-s@yandex.ru

Ключевые слова: Сжатие без потерь, кодирование длин серий, алгоритм Шеннона - Фано, алгоритм Хаффмана, алгоритм Лемпеля - Зива - Велча, арифметическое кодирование.

Повседневная деятельность человека приводит к генерации огромного количества данных, что создает большие проблемы для их хранения и передачи. Следовательно, существует растущий спрос на дополнительные исследования в области сжатия данных и теории связи для решения этих проблем. В статье представлена краткая характеристика наиболее распространенных современных методов сжатия изображений без потерь. Рассмотрены критерии оценки эффективности методов сжатия. На основе данных критериев, проводится сравнительный анализ алгоритмов. По итогу анализа сделаны выводы об оптимальном методе.

В настоящее время использование компьютера в различных видах деятельности повсеместно растет. Постоянный рост объемов данных, в частности изображений и видео, является наиболее сложной проблемой из-за ограниченной пропускной способности и ёмкости хранения; что отнимает много времени и средств [1]. К примеру, простая видеокамера как правило записывает 24 кадра в секунду, хотя, последние стандарты видео позволяют использовать гораздо большее количество кадров в секунду. Видео - это серия кадров (изображений), пропускаемых за секунду, в которой представление цветных изображений состоит из трёх каналов: красного, зеленого и синего. Предположим, вы хотите отправить или сохранить трехчасовой файл цветного фильма размером $1200 * 1200$, где каждую секунду передается 24 кадра, каждый пиксель которого закодирован 8 битами. Требуется приблизительно $(1200 * 1200 * 3 * 8 * 24 * 10800)$ бит = 8542968,75 Мбит = 1042,84 Гбайт, что является довольно сложной задачей для хранения на компьютере или отправки по вычислительной сети. Кроме того, скорость и задержка являются двумя основными ограничителями при передаче данных. Если видеофайл отправляется со скоростью 100 Мбит/с, требуется приблизительно $8542968,75 \text{ Мбит} / 100 = 85429,6875 \text{ с} = 23,73 \text{ ч}$. Именно поэтому сжатие изображений так востребовано при работе с мультимедиа.

Сжатие – это получение более компактного выходного потока информационных единиц после преобразования некоторого входного потока. Оно основано на уменьшении или полном устранении избыточности, содержащейся в исходных данных изображения [2].

Избыточность бывает нескольких типов:

1. Избыточность кодирования: проявляется в использовании неоптимального кодовых слов при кодировании пикселей.
2. Межпиксельная избыточность: соседние пиксели в изображении близки по цвету.
3. Визуальная избыточность: человеческое зрение при анализе изображения оперирует контурами, общим переходом цветов и сравнительно нечувствительно к малым изменениям в изображении.

Таким образом, используя алгоритмы сжатия, избыточные биты удаляются из изображения. Сжатие данных, не обладающих свойством избыточности, невозможно без значительных потерь в качестве изображения.

Существует огромное множество методов сжатия изображений, эффективность которых

принято оценивать при помощи таких характеристик, как средняя длина кода, время кодирования и декодирования, степень сжатия и т.д. Алгоритмы сжатия изображений широко применяются в визуализации медицинских изображений, в военной связи, при проведении телеконференций, при проведении магнитно-резонансной томографии, телевизионном и спутниковом вещании и т.д. В некоторых из этих областей требуется качественная визуальная информация, в других же качество изображений не является обязательным требованием [3].

Сжатие данных подразделяется на два типа: с потерями и без потерь. Сжатие с потерями – класс алгоритмов, который в результате декомпрессии позволяет восстановить исходное изображение с точностью до бита. В свою очередь при сжатии без потерь декомпрессированное изображение отличается от исходного, но степень отличия незначительна с точки зрения их дальнейшего использования. Данный тип сжатия работает только над устранением избыточности кодирования. Рассмотрим наиболее популярные методы сжатия без потерь.

Кодирование длин серий

Кодирование длин серий (Run-length encoding, RLE) – это самый простой из алгоритмов сжатия данных, который вместо использования статистической информации занимается кодированием набора подряд идущих одинаковых пикселей в виде значения пикселей и их количества. Обычно используется в форматах TIFF и PDF. RLE максимально эффективно работает для данных, содержащих длинные серии идентичных символов, что как правило редко встречается в полутонных изображениях, таких как фотографии, это является основной проблемой данного метода. Однако, для иконок и графических рисунков данный метод подходит наилучшим образом.

Алгоритм Шеннона – Фано

Методы энтропийного кодирования предлагаются для решения трудностей RLE алгоритма. Отличительной особенностью этих методов является кодирование пикселей изображения с помощью кодовых слов различной длины. Существует несколько широко применяемых методов энтропийного кодирования: алгоритмы Шеннона – Фано, Хаффмана и арифметическое кодирование. Алгоритм Шеннона - Фано – это метод кодирования без потерь, который берет отсортированные вероятности пикселей в порядке убывания, после чего делит их на два набора, где общая сумма каждого набора почти эквивалентна. Он является одним из модификаций алгоритма статического кодирования Хаффмана. Разница лишь в создании кодового слова. Все остальные процессы эквивалентны алгоритму Хаффмана. Алгоритм Шеннона - Фано иногда генерирует два разных кода для одного и того же пикселя и определяет неоптимальные коды. Это являются двумя основными проблемами данного алгоритма, поэтому с точки зрения перспективы, кодирование Шеннона - Фано является неэффективным методом сжатия данных.

Алгоритм Хаффмана

Дэвид Хаффман проиллюстрировал процедуру кодирования, которая последовательно создает оптимальное дерево, и рассмотрел проблемы, которые существуют в кодировании Шеннона – Фано [4]. Алгоритмы кодирования Хаффмана используют подсчет частоты появления каждого уникального пикселя для расчета вероятности его появления. После чего происходит назначение более коротких кодовых слов для пикселей с более высокой вероятностью и более длинных для пикселей с более низкой вероятностью. Для этой задачи создается бинарное дерево с использованием значений пикселей в виде листьев в соответствии с их вероятностями. Алгоритмы Хаффмана бывает двух видов: статические и адаптивные. Статический алгоритм Хаффмана сначала вычисляет частоты каждого пикселя, а затем генерирует бинарное дерево для процессов компрессии и декомпрессии. Таблица с кодами пикселей должна быть сохранена и передана вместе со сжатым изображением. Адаптивный алгоритм позволяет строить бинарное дерево и вычислять код пикселя по мере чтения следующего символа без предварительного анализа частот. Компрессия и декомпрессия осуществляются динамически без использования кодовой таблицы.

Арифметическое кодирование

Арифметическое кодирование – это процедура сжатия данных без потерь, при которой набор символов представляется с использованием фиксированного числа битов. В отличие от алгоритма Хаффмана, кодовое слово не используется для представления пикселя, так как нет привязки входных символов группам бит полученного кода. Это даёт алгоритму большую гибкость в представлении дробных частот встречаемости символов. В исходном сообщении принимаются во внимание вероятности отдельного пикселя и кумулятивные вероятности набора пикселей. В процессе кодирования кумулятивные вероятности каждого пикселя вычисляются, и создается первоначальный диапазон. При чтении каждого пикселя выбирается соответствующий поддиапазон в пределах диапазона кумулятивной вероятности. Затем выбранный диапазон делится на части в соответствии с вероятностями входного алфавита. Потом считывается следующий пиксель и выбирается соответствующий поддиапазон. Таким образом, пиксели читаются многократно, пока сообщение не закончится. Наконец, из последнего поддиапазона должно быть взято число в качестве результата процесса кодирования. Это будет доля в этом поддиапазоне. Чтобы декодировать сжатое таким образом изображение, необходимо количество символов исходного сообщения и распределение вероятностей.

Алгоритм Лемпеля - Зива - Велча

Алгоритм Лемпеля – Зива – Велча (Lempel – Ziv – Welch, LZW) – алгоритм, изобретенный Авраамом Лемпелем, Яаковом Зивом и Терри Велчем и обычно используемый для сжатия текста без потерь. Этот метод был опубликован в 1984 году как обновленная версия LZ78 [5], он прост в реализации и широко применяется для сжатия файлов в Unix системах. LZW вместо статистической модели использует словарь. Словарь – это набор возможных фраз, который хранится в виде таблицы и использует индексы записей для представления больших и повторяющихся фраз. В процессе сжатия эти значения индекса используются вместо повторяющихся фраз. Метод считывает последовательность пикселей, группирует их в строки, а затем проверяет существуют ли в словаре такая строка. Если существует, считывается следующий пиксель, а если не существует, строка заносится в словарь, преобразуется в код, и поиск начинается снова. Словарь создается динамически в процессе сжатия и нет необходимости передавать его с закодированным сообщением, так как в процессе декодирования, тот же словарь создается динамически. Поиск в словаре является серьезной проблемой в технике сжатия LZW, потому что это сложный и трудоемкий процесс. Кроме того, изображение, которое содержит небольшое количество повторяющихся данных вообще, не может быть уменьшено.

Результаты анализа

Для анализа был использован набор из 10 медицинских изображений, соответствующих стандарту DICOM, представленных на рисунке 1. В качестве критериев оценки эффективности алгоритмов были использованы время кодирования, время декодирования, средняя длина кода и степень сжатия. Средняя длина кода определяет сколько в среднем необходимо битов для хранения одного пикселя, а степень сжатия представляет соотношение числа бит исходного и сжатого изображений. Чем меньше время кодирования и декодирования, средняя длина кода и чем больше степень сжатия, тем быстрее работает алгоритм и меньше памяти использует.

Таблица 1 показывает, что арифметическое кодирование – это самый медленный алгоритм (4,6713 мс), а LZW – самый быстрый (0,1088 мс), тогда как Шеннона - Фано, Хаффмана и RLE для кодирования изображений по отдельности требуется 0,649, 0,2956 и 0,1605 миллисекунды. Арифметическое кодирование использует на 97%, 86%, 94% и 98% больше времени, чем кодирование RLE, Шеннона - Фано, Хаффмана и LZW, соответственно. Однако кодирование Хаффмана и Шеннона - Фано использует в среднем гораздо меньше времени при декодировании, тогда как арифметическое кодирование использует больше времени, что показано в таблице 2. С другой стороны, LZW использует больше времени декодирования, чем алгоритмы Шеннона - Фано и Хаффмана, но меньше, чем арифметическое кодирование и RLE.

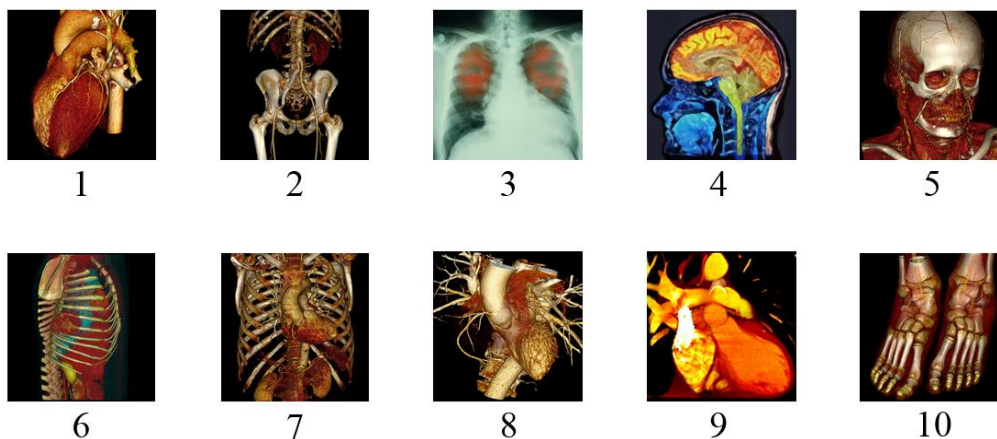


Рис. 1. Набор медицинских изображений

Таблица 1

Сравнение времени кодирования

| Изображения | RLE | Алгоритм Шеннона - Фано | Алгоритм Хаффмана | LZW | Арифметическое кодирование |
|-------------|--------|-------------------------|-------------------|--------|----------------------------|
| 1 | 0.1670 | 0.6735 | 0.2150 | 0.1060 | 3.7628 |
| 2 | 0.1870 | 0.7304 | 0.2534 | 0.1060 | 3.3215 |
| 3 | 0.1370 | 0.5079 | 0.1816 | 0.1060 | 7.3937 |
| 4 | 0.1120 | 0.5085 | 0.2059 | 0.1060 | 3.2170 |
| 5 | 0.1860 | 0.6756 | 0.2139 | 0.1180 | 5.9698 |
| 6 | 0.1320 | 0.5369 | 0.1818 | 0.1060 | 3.6537 |
| 7 | 0.1650 | 0.7816 | 0.2183 | 0.1180 | 4.6419 |
| 8 | 0.1860 | 0.6002 | 0.2252 | 0.1070 | 4.4352 |
| 9 | 0.1610 | 0.8058 | 1.0607 | 0.1080 | 5.5250 |
| 10 | 0.1720 | 0.6697 | 0.2004 | 0.1070 | 4.7927 |
| Среднее | 0.1605 | 0.6490 | 0.2956 | 0.1088 | 4.6713 |

Таблица 2

Сравнение времени декодирования

| Изображения | RLE | Алгоритм Шеннона - Фано | Алгоритм Хаффмана | LZW | Арифметическое кодирование |
|-------------|--------|-------------------------|-------------------|--------|----------------------------|
| 1 | 0.059 | 0.0083 | 0.0073 | 0.107 | 4.5913 |
| 2 | 0.079 | 0.0078 | 0.0071 | 0.025 | 4.1705 |
| 3 | 0.060 | 0.0047 | 0.0052 | 0.030 | 5.6244 |
| 4 | 0.039 | 0.0032 | 0.0033 | 0.025 | 4.2222 |
| 5 | 0.065 | 0.0045 | 0.0117 | 0.033 | 9.3637 |
| 6 | 0.057 | 0.0078 | 0.0039 | 0.029 | 4.4086 |
| 7 | 0.065 | 0.0095 | 0.0083 | 0.036 | 5.7221 |
| 8 | 0.069 | 0.0058 | 0.0092 | 0.032 | 5.4117 |
| 9 | 0.054 | 0.0093 | 0.0077 | 0.035 | 7.6485 |
| 10 | 0.058 | 0.0042 | 0.0073 | 0.032 | 5.7914 |
| Среднее | 0.0605 | 0.0065 | 0.0071 | 0.0384 | 5.6954 |

Таблицы 3 и 4 показывают среднюю длину кода и степень сжатия соответственно. RLE использует в среднем 11.3044 бит на пиксель, что на 31.2866% больше используемой в исходных изображениях памяти, необходимой для хранения одного пикселя, и по этой причине, данный метод не используется напрямую при сжатии изображений. С другой стороны, LZW использует наименьшее количество бит (6,2449) на пиксель, но проблема LZW заключается в том, что сжатое при помощи него изображение иногда использует больше памяти, чем оригинал, что произошло для изображения 4, показанного в таблице 3. Арифметическое кодирование использует второе наименьшее количество бит на пиксель в среднем. Таким образом, арифметическое кодирование является наилучшим методом кодирования, поскольку оно обеспечивает лучший коэффициент сжатия, чем другие методы без LZW, показанные в таблице 4.

Таблица 3

Сравнение средних длин кода закодированного пикселя

| Изображения | RLE | Алгоритм Шеннона - Фано | Алгоритм Хаффмана | LZW | Арифметическое кодирование |
|-------------|---------|-------------------------|-------------------|--------|----------------------------|
| 1 | 8.8869 | 5.905 | 5.3491 | 5.0305 | 5.3196 |
| 2 | 7.9405 | 5.430 | 4.6826 | 4.3299 | 4.6721 |
| 3 | 12.1298 | 7.858 | 7.4269 | 7.1653 | 7.3973 |
| 4 | 14.3269 | 7.832 | 7.3848 | 8.0182 | 7.3487 |
| 5 | 10.7583 | 6.725 | 6.3174 | 6.0834 | 6.2889 |
| 6 | 13.8656 | 7.611 | 7.1844 | 7.3612 | 7.1487 |
| 7 | 12.1939 | 7.039 | 6.5289 | 6.4581 | 6.4998 |
| 8 | 11.0767 | 6.556 | 6.1967 | 6.0462 | 6.1743 |
| 9 | 11.4107 | 7.271 | 6.7361 | 6.2314 | 6.6091 |
| 10 | 10.4550 | 6.502 | 6.0550 | 5.7252 | 6.0128 |
| Среднее | 11.3044 | 6.8729 | 6.3862 | 6.2449 | 6.3471 |

Таблица 4

Сравнение коэффициентов сжатия

| Изображения | RLE | Алгоритм Шеннона - Фано | Алгоритм Хаффмана | LZW | Арифметическое кодирование |
|-------------|--------|-------------------------|-------------------|--------|----------------------------|
| 1 | 0.9003 | 1.3552 | 1.4957 | 1.5904 | 1.5040 |
| 2 | 1.0076 | 1.4738 | 1.7086 | 1.8478 | 1.7124 |
| 3 | 0.6596 | 1.0182 | 1.0773 | 1.1166 | 1.0816 |
| 4 | 0.5585 | 1.0217 | 1.0834 | 0.9978 | 1.0887 |
| 5 | 0.7437 | 1.1899 | 1.2665 | 1.3152 | 1.2722 |
| 6 | 0.5769 | 1.0508 | 1.1134 | 1.0867 | 1.1190 |
| 7 | 0.6560 | 1.1363 | 1.2252 | 1.2386 | 1.2307 |
| 8 | 0.7221 | 1.2200 | 1.2909 | 1.3230 | 1.2956 |
| 9 | 0.701 | 1.1001 | 1.1875 | 1.2837 | 1.2103 |
| 10 | 0.7651 | 1.2300 | 1.3211 | 1.3972 | 1.3304 |
| Среднее | 0.7291 | 1.1796 | 1.2770 | 1.3197 | 1.2845 |

В результате, можно сказать, что арифметическое кодирование является лучшим алгоритмом, когда требуется высокая степень сжатия; однако для приложений реального времени данный

метод не подойдёт, поскольку на этапах кодирования и декодирования ему требуется больше времени, чем приложение может себе позволить. Поиск в словаре является большой проблемой для LZW кодирования и обеспечивает худшие результаты для сжатия изображений. Кодирование Шеннона - Фано иногда не даёт оптимального кода и предоставляет два разных кода для одного и того же элемента, что является причиной его устаревания в настоящее время. Кодирование длин серий не подходит для непосредственного сжатия изображения, так как в среднем сжатое изображение занимает больше места чем оригинал. Таким образом, можно утверждать, что кодирование Хаффмана является оптимальным вариантом среди представленных методов сжатия без потерь.

Заключение

В этом исследовании был проведен подробный анализ некоторых распространенных методов сжатия изображений без потерь, таких как: кодирование длин серий, алгоритмы Шеннона - Фано, Хаффмана, Лемпеля - Зива - Велча и арифметическое кодирование. Актуальность этих методов объясняется тем фактом, что большинство других, недавно разработанных алгоритмов без потерь или с потерями включают в свой состав данные методы. Все вышеупомянутые алгоритмы были проанализированы с использованием общего для всех набора медицинских данных. Для анализа были использованы стандартные показатели эффективности сжатия, такие как: время кодирования, время декодирования, средняя длина кода закодированного пикселя и коэффициент сжатия. По итогу исследования, было выяснено, что кодирование Хаффмана превосходит другие методы сжатия без потерь.

Литература

1. Bovik, A.C. Handbook of Image and Video Processing; Academic Press: Cambridge, MA, USA, 2010.
2. Golomb S. Run-length encodings (Corresp). IEEE Trans. Inform. Theory 1966, 12, pp. 399-401.
3. Bhavani, S.; Thanushkodi, K. A survey on coding algorithms in medical image compression. Int. J. Comput. Sci. Eng. 2010, 2, 1429–1434.
4. Huffman D.A., "A Method for the Construction of Minimum-Redundancy Codes," in Proceedings of the IRE, vol. 40, no. 9, pp. 1098-1101, Sept. 1952.
5. Welch, "A Technique for High-Performance Data Compression," in Computer, vol. 17, no. 6, pp. 8-19, June 1984.

COMPARATIVE ANALYSIS OF MODERN LOSSLESS IMAGE COMPRESSION METHODS

Dmitry G. Nuralin

*Student of group M091801(72), MTUCI
nurmity@yandex.ru*

Sergey V. Shevelev

*MTUCI, PhD., associate professor of MNaCS department
shevelev-s@yandex.ru*

Keywords: *Lossless compression, RLE, Shannon – Fano algorithm, Huffman algorithm, LZW, arithmetic coding.*

Everyday human activity leads to the generation of a huge amount of data, which creates big problems for their storage and transmission. Therefore, there is a growing demand for additional research in data compression and communication theory to solve these problems. The article provides a brief description of the most common modern lossless image compression methods. The criteria for evaluating the effectiveness of compression methods are considered. Based on these criteria, a comparative analysis of the algorithms is performed. Based on the analysis, conclusions are drawn about the best method.

МЕТОДЫ ОБНАРУЖЕНИЯ АНОМАЛИЙ В ПОТОКОВЫХ ВИДЕОДАНЫХ

Векшин Роман Владимирович
студент группы 2МИБ1701 МГУСИ
vekshin@gmail.com

Буянов Борис Яковлевич
МГУСИ, к.т.н., доцент кафедры ИСУИА
b.buyanov@gmail.com

Ключевые слова: Компьютерное зрение, обнаружение аномалий, разряженное кодирование, автоэнкодеры, многовариантное обучение, машинное обучение, искусственный интеллект.

В статье приведен обзор различных архитектур искусственных нейронных сетей (ИНС) и методов их обучения для обеспечения эффективного распознавания инцидентов, фиксируемых с помощью видеокамер системы охранного и технологического видеоконтроля. Детально рассмотрен один из наиболее эффективных методов обнаружения аномальных событий в видеопотоке – метод многовариантного обучения ИНС.

В настоящее время важным аспектом автоматизированных систем является оперативное и достоверное распознавание аномальных явлений – выходящих за рамки штатных событий. В частности, в системах охранного и технологического видеонаблюдения, при большом количестве видеокамер, необходимо обеспечить незамедлительное информирование оператора системы видеоконтроля в автоматизированном режиме о критически важных инцидентах, таких как автомобильные аварии, нападения на человека, массовые беспорядки, а также выявление нарушений технологического процесса на производстве.

В первую очередь, следует отметить, что для обнаружения определенных событий, фиксируемых видеокамерами, не применимы популярные и широко известные алгоритмы распознавания объектов на кадрах видеоряда, поскольку задачи детектирования «аномальных» инцидентов предполагают выявление признаков из действий объектов в сцене и их совокупностей.

Из передовых подходов к решению задач классификации в большом объеме размеченных данных одними из лучших результатов обладают различные вариации методов, основанных на алгоритмах разряженного кодирования (*sparse coding*) [1] и автоинкодерах (*deep auto-encoders*) [2]. Метод разряженного кодирования применительно к детектированию аномалий в видеоданных предусматривает обучение «словаря» ИНС на основе начальных фрагментов видеоряда, которые содержат шаблоны штатных событий охватываемой сцены. Последующая обработка видеоданных той же сцены будет выявлять события не поддающиеся идентификации из «словаря», и будут характеризованы как аномальные. Недостатком метода разряженного кодирования при детектировании аномалий в видеопотоке связан с отсутствием возможности достоверно составить «словарь» всех возможных штатных событий, а также в условиях значительных изменений окружающей среды в сцене, фиксируемой видеокамерой, метод разряженного кодирования подвержен значительному количеству ложных срабатываний. Одним из наиболее эффективных архитектур ИНС на текущий момент, с точки зрения обнаружения аномалий в большом объеме слабо размеченных видеоданных, является метод многовариантного обучения (*multiple instance learning*) [3]. Главным характерным отличием метода многовариантного обучения является разделение видеофрагментов, используемых в процессе обучения, на группы – пакеты. Каждому пакету присваивается маркер. Пакет маркируется как позитивная, в случае, если в нем присутствует видеофрагмент с инцидентом, который характеризуется как аномальный. Негативным маркером помечается пакет видеофрагментов, на которых сцена не содержит «аномальных» событий. Таким образом достигается, главным образом, возможность обучения ИНС слабо размеченными данными – нет необходимости тратить ресурсы на маркирование каждого экземпляра видеоданных подаваемого на обуче-

ние сети. Учитывая всю сложность и разнообразие событий на видеопоследовательностях, для достижения высокой точности детектирования, необходимо обучать ИНС на фрагментах видеоданных, содержащих не только «нормальные» события, но и аномальные.

На структуре (Рис.1) приведен образец архитектуры ИНС с многовариантным обучением [4]. Перед обработкой, видеоряд разбивается на фрагменты, из которых формируются пакеты – позитивный и негативный. Пакеты подаются на вход сверточной ИНС C3D ConvNets [5], которая обеспечивает извлечение пространственно-временных признаков, из видеофрагментов в пакете. На следующем шаге применяется полносвязная многослойная сверточная ИНС, обеспечивающая вычисление метрики «аномальности» между образцами с максимальными значениями в негативном и позитивном пакетах. Для обучения ИНС использовалась база данных, состоящая из 1900 видеозаписей различной длины – от десятков секунд до 10 минут, из которых 950 – «аномальные» и 950 – «нормальные». На видеозаписях содержатся сцены инцидентов, такие как ограбления, дорожные инциденты, взрывы, уличные конфликты и пр. Видеоролики имеют размер 240x320 пикселей и скорость отображения 30 кадров в секунду и делятся на фрагменты – по 32 фрагмента в пакет. Пространственно-временные характеристики видеофрагментов, состоящих из 16 кадров, получаемые на выходе ИНС C3D, приведены к L2-норме, и далее 4096-мерный вектор подаются на вход полносвязной трехслойной сверточной ИНС с 512, 32 и 1 нейронами соответственно. Использование 3-слойной ИНС наиболее эффективно и увеличение количества слоев не приводит к качественному улучшению результатов. В качестве функций активации используется *ReLU* на первом слое и сигмоида на последнем с применением метода оптимизации обратного распространения ошибки – *adagard*, с начальной скоростью обучения 0,001. Для снижения вероятности переобучения применяется метод *dropout*-регуляризации [6]. В обучающую выборку включены 800 «нормальных» видеороликов и 810 «аномальных», и тестовая выборка содержит 150 и 140 видеороликов соответственно.

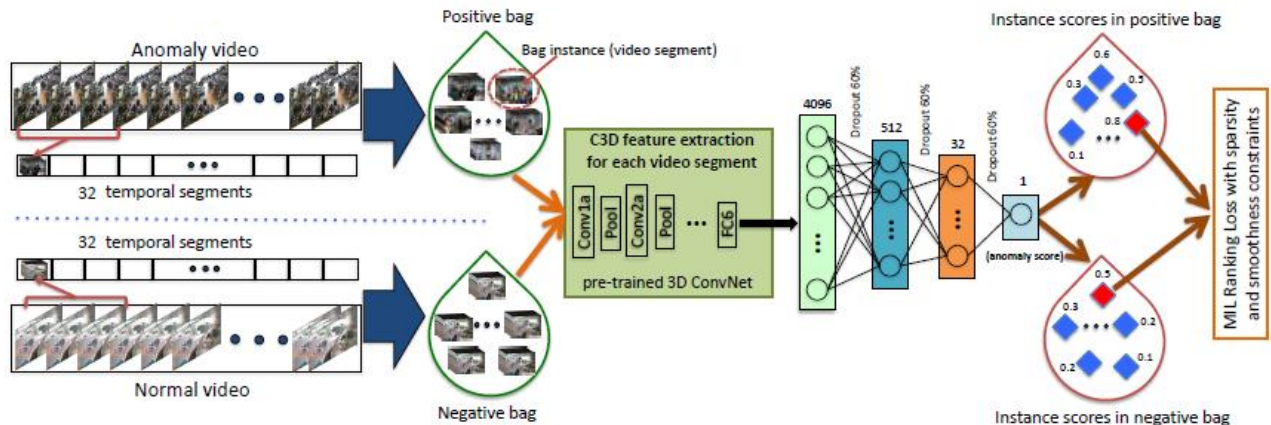


Рис. 1. Структура процесса детектирования аномалий на видео методом многовариантного обучения

В тех случаях, когда для обучения доступно большое количество маркированных данных задача сводится к решению классической регрессии через обучения с учителем, и в качестве классификатора можно использовать метод опорных векторов (1).

$$\min_w \left[\frac{1}{T} \sum_{i=1}^k \max(0, 1 - y_i(w, \phi(x) - b)) \right] + \frac{1}{2} \|w\|^2, \quad (1)$$

В приведенном выражении функция под знаком суммы - кусочно-линейная функция потерь, в которой y_i – метка фрагмента, $\phi(x)$ – значение каждого фрагмента, w – непосредственно классификатор, b – смещение. Для создания надежного классификатора необходимо строгое маркирование положительных и отрицательных образцов, что сложно в случае необходимости расчета временных маркеров применительно к видеоданным. Многовариантный метод обучения не учитывает временные характеристики и рассчитывает общее значение, является ли фрагмент в пакете аномальными или нет. Применив функцию потерь к пакету фрагментов, получим (2):

$$\min_w \left[\frac{1}{m} \sum_{j=1}^m \max(0, 1 - Y_{\beta_j} (\max_{i \in \beta_j} (w \cdot \phi(x_i)) - b)) \right] + \|w\|^2, \quad (2)$$

, где Y_{β_j} – метка пакета, j – общее количество пакетов, $\phi(x_i)$ – значение каждого фрагмента, w – классификатор, b – смещение. Таким образом приведенный метод рассматривает поиск аномалий в видеоряде с точки зрения решения задачи ранжирования, нацеленной на максимальные значения аномальных фрагментов в объеме всех видеоданных, с промежуточным ранжированием в объеме пакета при обучении, в отличие от иных подходов, рассматривающих задачу с точки зрения обнаружение определенных паттернов, что ведет к низкой результативности таких методов. Следовательно, задействовав функцию потерь удастся потенциально выявить более высокие значения аномальных инцидентов на видео относительно штатных событий в каждом из пакетов, применяя функцию многовариантного ранжирования (3):

$$\max_{i \in \beta_p} f(\tau_p^i) > \max_{i \in \beta_n} f(\tau_n^i), \quad (3)$$

, где β_p – пакет с позитивным маркером, β_n – пакет с негативным маркером, $f(\tau_p)$ – прогнозные значения «аномального» фрагмента, $f(\tau_n)$ – прогнозные значения «нормального» фрагмента. Выбор наивысших значений «аномальных» фрагментов происходит на уровне пакетов, что позволяет обеспечить высокую скорость обработки и достаточный уровень точности метода. «Аномальные» фрагменты с максимальным значением «аномальности» в позитивных пакетах маркировать как «истинно позитивные». «Нормальные» фрагменты с максимальным значением «аномальности» в позитивных пакетах маркировать как «ложно позитивные». Ранжирование «истинно позитивных» и «ложно позитивных» фрагментов возможно через применение кусочно-линейной функции потерь (4):

$$l(\beta_p, \beta_n) = \max(0, 1 - \max_{i \in \beta_p} f(\tau_p^i) + \max_{i \in \beta_n} f(\tau_n^i)), \quad (4)$$

Для повышения точности следует учитывать временные характеристики последовательностей аномальных фрагментов. Поскольку «аномальные» инциденты, как правило, возникают на короткий промежуток времени возникает разреженность «аномальных» фрагментов в контексте всего пакета, и наблюдается распределенный во времени характер сигнала (5):

$$l(\beta_p, \beta_n) = \max(0, 1 - \max_{i \in \beta_p} f(\tau_p^i) + \max_{i \in \beta_n} f(\tau_n^i)) + \sigma_1 \sum_i^{n-1} (f(\tau_p^i) - f(\tau_p^{i+1}))^2 + \sigma_2 \sum_i^n f(\tau_p^i), \quad (5)$$

, где выражения под знаком суммы добавляют к значениям сигнала «аномальных» фрагментов дополнительное распределение и корреляцию соответственно. В приведенной модели многовариантного обучения применяется обратное распространение ошибки от максимальных значений как в позитивных, так и в негативных пакетах, что за счет итоговой генерализации (6) позволяет достичь высокой точности (Рис 2).

$$L(W) = l(\beta_p, \beta_n) + \|W\|_F, \quad (6)$$

Приведенный ROC-анализ (Рис.3) наглядно показывает превосходство метода на основе многовариантного обучения ИНС (красная кривая) над методами на основе разреженного кодирования (голубая кривая), использующих автоинкодеры (черная кривая) и бинарный классификатор (синяя кривая).

Заключение

Результаты показывают, что при наличии относительно большого количества видеоданных с отсутствием и частичным наличием на них инцидентов, которые могут быть характеризованы как аномалии, с помощью метода многовариантного обучения, ИНС может обнаруживать инцидент с высокой точностью по времени и классу инцидента. По количеству «ложных» срабатываний метод существенно превосходит передовые алгоритмы, что говорит о потенциальном прорыве в развитии систем обработки видеоданных и распознавании инцидентов на них.

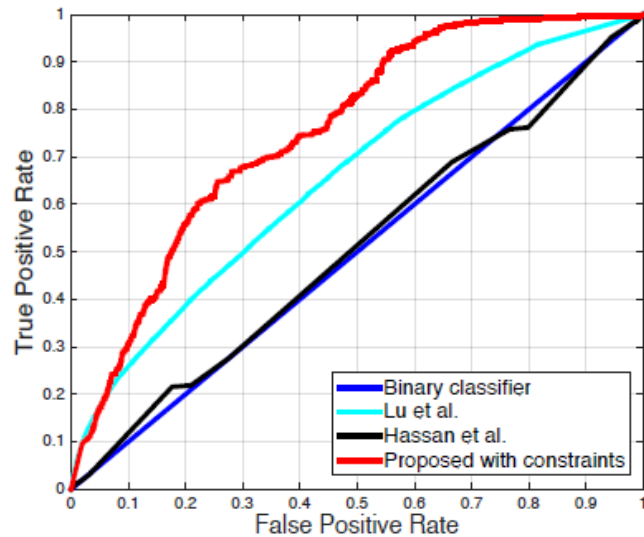


Рис. 3. ROC-анализ

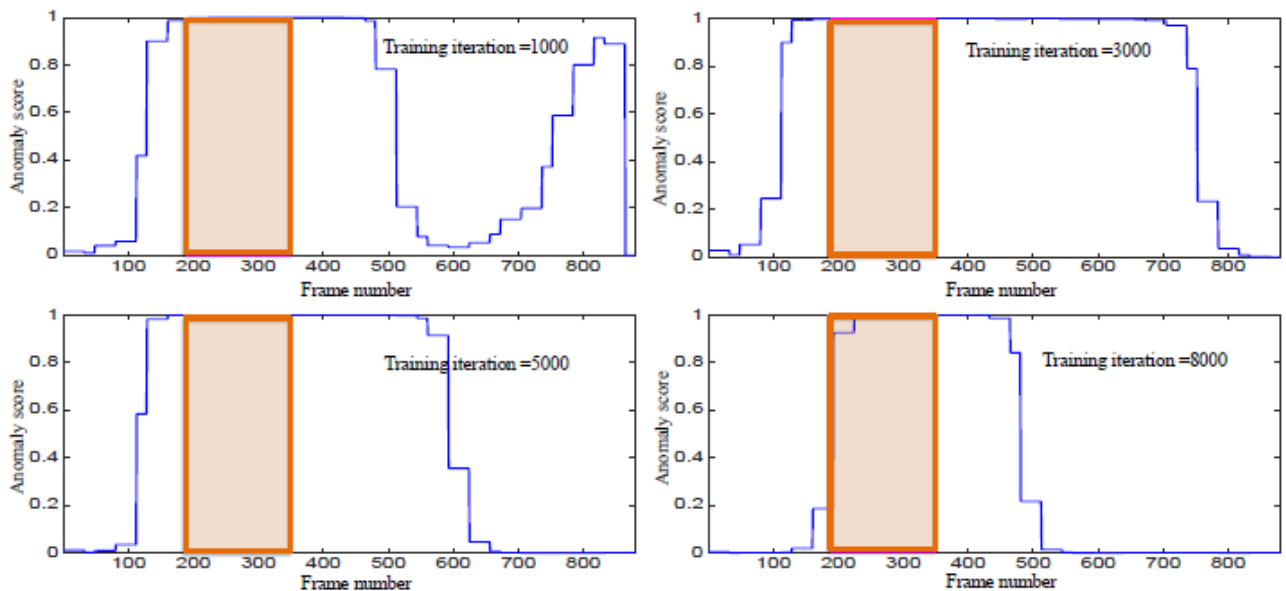


Рис. 2. Показатели точности обнаружения аномальных инцидентов при помощи ИНС с многовариантным обучением при различном количестве эпох (красная зона – аномалия, синяя зона – результат ИНС)

Литература

1. Abnormal Event Detection at 150 FPS in MATLA // cse.cuhk.edu.hk/en URL: https://www.cse.cuhk.edu.hk/leojia/papers/abnormaldect_iccv13.pdf (дата обращения: 22.10.2019).
2. Abnormal Event Detection in Videos using Spatiotemporal Autoencoder // arXiv.org URL: <https://arxiv.org/pdf/1701.01546.pdf> (дата обращения: 22.10.2019).
3. Multiple Instance Learning: A Survey of Problem Characteristics and Applications // arXiv.org URL: <https://arxiv.org/pdf/1612.03365.pdf> (дата обращения: 25.10.2019).
4. Real-world Anomaly Detection in Surveillance Videos // arXiv.org URL: <https://arxiv.org/pdf/1801.04264.pdf> (дата обращения: 25.10.2019).
5. Learning spatiotemporal features with 3d convolutional networks // arXiv.org URL: <https://arxiv.org/pdf/1412.0767.pdf> (дата обращения: 25.10.2019).
6. Dropout: A Simple Way to Prevent Neural Networks from Overfitting // cs.toronto.edu URL: <http://www.cs.toronto.edu/~hinton/absps/JMLRdropout.pdf> (дата обращения: 25.10.2019).

ANOMALY DETECTION METHODS IN STREAMING VIDEO DATA

Roman V. Vekshin

Student of group 2MIB1701, MTUCI

vekshin@gmail.com

Boris I. Buyanov

MTUCI, PhD., associate professor of ISAAC department

b.buyanov@gmail.com

Key words: *computer vision, anomaly detection, sparse coding, autoencoders, multiple instance learning, machine learning, artificial intelligence.*

The article represents an overview of various architectures of artificial neural networks (ANNs) and their training methods to ensure effective recognition of incidents recorded via video cameras of the security and technological monitoring systems. One of the most effective methods for detecting abnormal events in a streaming video is considered - the multiple instance learning method.

ИССЛЕДОВАНИЕ ПОДХОДОВ К УПРАВЛЕНИЮ РИСКАМИ ПРИ СОЗДАНИИ ОБЛАЧНОЙ ИНФРАСТРУКТУРЫ ТЕРРИТОРИАЛЬНО-РАСПРЕДЕЛЁННОЙ КОМПАНИИ

Петухов Денис Алексеевич
студент группы М111801(74) МГУСИ
zmreact@gmail.com

Докучаев Владимир Анатольевич
МГУСИ, д. т. н., профессор, заведующий кафедрой МСuУС
v.dok@tlsf.ru

Ключевые слова: управление рисками, облачная инфраструктура, фреймворк, руководство, анализ

Проведено исследование в области руководств по управлению рисками, применимых при создании облачной инфраструктуры территориально-распределённой компании. В частности, рассматриваются руководства ГОСТ Р ИСО 31000:2018, COSO ERM 2017, NIST SP.800-37R2 и ISACA Risk IT Framework. Выделены отличия подходов к управлению рисками в различных руководствах. Проведены исследования сведений, необходимых для исполнения данных документов. В частности, было предложено применение документов ENISA и результатов научной работы, представленной на 3-ей международной конференции IEEE. Обозначена главная задача для дальнейших исследований.

За последние несколько лет развитие вычислительной техники и информационных технологий открыло возможность создания нового рынка цифровых услуг. Облачные технологии являются одним из примеров концепции услуги, которая предоставляется потребителю по требованию. Согласно обзорам Gartner рынок облачных услуг в 2018 году увеличился на 19% по сравнению с предыдущим годом [1]. Эта статистика подтверждает тенденцию смены приоритета создания компаниями собственных вычислительных платформ в пользу использования внешних [2]. Это открывает возможности для партнерских отношений компаний в предоставлении цифровых услуг. Для любой компании, определившей облачную инфраструктуру одним из приоритетных инструментов для своего бизнеса, значимым является анализ рисков при создании такой технологической инфраструктуры.

Анализ компонентов облачной инфраструктуры

Облачная инфраструктура в общем случае включает в себя компоненты ЦОД, а также технологии, позволяющие предоставлять потребителю некоторый цифровой или физический ресурс по требованию. Так, например, инструмент OpenStack Ironic осуществляет предоставление физических вычислительных машин [3]. Он представляет собой набор API, взаимодействующий с физическими машинами. Как правило, используются интерфейсы PXE и IPMI для подготовки, включения и выключения машин. Применение технологий виртуализации позволяет предоставлять по требованию виртуальные программные, сетевые, вычислительные компоненты, или диски, выделяемые из общего пула физических ресурсов. Компоненты ЦОД могут включать в себя следующие компоненты, представленные на рис. 1 [2].

При создании облачной инфраструктуры целесообразно рассматривать риски, связанные с данными компонентами, но не ограничиваться ими. Предлагается учитывать риски, возникающие на этапах проектирования, ввода в эксплуатацию, эксплуатации и вывода из эксплуатации данной системы. Так же дополнительно необходимо обратить внимание на фактор территориального распределения компании, учёт которого раскрывает специфические риски, связанные в том числе с распределённой структурой облачной инфраструктуры, показанной на рис. 2 [2].

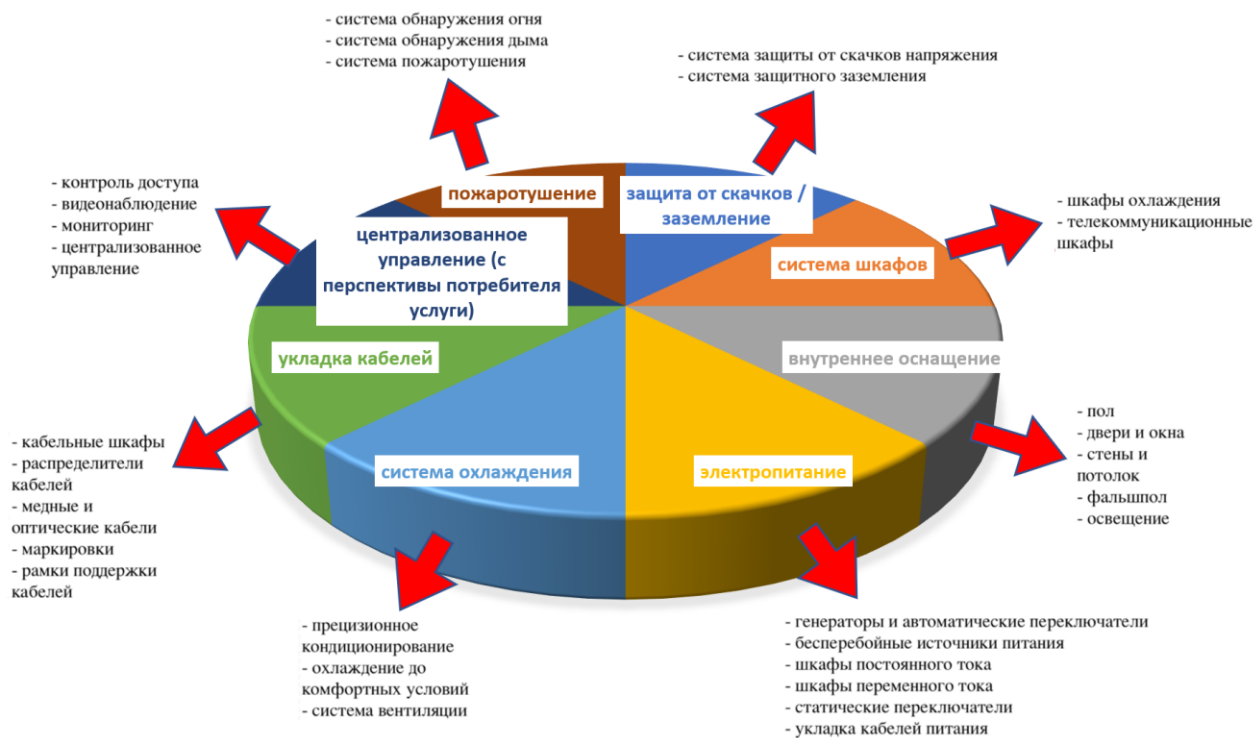


Рис. 1. Компоненты инфраструктуры ЦОД

Анализ нормативных документов в области управления рисками

В области управления рисками существует как минимум четыре нормативных документа - руководства, в которых описывается структура и методология задачи управления рисками. Этими документами являются ГОСТ Р ИСО 31000:2018, COSO и NIST SP.800-37R2, ISACA Risk IT Framework. Каждый документ содержит свои особенные рекомендации по управлению рисками. Для анализа общей структуры управления рисками сведём её составляющие в таблицы 1 и 2, сравним и выделим особенности, на которые акцентируется внимание в данных руководствах.

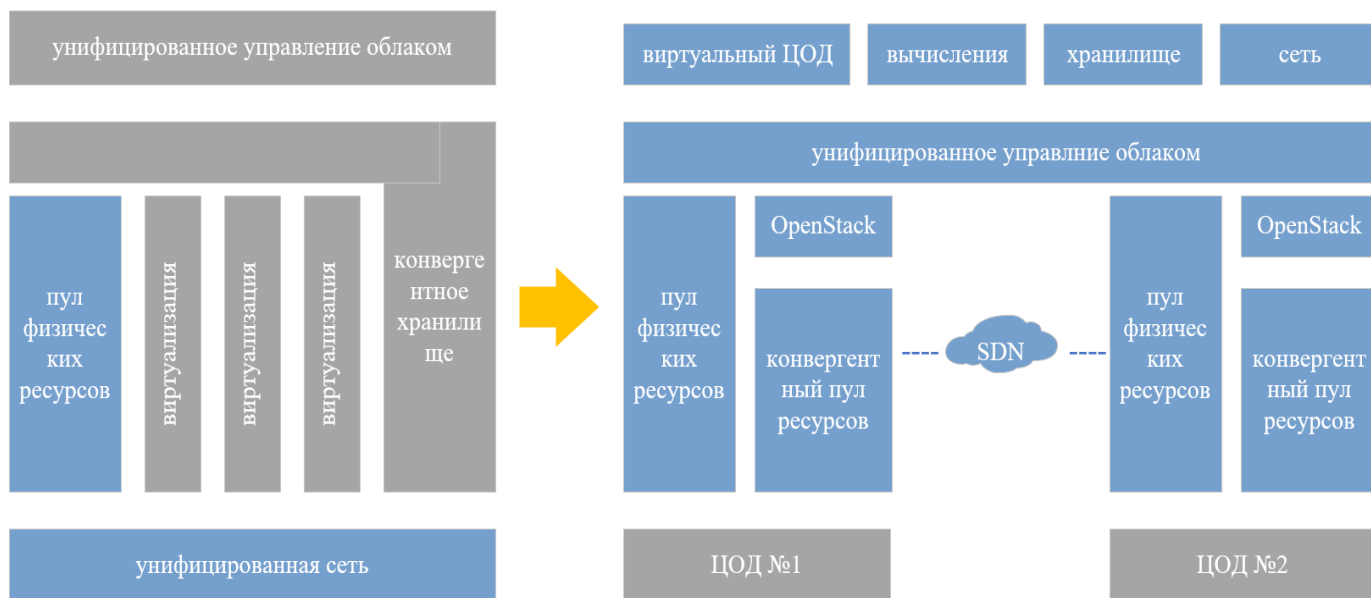


Рис. 2. Логическая инфраструктура локальной и распределенной облачной инфраструктуры

Таблица 1

Структуры управления рисками в различных руководствах

| Рассматриваемый предмет | ГОСТ | COSO | NIST | ICASA |
|--------------------------------------|---|---|---|--|
| Процесс на протяжении всех этапов | Обмен информацией, мониторинг, документирование | Нет данных | Нет данных | Обмен информацией, отчетность |
| Мотивация управления рисками | Создание и защита стоимости | Миссия, видение, ключевые ценности компании | Результаты подготовительных мероприятий на организационном уровне и на системном уровне | Бизнес цели компании |
| Последовательность исполнения этапов | Этапы выполняются последовательно | Этапы выполняются параллельными этапами: 1 и 5 вместе, а также 2,3,4 вместе | Этапы могут выполняться в любой последовательности | Этапы используются при в соответствии с алгоритмом управления риском |

Таблица 2

Этапы управления рисками в различных руководствах

| | | | | |
|--------|---|---|--|------------------------------|
| Этап 1 | Определение предметной области, контекста и критериев риска | Руководство и культура в отношении к рискам | Подготовительные мероприятия на организационном уровне и на системном уровне | Политика в отношении к риску |
| Этап 2 | Идентификация, анализ и оценка риска | Определение бизнес-ценностей | Классификация влияния угроз на информацию и систему | Оценка риска |
| Этап 3 | Воздействие на риск | Идентификация, оценка, приоритизация риска, реализация отклика и разработка профиля риска | Выбор мер действия по отношению к риску | Отклик на риск |
| Этап 4 | Отсутствует | Мониторинг и пересмотр | Реализация мер и описание их применения в среде эксплуатации | Отсутствует |
| Этап 5 | Отсутствует | Обмен информацией и отчетность | Оценка применения мер | Отсутствует |
| Этап 6 | Отсутствует | Отсутствует | Рассмотрение результата руководством | Отсутствует |
| Этап 7 | Отсутствует | Отсутствует | Мониторинг системы и мер, документирование изменений, отчетность об угрозах | Отсутствует |

В данной таблице структуры, описанные в документах, были адаптированы для максимального соответствия и возможности их сравнения. Из таблицы 1 видно, что во всех руководствах на первое место ставится задача определения бизнес-целей предприятия. Так, например, для провайдера цифровых услуг одной из бизнес-целей может являться получение выручки и поддержание репутации при предоставлении некой услуги с заданными параметрами качества. Вторым этапом является идентификация и оценка риска. Третьим этапом является реализация отклика на риск. Между всеми этапами происходит обмен информацией.

Модели ГОСТ, COSO и NIST позиционируются, как универсальные. В то же время, модель ISACA отличается направленностью на ИТ инфраструктуры, что имеет значение в исследуемой теме. В частности, в модели ISACA особенно выделяется необходимость в рассмотрении нескольких сценариев риска, что является важным при создании облачной инфраструктуры (рисунок 3).

Из предшествующего абзаца следует, что для уточнения применения данных моделей к исследуемой теме их необходимо рассматривать вместе с дополнительными сведениями. В каждой из моделей выделяются этапы обмена информацией и отчетности. Эти этапы в том числе подразумевают, что организация при управлении рисками в качестве одного из источников данных может использовать общедоступные отчеты. В данном случае на этапах идентификации риска она может руководствоваться такими документами, как отчеты ENISA:

- (ENISA Security aspects of virtualization, 2017);
- (ENISA Hardware Threat Landscape and Good Practice Guide, 2017);
- (ENISA Threat Landscape Report 2018. 15 Top Cyberthreats and Trends, 2019).

Данные отчеты охватывают угрозы облачной инфраструктуры с точки зрения виртуализации, аппаратной части и кибербезопасности. В отчетах сведены статистики угроз, а также проведена их классификация.

Вопрос оценки риска требует использование математического аппарата. Технические риски на практике принято оценивать произведением вероятности наступления угрожающего события компоненту технической системы на вероятность причинения определенного уровня ущерба компоненту. Риск в таком случае можно рассчитать по формуле 1, приведённой ниже [4]:

$$R = p(a, t) \cdot d_l(a, t) \quad (1)$$

где R – риск, %

p – вероятность наступления угрожающего события t , %

d_l – вероятность причинения угрожающим событием t ущерба уровня l компоненту a , %

a – компонент технической системы

t – угрожающее событие

Нахождение переменных данной формулы требует изучение таких отчетов, как ENISA, а также результатов исследований, таких как (P. Saripalli, B. Walters A Quantitive Impact and Risk Assessment Framework for Cloud Security, IEEE 3rd International Conference on Cloud Computing, 2010). Также компания, относящаяся к субъектам с критической информационной инфраструктурой по 187-ФЗ может руководствоваться данными категорирования объектов КИИ.

Автор статьи считает, что проведённые исследования можно продолжить с целью формирования классификации рисков применительно к исследуемому вопросу. Классификация рисков может быть одним из документов отчетности организации, направленным на предоставление информации для принятия решений по риск-менеджменту (ГОСТ Р ИСО 31000:2018, 2018), (NIST SP.800-37R2, 2018) и является одной из определяющих эффективности управления рисками компании [5].

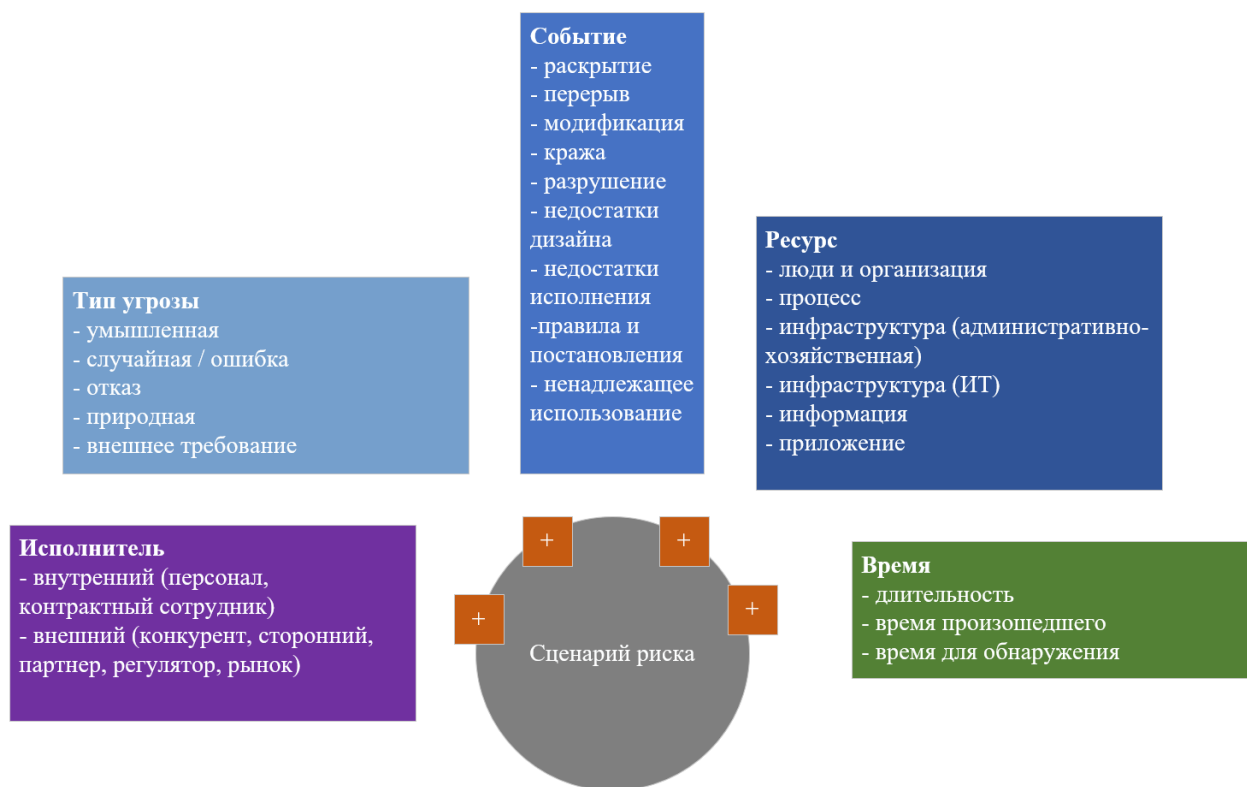


Рис. 3. Конструктор сценария риска в модели ISACA
 Источник: Risk It Framework / ISACA. 2009. Изображение: электронное

Заключение

Создание облачной инфраструктуры является сложным техническим процессом, который включает этапы проектирования, ввода в эксплуатацию, эксплуатации и вывода из эксплуатации системы. Проведённый анализ нормативных и правовых документов, применяемых компаниями при управлении рисками, позволил определить уточняющие документы в области облачных технологий, которые следует использовать при создании подобных систем. В качестве следующего этапа исследований предлагается разработка классификации рисков, возникающих при создании облачной инфраструктуры территориально-распределённой компании.

Литература

1. Cloud computing (world market) — Текст : электронный // TADVISER: [сайт]. 2019. URL: [http://tadviser.com/index.php/Article:Cloud_computing_\(world_market\)](http://tadviser.com/index.php/Article:Cloud_computing_(world_market)). (дата обращения: 27.07.2019).
2. HCIA Storage Training Material V4.0. / Huawei. 2018. Текст. Изображение : электронные.
3. OpenStack Docs: Bare Metal Service User Guide. Текст : электронный // OpenStack Docs : [сайт]. — 2019. — URL: <https://docs.openstack.org/ironic/pike/user/>. (дата обращения: 20.10.2019).
4. Measuring IT risk — Текст : электронный // Wikipedia : [сайт]. 2019. URL: https://en.wikipedia.org/wiki/IT_risk#Measuring_IT_risk. (дата обращения: 05.10.2019).
5. ПАО «Ростелеком». Годовой отчёт 2017. Управление рисками — Текст : электронный // Ростелеком : [сайт]. 2017. URL: http://ar2017en.rostelecom.ru/rostelecom/annual/2017/gb/Russian/pdf/RT_AR_2017_Ch5_rus.pdf. (дата обращения: 01.11.2019).

RISK MANAGEMENT RESEARCH FOR CLOUD INFRASTRUCTURE OF TERRITORIAL DISTRIBUTED COMPANY

Denis A. Petukhov

*Graduate student, group M111801 (74), MTUCI
zmreact@gmail.com*

Vladimir A. Dokuchaev

*MTUCI, professor Head of Department MN&TS
v.dok@ilsf.ru*

Keywords: *risk management, cloud infrastructure, framework, guide, analysis*

In this article a research was conducted on risk management guides that are applicable while creating a cloud infrastructure of a territorial distributed company. The GOST R ISO 31000:2018, COSO ERM 2017, NIST SP.800-37R2 and ISACA Risk IT Framework were examined. The differences between approaches were highlighted. A research of the source data for execution of these documents was performed. Particularly, an application of ENISA documents and a research work presented on IEEE 3rd International Conference was proposed. Lastly, the main task for the further research was defined.

МАШИННОЕ ОБУЧЕНИЕ В ИГРОВЫХ ПРОЕКТАХ

Абросимов Данил Дмитриевич
Студент группы БСТ1602-МТУСИ
dmitriwb@mail.ru

Кононов Руслан Валерьевич
Студент группы БСТ1602-МТУСИ
dremming@mail.ru

Беленькая Марина Наумовна
МТУСИ, доцент кафедры МСиУС

Ключевые слова: Искусственный интеллект, игры, геймдев, разработка игр, машинное обучение, алгоритмы машинного обучения.

Введение в тему машинного обучения. Приведены алгоритмы машинного обучения. Показаны преимущества использования машинного обучения в игровых проектах. Представлен пример машинного обучения в играх на основе игры «Гонки».

В наше время невозможно представить жизнь без информационно-технических технологий, все мы уже привыкли пользоваться поисковыми системами не только по тексту, но и по картинкам. Также, очень удобно получать предложения от магазинов, которые учитывают наши вкусы и даже место проживания, а продавцам стало удобно размещать адаптивную рекламу для продвижения своего продукта только тем, кто может быть в нем заинтересован. Все это стало возможным благодаря обработке огромного количества данных и обучению приложений фильтрации информации по заранее заданным признакам. Люди всегда стремились автоматизировать рутинную работу, и обработка данных не стала исключением. Цель машинного обучения – предсказать результат по входным данным. Чем разнообразнее поступают данные на вход, тем легче машина найдет закономерность в данных и выведет результат. К примеру, мы можем научить машину определять: является ли письмо спамом или нет, подав на входы примеры спама. После анализа, машина найдет закономерность в письмах и будет отсеивать все поступающие письма по найденным признакам. При этом, она будет учитывать новые признаки в письмах, которые посчитала спамом.

Машинное обучение является основой многих важных приложений, в том числе веб-поиска, защиты от спама, распознавания речи, рекомендации по продукту и многого другого. Подобное обучение освобождает людей от дополнительной работы, однако, чаще всего, это является очень трудоемкой задачей. Совершенно непросто создать адаптивную рекламу, которая сама будет подстраиваться под пожелания каждого пользователя. Во-первых, сложно понять: чему нужно обучать программу, во-вторых, как именно эту программу обучать. Существует огромное количество вариантов изучения машинного обучения, в данной статье будет рассмотрен метод, который можно визуализировать с помощью виртуальной среды. Его плюс состоит в том, что он очень близок к реальности. Данный способ заключается в использовании игрового ядра для визуализации результатов, которых мы добились. Это очень важно в машинном обучении, ведь только так мы сможем понять: в правильном ли направлении наша программа учится или нет. Таким образом, симулируя искусственный интеллект, основанный на машинном обучении в виртуальной среде, у нас получается результат, приближенный к реальности.

Применение машинного обучения в игровых проектах

Машинное обучение – это набор алгоритмов и технологий, предназначенных для улучшения процесса поиска и принятия решений машиной. Данная технология строится на основе статических методов, созданных для поиска правил и закономерностей в рассматриваемых процессах. Основная цель машинного обучения – разработка модели, способной создать причинно-следственную связь, а также, уметь прогнозировать ситуацию.

В настоящее время, машинное обучение нашло широкое применение в сфере игровой индустрии среди больших корпораций из-за своих возможностей. Например, данные алгоритмы позволяют правильно выбрать цену для продукта, бороться с мошенниками, моделировать поведение реального игрока. Возможности у машинного обучения действительно колоссальные, именно поэтому эта область является наиболее быстро развивающейся в наши дни.

Отладка игровых карт

Одним из главных достоинств машинного обучения является возможность имитации поведения реального игрока. Данная возможность используется тогда, когда необходимо найти ошибки в дизайне карты.

Увеличение аудитории

Основная цель каждого игрового проекта – это заработок денег. Как правило, увеличения дохода пропорционально аудитории игры. Именно для этого большие компании используют методы машинного обучения. Искусственный интеллект исследует интересы пользователей, их реакцию на игровые факторы, которые влияют на различного рода бизнес метрики. Данные, собранные машиной, интерпретируются и обрабатываются, а также, на их основе выносятся решения по изменению определенных игровых настроек для удержания аудитории.

«Гибкий» игровой искусственный интеллект

У игрового может быть большое количество пользователей. Очевидно, что вся масса игроков делится на некоторое количество категорий по множеству критериев. Для того, чтобы не было такого, что игра интересна лишь узкому кругу пользователей и применяется машинное обучение. Целью такого вида алгоритмов является подстраивание поведения искусственного интеллекта под игрока каждой категории и создание идеальных условий для того, чтобы выбранный игрок не заскучал.

Предсказание ухода клиентов

Не секрет, что игра может попросту приесться, надоест клиенту. Цель алгоритмов машинного обучения – проанализировать данные каждого клиента и ввести определенные изменения для предотвращения потери интереса к игре. Примером может являться случай, когда игрока слишком часто «убивают» в игре. Для того, чтобы пользователь попросту не бросил играть из-за слишком умного искусственного интеллекта игры, алгоритмы уменьшают определенные параметры игрового ИИ, тем самым, уменьшая и сложность прохождения игровых уровней.

Индивидуальные предложения

Основной заработок с игр, созданных для мобильных устройств, является покупка внутриигровых ресурсов. Для того, увеличения доходов и были созданы алгоритмы, позволяющие определить, какую покупку захочет совершить клиент. Машина собирает данные о внутриигровых покупках, активности и ряде других параметров для создания цифрового портрета пользователя. После этого происходит определение того, каким товаром может быть заинтересован выбранный игрок.

Частота показываемой рекламы

Вторым по величине способом заработка на мобильной игре является монетизация показываемой рекламы. Обратной стороной медали является тот факт, что пользователю может попросту не понравиться большой объем рекламы в приложении. В этой ситуации и приходят на помощь алгоритмы машинного обучения. Программа анализирует активность пользователей, а также то, сколько игроков покинуло проект и находит между ними ряд общих признаков, по которым строит выводы, позволяющие выставить оптимальную частоту показа рекламы для обеспечения заработка и уменьшения оттока клиентов.

Алгоритмы машинного обучения

Для реализации машинного обучения было придумано множество алгоритмов. В данной статье будет разобрано три самых популярных и эффективных из них.

Обучение с подкреплением

Данный алгоритм основан на идее взаимодействия искусственного интеллекта с окружающей средой, а также анализом результатов, полученных при взаимодействии и нахождением порядка действий, нужных для достижения максимально долговременного выигрыша. Иначе говоря, машина ищет наиболее выгодную для себя стратегию, которая обязательно приведет к получению желаемого результата.

Стоит отметить, что возможных стратегий может быть огромное количество, поэтому, чтобы правильно выбрать нужную, машина должен протестировать все стратеги несколько раз, что крайне невыгодно. Именно поэтому, данный алгоритм допускает влияние результатов, полученных после тестов одной стратегии на оценку других. Проще говоря, если награда, которую машина получит после выполнения задания, в котором нужно будет убить дракона с помощью палки, будет больше, чем награда за задание, в котором нужно будет всего лишь добежать куда-то, то машина сначала оценит награду за первое задание в 1(максимальная выгода), а награду за второе задание, к примеру, в 0.5. Очевидно, что сначала машина будет пытаться получить награду за более сложное задание, ведь она больше, но после столкновения с окружающей средой(палка, дракон) и после ряда неудач, ценность данной награды в глазах алгоритма будет падать до тех пор, пока она не станет меньше, чем награда за более легкое задание.

Имитационная система обучения

При данном типе обучения алгоритм лишь наблюдает за действиями игрока с окружающей средой и обучается, структурируя данные, полученные из наблюдений. Например, с помощью этого алгоритма возможно создать идеального игрока в какую-либо игру, но для этого, потребуется огромное количество времени, потраченного на игру реальным человеком, нужное для того, чтобы машина проанализировала поведение геймера.

Обратное обучение с подкреплением

Машина обучается, следя за клиентом, пытаясь понять, какие цели он преследует при определенном взаимодействии с окружающей средой. Например, в случае, когда перед игроком появится непобедимый дракон, а из оружия у него будет только палка, что он выберет: бездумно пойдет в бой или отступит и вернется с уже более совершенным оружием. Этот вид обучения повсеместно используется не только в играх, но и в реальной жизни. Примером может служить технология беспилотного управления автомобилем. Причиной этому является огромный потенциал данного алгоритма, ведь если когда-нибудь ИИ достигнет интеллекта человека, то тогда он сможет понимать его цели и помогать их воплощать.

Заключение

1. Алгоритмы машинного обучения существуют уже десятилетия, но они приобрели особую популярность из-за новой волны развития искусственного интеллекта. В особенности, модели глубокого обучения поддерживают современные приложения искусственного интеллекта. Большое количество крупных компаний, таких как Google, Microsoft, Amazon и другие, стараются подписать заказчиков на услуги платформы, которые охватывают спектр функций машинного обучения. Особенно популярными функциями являются обучение приложений, сбор, анализ и подготовку данных.

2. Машинное обучение позволяет симулировать и визуализировать среду, в которой удобно наблюдать за результатами машинного обучения. Машинное обучение – это очень перспективное направление в том числе и в играх, так как оно позволяет решать большое количество проблем, связанных с созданием интересного игроку искусственного интеллекта, составляет персональные предложения каждому пользователю в внутриигровом магазине, а также делает игру интересней.

3. Как пример, нам удалось найти пример продукта, созданного на игровом движке Unreal Engine 4, в котором используются алгоритмы машинного обучения. Код данного продукта можно найти на GitHub [1]. В данном проекте разработчику визуализировать процесс машинного обучения в виде гонок, в которых искусственный интеллект подстраивается под ситуацию и сам решает, как ему лучше преодолевать препятствия.

4. Изучать машинное обучение с помощью игрового ядра очень удобно и перспективно: оно позволяет нам автоматизировать многие процессы и создавать искусственный интеллект, который может принимать решения, основываясь на своих ошибках.

Литература

1. Пример на GitHub - https://github.com/dani2442/AI_Car.

MACHINE LEARNING IN GAME DEVELOPMENT

Danil D. Abrosimov

*Student of the BST1602-MTUSI group
dmitriwb@mail.ru*

Ruslan V. Kononov

*Student of the BST1602-MTUSI group
dremming@mail.ru*

Marina N. Belenkaya

MTUCI, Associate Professor of the Department of MSiUS

Key words: *Artificial Intelligence, games, game development, machine learning, machine learning algorithms.*

Introduction to the topic of machine learning. The machine learning algorithms are given. The advantages of using machine learning in game projects are shown. An example of machine learning in games based on the Racing game is provided.

ПОСТРОЕНИЕ КАСКАДНОЙ СХЕМЫ ЭФФЕКТИВНОГО КОДИРОВАНИЯ ГЕНОМНЫХ ДАННЫХ С УДАЛЕНИЕМ РАЗНЫХ ТИПОВ ИЗБЫТОЧНОСТИ ДАННЫХ

*Келимов Эмиль Рустемович,
МТУСИ, Студент группы М111801(74)
semkel07@gmail.com*

*Гузеев Алексей Валерьевич,
МТУСИ, к.т.н., доцент кафедры МСцУС
a_guzeev@mtuci.ru*

Ключевые слова: биоинформатика, словарный метод кодирования информации, алгоритм Хаффмана, арифметический метод кодирования, алгоритм LWZ, сжатие данных, геномные данные.

В данной статье был предложен словарный метод кодирования геномных данных путем поиска подстрок наибольшей длины. В статье рассмотрены результаты кодирования по предложенному алгоритму. Сравнение данного метода с популярными методами кодирования информации, такие как алгоритм Хаффмана, арифметический метод, алгоритм LZW. Также в статье рассматриваются результаты каскадной схемы кодирования геномных данных с предложенным методом.

С каждым годом количество генетических данных растет в невероятных масштабах, то вопрос о хранении данных растет довольно остро. Одним из решений в вопросе оптимизации хранения геномных данных является сжатие данных. Необходимо понимать, что в генетических данных хранятся биологическая информация, при изменении которой можно неверно интерпретировать функцию последовательности. Поэтому одним из требований к кодированию геномных данных является то, что алгоритм должен сжимать данные без потерь.

Пусть $A = \{a_1, a_2, a_3, \dots, a_n\}$ будет исходным алфавитом, где n – это количество различных символов, а исходной последовательностью для последующего кодирования пусть будет S . Последовательности ДНК состоят из четырех азотистых оснований [5]: аденин (A), тимин (T), гуанин (G), цитозин (C). Тогда можно сказать, что исходный алфавит для геномной последовательности S является $A = \{A, T, G, C\}$.

Предложение алгоритма сжатия геномных данных.

Одной из особенностей генетических последовательностей является то, что из-за различных биологических процессов [1] некоторые последовательности могут появляться множество раз на небольших участках генома. Рассмотрим на конкретном примере логику предлагаемого алгоритма сжатия данных. Для этого возьмем часть последовательности ДНК [4], которая отображена в таблице 1. Видно, что подстроки 'ATCAATATGACG' и 'AAAAATACTT' появляются несколько раз в последовательности S .

Таблица 1

Геномная последовательность

```
gATCAATATGACGaatgcaacaataaattcttatgggattcttgctgtaAAAAATACTTcacgtgataaagtcataaccacacgg  
ttggcagctcagaagcttttgccgtttagcgtttctggcgcggcttacAAAAATACTTgtttgatccacgtttgcagatcatttcggaattgga  
aagtggtgctctcATCAATATGACGcct
```

Смысл алгоритма заключается в том, чтобы найти часто повторяющиеся подстроки длиной L в исходной строке, а найденные подстроки затем заменить на любые символы, которые пусть будут $t = \{t_1 \dots t_n\}$, не принадлежащий алфавиту $A = \{A, T, G, C\}$. Где n – это количество найден-

ных подстрок. Затем необходимо запомнить соответствие между символами t и повторяющимися подстроками в словарь T . Далее поэтапно находить подстроки меньшей длины $n - 1$ и вставлять символы за место найденных подстрок. Первоначально данный алгоритм использовался для нахождения подстроки длиной L , которая встречалась максимальное количество раз.

Ниже в таблице 2 представлен псевдокод предлагаемого алгоритма.

Таблица 2

Псевдокод предлагаемого алгоритма

```

wordCount(Text, Word):
    count = 0
    for i = 0 to |Text| - |Word|
        if Text(i, |Word|) = Word
            count = count + 1
    return count
allFrequentWords(Text, l, n):
FrequentWords = []
    for i = 0 to |Text| - l
        Pattern = Text(i, l)
        COUNT[i] = WordCount(Text, Pattern)
    for i = 0 to |Text| - l
        if COUNT[i] > n
            append Text(i, l) into FrequentWords
    remove duplicates in FrequentWords
    return FrequentPatterns

encodeText(Text, L, w, n):
dictionary = [UTF - 8 symbols]
T = []
    rep_word = next symbol in dictionary
    for i = L to w:
        freq_word = allFrequentWords(Text, i, n)
        for word in freq_word:
            while rep_word in 'ATGC':
                rep_word = next symbol in dictionary
            Text = Text.replace(word, rep_word)
            append (rep_word, word) into T
            rep_word = next symbol in dictionary

    return Text, T

```

Разберем подробнее данный алгоритм. На вход функции *wordCount* принимаются две строки: *Text*, *Word*. *Text* представляет собой исходную последовательность, в которой необходимо найти подстроку *Word*. $|Text|$ и $|Word|$ демонстрируют длину соответствующих строк. Подстрока длиной $|Word|$, начинающая с i -ой позиции текста отображена как $Text(i, |Word|)$. На выходе функция возвращает число, сколько раз появилось данное слово в тексте.

Алгоритм поиска *allFrequentWords* последовательно перебирает все возможные подстроки длиной l , которые присутствуют в тексте, и записывает их количество в массив *COUNT*. После

этого выбираются только те подстроки, которые встречались больше n раз.

На вход функция *encodeText* получает геномную последовательность ДНК *Text*, максимальную и минимальную длины искомым подстрок (L, w) , также минимальное количество n появлений подстроки в тексте. Далее от L к w вызывается алгоритм *allFrequentWords* находя частые подстроки и заменяя найденные подстроки на символы. Функция *encode* возвращает закодированный текст исходной последовательности и созданный словарь.

Таким образом, например, заменим подстроки 'ATCAATATGACG', длина которого равна $L = 12$, на символ '/', а найденную фразу записать в словарь. Следующая же часто встречающаяся подстрока наибольшей длины будет 'AAAAATACTT', где $L = 10$. Заменим найденную подстроку на символ '0', полученную подстроку также запишем в словарь. Тогда $A = \{A, T, G, C, /, 0\}$, а словарь соответствий $T = \{ '/': 'ATCAATATGACG', '0': 'AAAAATACTT' \}$. Далее поэтапно кодировать фразы наименьшей длины, а найденные подстроки заносить в словарь. Результатом кодирования полной геномной последовательности [4] до длины $L = 2$ включительно представлен на рис. 1, где слева отображена исходная, а справа закодированная последовательность.

| | |
|---|---|
| <pre>ATGCGTGGTCTGGATTATCGGTGGATTGAAGCTCTAGATAGTGTGGTGA GTAAGGCAGTTTTGAACGAGCGGCTGAGCAGCTTTTATCTCCCAATC TGCTGTATCACAACGGATTAACAGTTAGAAAAATACTTAGCTCAGCCA GTGTTGATCCGTGAGCAGCCACCAAGGCCAACCTTGGTTGGCAAAAAGT TGCTTGGGCTCTATCGCCGAGTCTGTTAATCGAACAAGAGTTGGTTCC GGAGTTGACGAACCAAGAGCACGTTGCCCCGTTTCTATGTCGATTGCG ACCAATGCAGATAGCCTAGCCACTTGGTTGTTACCGGCGCTGGATAAGG TAATGAAAAGCCGCAAGTGAATTGAATCTGGTGATTTATGGTGAGTC ACGTACCCCTCGATAAGCTCAAAAATGGTGAAGTGGTGGTGCATCAGT TTGGAGCCACAACCGATCACTGGGTGTAGTGTGAGTATTTGGGTGAGA TGGAGTATCTGTGCGTGCACAAGCCCTGAGTCTATCAAAAATACTTTGCT AAGGGGGTCACCCCGCTTATTAATCAAAAGCGCCTGCGGTGTCTTACG ATCAATATGACGAATTGCACAATAAATCTTATGGGATTACTTTGCTGTA CCACGTGATAAAGTCATCAACCACACGGTTGGCAGCTCAGAAGCTTTTG TCCGTTTAGCGCTTTCTGGCGCGGCTTACTGTTTGATCCACGTTTGCGAG ATCATTTCCGAATTGGAAAAGTGGTGCTCTCATCAATATGACGCCTGATTT TATGCTGTGATATCCGATTTTTGGCATATTGGCAGCTTGAAACGGGGCG TCTTACTTGAAATTAGCGAAGCGATCACTGCTTATGCGAAAAGCGTACT ACCTCAATAA</pre> | <pre>AQp=qR^?_AS3`G@Tb{U4rTVc{6TB>}d~G0DjdG7 G4CEWC5eXfTgFshTHb}liClvAE JjwYEGZkAc8 C_x5l wC=aTy9CW¥:}hGKA3[zCZAD}Amx;RCy@GuCB1pGS LnMGm VGQCesLF0faM tCiAH`jk&<G/yTiANETO>§ T6C{TX\AxCvG2CA?oTtTPCYgiUlT7J8CnC:;jrc/;K ~fG Aq ^ToG\2]Cg<T]P]G1OC9G~§zTN</pre> |
|---|---|

Рис. 1. Результат кодирования с заменой слов

Для того, чтобы оценить работу алгоритма сжатия воспользуемся коэффициентом сжатия, который описывается следующей формулой [2]:

$$k = \frac{S_c}{S_u}, \quad (1)$$

где S_c – объем сжатых данных,

S_u – объем исходных данных.

Тогда объем исходных данных составляет $S_c = 7176$ бит, а полученный объем сжатых данных $S_u = 1864$ бит. Тогда коэффициент сжатия будет равен $k = 0,25975$. То есть объем сжатых данных в 3.85 меньше объема исходных данных.

Для того, чтобы оценить качество сжатия алгоритма сравним его с другими популярными алгоритмами сжатия, а именно [2,3]:

1. Метод Хаффмана;
2. Арифметический метод;
3. Алгоритм LWZ.

В таблице 3 отображены результаты кодирования сжатия данных исходной последовательности вышеперечисленных алгоритмов.

Таблица 3

Сравнительная таблица результатов четырех алгоритмов

| | Предложенный алгоритм | Метод Хаффмана | Арифметический метод | LZW |
|-------|-----------------------|----------------|----------------------|-------|
| S_u | 7176 | 7176 | 7176 | 7176 |
| S_c | 1864 | 1794 | 32 | 2691 |
| k | 0,25975 | 0,25 | 0,00446 | 0,375 |

Таким образом, результат предложенного алгоритма показал себя чуть хуже, чем алгоритм Хаффмана, но показал сжатие данных лучше, чем кодирование по LZW. При этом арифметический метод показал лучший результат среди всех рассматриваемых алгоритмов сжатия.

Как и алгоритм LZW, предложенный метод является словарным методом кодирования информации. Хотя и предложенный алгоритм показал сжатие данных лучше, чем LZW, стоит отметить и недостатки данного алгоритма, которые касаются в затрачиваемых вычислительных ресурсах при использовании данного алгоритма. Каждый раз при вызове, функция *PatternCount* проверяет на наличие подстроки на i -ой позиции в последовательности. Таким образом происходит $|Text| - L + 1$ проверок. При этом при каждой проверке происходит посимвольное сравнение, которое равно длине L . Только при вызове *PatternCount* общее количество шагов равно $(|Text| - L + 1) * L$. Далее функция *allFrequentWords* должна вызвать функцию *PatternCount* в количестве $|Text| - L + 1$ раз. Последняя функция *encodeText* вызывает *allFrequentWords* до $L - 1$ раз. Таким образом общая сложность алгоритма составляет $O(|Text|^2 L^2)$. При этом сложность алгоритма LZW составляет $O(|Text|)$ [2], которое является преимуществом среди других алгоритмов, так как оно кодирует данные за один проход. В то же время арифметическому методу и алгоритму Хаффмана необходимо два прохода или же $O(|Text|^2)$ [2,3]. Первый проход необходим для нахождения вероятностей появления символов. Второй проход является кодированием информации. Сравнения временную сложность алгоритмов, можно сделать вывод, что предлагаемый метод является неэффективным.

Каскадная схема сжатия данных.

Рассмотрим, как предложенный алгоритм работает вместе с другими алгоритмами, а именно с методом Хаффмана и арифметическим методом. В данном случае не рассматривается работа предложенного метода с алгоритмом LZW, так как оба метода являются словарными. В таблице 4 отображены результаты работы алгоритма с заменой слов с алгоритмом Хаффмана. В второй строке отображена исходный объем S_c , во третьей строке объем сжатых данных S_{u1} по предложенному алгоритму, в четвертой строке объем сжатых данных S_{u2} после применение алгоритма Хаффмана на текст с замененными словами. В пятой строке вычисляется коэффициент сжатия k_s по предложенному методу для всех строк длиной L . В шестой строке вычисляется коэффициент сжатия, который отличается от (1) по следующему принципу:

$$k_q = 100 * \frac{S_{u2}}{S_{u1}}, \quad (2)$$

то есть отношение объема сжатых данных по предложенному алгоритму S_{u1} на объем сжатых данных после применение алгоритма Хаффмана на текст с замененными словами. В столбцах указаны длины замененных слов, то есть 12-6 означает, что в исходной последовательности были заменены все слова длиной $6 \leq L \leq 12$.

Можно заметить, что, заменяя слова длиной до $L = 8$ эффективность при последующем использовании алгоритма Хаффмана падает, сравнивая с результатом кодирования только по алгоритму Хаффмана. Начиная замену слов длиной до $L = 7$ итоговый объем сжатых данных уменьшается, по сравнению с объемом сжатых данных при использовании алгоритмов отдельно. Когда же замещаются фразы до длины $L = 2$ включительно, то качество сжатия ухудшается и объем сжатых данных увеличивается, если сравнивать объем сжатых данных до длины $L = 3$. Объясняется ухудшение качества сжатия тем, что количество уникальных символов в последовательности растет, соответственно, растет длина кодирующего кода. При этом польза от замены подстрок с каждой итерацией уменьшается.

В таблице 5 отображен результат кодирования по алгоритму Хаффмана. В таблице 6 отоб-

ражен результат каскадного кодирования. В левой части таблицы 6 исходные символы {A, T, G, C} кодируются не менее тремя битами, в то же время используя только алгоритм Хаффмана каждый символ кодируется двумя битами. Этим же объясняется ухудшение качества кодирования при замене слов длиной $L = 12$, заменяя только одну подстроку символ 'C' вынужден кодироваться тремя битами.

Таблица 4

Результаты работы алгоритма с алгоритмом Хаффмана

| L | 12 | 12-10 | 12-9 | 12-8 | 12-7 | 12-6 | 12-5 | 12-4 | 12-3 | 12-2 |
|--------------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| S_{σ} , бит | 7176 | 7176 | 7176 | 7176 | 7176 | 7176 | 7176 | 7176 | 7176 | 7176 |
| S_{u1} , бит | 7008 | 6872 | 6616 | 6288 | 5624 | 4064 | 3432 | 2408 | 2200 | 1912 |
| S_{u2} , бит | 1944 | 1918 | 1862 | 1808 | 1716 | 1562 | 1541 | 1472 | 1468 | 1499 |
| k_{σ} , % | 97,66 | 95,76 | 92,20 | 87,63 | 78,37 | 56,63 | 47,83 | 33,56 | 30,66 | 26,64 |
| k_{σ} , % | 27,74 | 27,91 | 28,14 | 28,75 | 30,51 | 38,44 | 44,9 | 61,13 | 66,73 | 78,4 |

Таблица 3

Результат кодирования по алгоритму Хаффмана

| Символ | Частота | Код |
|--------|---------|-----|
| A | 227 | 01 |
| C | 190 | 00 |
| G | 228 | 10 |
| T | 252 | 11 |

Таблица 4

Каскадное кодирование словарным методом и алгоритмом Хаффмана

| $2 \leq L \leq 12$ | | | $L = 12$ | | |
|--------------------|---------|---------|----------|---------|-----|
| Символ | Частота | Код | Символ | Частота | Код |
| A | 27 | 000 | A | 217 | 01 |
| C | 12 | 0010 | G | 224 | 10 |
| G | 15 | 0100 | T | 246 | 11 |
| T | 19 | 1111 | C | 186 | 000 |
| ... | ... | ... | / | 2 | 001 |
| ~ | 2 | 1101000 | | | |

Аналогично, в таблице 7 представлены расчеты предложенного алгоритма с арифметическим алгоритмом.

Длина в битах после сжатия данных по арифметическому кодированию не изменяется и равна 32 бит, несмотря на изменения параметров предлагаемого метода. Таким образом, использовать предлагаемый словарный метод вместе арифметическим методом бессмысленно.

Результаты работы алгоритма с арифметическим алгоритмом

| L | 12 | 12-10 | 12-9 | 12-8 | 12-7 | 12-6 | 12-5 | 12-4 | 12-3 | 12-2 |
|----------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| S_c , бит | 7176 | 7176 | 7176 | 7176 | 7176 | 7176 | 7176 | 7176 | 7176 | 7176 |
| S_{u1} , бит | 7008 | 6872 | 6616 | 6288 | 5624 | 4064 | 3432 | 2408 | 2200 | 1912 |
| S_{u2} , бит | 32 | 32 | 32 | 32 | 32 | 32 | 32 | 32 | 32 | 32 |
| k_s , % | 97,66 | 95,76 | 92,20 | 87,63 | 78,37 | 56,63 | 47,83 | 33,56 | 30,66 | 26,64 |
| k_q , % | 0,457 | 0,466 | 0,484 | 0,509 | 0,569 | 0,787 | 0,932 | 1,329 | 1,454 | 1,674 |

Заключение

Качество сжатия предлагаемого алгоритма практически такая же, как и при использовании кодирования Хаффмана. При этом он показал качество сжатие лучше, чем другой словарный метод LZW. Как было показано, из-за больших расходов на вычислительные ресурсы данный алгоритм не рекомендуется использовать, когда данные используются часто и время работы алгоритма приоритетно над качеством сжатия данных. Данный алгоритм можно использовать только в тех случаях, когда данные изменяются редко или лежат в архиве.

При использовании данного алгоритма с алгоритмом Хаффмана необходимо понимать, что кодирование данных эффективно только тогда, когда необходимо закодировать только часть повторяющихся подстрок, в данном случае при $3 \leq L \leq 12$, при более тщательном кодировании словарным методом приводит к тому, что итоговое качество сжатия данных начнет падать. При использовании данного метода с арифметическим методом, который показал лучшую степень сжатия данных, итоговый объем данных не изменился, поэтому в использовании предложенного метода вместе с арифметическим методом нет нужды.

Литература

1. *Compeau Phillip и Pevzner Pavel*. Bioinformatics Algorithms: An Active Learning Approach. 2015. 2-е. С. 5-15.
2. *Salomon David*. Data Compression: The Complete Reference. London: Springer, 1998. 4-е, 1091 с.
3. *Nelson Mark и Gailly Jean-loup*. The Data Compression Book. [б.м.]: IDG Books Worldwide, 1995. 2-е.
4. *Vibrio cholerae O1 biovar El Tor str. N16961 chromosome I, complete sequence* [Электронный ресурс]. *Heidelberg J.F., Eisen, J.A., Nelson, W.C., Clayton, R.A., Gwinn, M.L., Dodson, R.J., Haft, D.H., Hickey, E.K., Peterson, J.D., Umayam, L.A., Gill, S.R., Nelson, K.E., Read, T.D., Tettelin, H., Richardson, D., Ermolaeva, M.D., Vamathevan, J., Bass, S., Qin, H., Dragoi, I.* // The National Center for Biotechnology Information. - [https://www.ncbi.nlm.nih.gov/nuccore/NC_002505.1?report=fasta&log\\$=seqview&from=512504&to=513400](https://www.ncbi.nlm.nih.gov/nuccore/NC_002505.1?report=fasta&log$=seqview&from=512504&to=513400) (дата обращения: 25.10.2019).
5. *Terry Brown A.* (2006). Genomes 3. 3-d edition. 713 с.

CONSTRUCTION A CASCADE SCHEME OF EFFECTIVE ENCODING OF GENOMIC DATA WITH DELETING DIFFERENT TYPES OF DATA REDUNDANCY

Emil R. Kelimov

MTUCI, student of group M111801(74)

semkel07@gmail.com

Alexey V. Guzeev

MTUCI, PhD, Associated professor of the department of multimedia networks and communication services

a_guzeev@mtuci.ru

Key words: *bioinformatics, dictionary method for coding information, Huffman algorithm, arithmetic coding method, LWZ algorithm, data compression, genomic data.*

This article proposed a dictionary method for coding genomic data by searching for substrings of the greatest length. The article discusses the coding results of the proposed algorithm. Comparison of this method with popular methods of encoding information, such as the Huffman algorithm, the arithmetic method, the LZW algorithm. The article also discusses the results of a cascade coding scheme for genomic data with the proposed method.