

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное
учреждение высшего профессионального образования
«Пермский национальный исследовательский
политехнический университет»

**ИННОВАЦИОННЫЕ ТЕХНОЛОГИИ:
ТЕОРИЯ, ИНСТРУМЕНТЫ, ПРАКТИКА**

Материалы VII Международной
интернет-конференции молодых ученых,
аспирантов, студентов
(01 ноября 2015 г. – 30 ноября 2015 г.)

Издательство
Пермского национального исследовательского
политехнического университета
2016

Представлены работы молодых ученых, аспирантов, а также лучшие работы студентов.

Конференция посвящена вопросам системных исследований и моделирования, информационно-измерительных и управляющих систем, систем телекоммуникации и связи, энергетики и энергоресурсосбережения, информационных технологий и средств автоматизации, аппаратно-программного обеспечения информационно-управляющих систем.

Публикуемые результаты исследований могут быть интересны молодым исследователям, преподавателям и специалистам, интересующимся теоретическими и прикладными разработками в данной предметной области.

Редакционная коллегия:

А.М. Костыгов, доцент, канд. техн. наук;

Б.В. Кавалеров, доцент, д-р техн. наук;

А.В. Кычкин, доцент, канд. техн. наук (отв. редактор);

А.Б. Петроченков, доцент, канд. техн. наук;

Н.М. Труфанова, профессор, д-р техн. наук;

Р.А. Файзрахманов, профессор, д-р экон. наук;

А.Г. Щербинин, профессор, д-р техн. наук;

А.А. Южаков, профессор, д-р техн. наук

Рецензент

д-р техн. наук, профессор, генеральный директор ОАО «НИИУМС» *В.Н. Аликин*

Секция 1

НОВЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И СИСТЕМЫ

ОСНОВНЫЕ МЕТОДЫ РЕАЛИЗАЦИИ КОНЦЕПЦИИ BYOD

Студенты Р.А. Андреев,
П.А. Феофилова, А.С. Бадртдинов,
канд. физ.-мат. наук, доцент Е.Л. Кротова
Пермский национальный исследовательский
политехнический университет

В статье рассмотрены основные методы реализации концепции BYOD, их достоинства и недостатки.

Ключевые слова: BYOD, сетевой подход, MDM, виртуализация.

BASIC METHODS OF BYOD CONCEPT REALIZATION

Students R.A. Andreev, P.A. Feofilova, A.S. Badrtdinov,
Ph.D. in Physics and Mathematical Sciences,
Associate Professor E.L. Krotova
Perm National Research Polytechnic University

This article reviews basic methods of BYOD realization, their advantages and disadvantages.

Keywords: BYOD, network approach, MDM approach, virtualization.

В наши дни мобильные устройства, такие как смартфоны, планшетные компьютеры и ноутбуки, стали очень популярными. Практически у каждого человека сейчас имеется хотя бы одно такое устройство. В связи с этим многие организации начинают задумываться о реализации концепции BYOD либо уже ее применяют. Bring your own device («Принеси свое собственное устройство») означает, что сотрудники организации используют свою личную технику на рабочем месте. Отказ от запрета мобильных устройств в корпоративной среде должен сопровождаться применением гибких сбалансированных методов, решающих возникающие задачи. Существует несколько подходов к решению данной проблемы.

Ключевой идеей сетевого подхода к решению проблем BYOD является технология управления доступом к сети (NAC). Система NAC контролирует, как и какие устройства получают доступ к сети. Различные решения этого метода представляются компаниями Cisco Systems и Meru Networks [1].

Концепция BYOD, созданная с помощью сетей Cisco, реализована на базе неограниченной архитектуры сети Cisco, что позволяет

уполномоченным лицам с помощью любого устройства в любом месте или в любое время подключиться к сети своей организации. В целях удовлетворения требованиям соединения от различных устройств были разделены точки подключения к проводной, беспроводной локальной сети (LAN), беспроводной глобальной сети (WAN). Этот подход помогает охватывать и поддерживать большое разнообразие устройств, а также обеспечивать более безопасное подключение к сети. При подключении к сети любое новое устройство оставляет метку, что в дальнейшем может быть использовано для подключения к устройству без необходимости повторной идентификации. Также были включены централизованные системы безопасности для обнаружения устройств и пользователей и применения различных политик и правил для каждого пользователя. Система управления сетью отвечает за управление сетью и другими функциями производительности. При подключении к сети пользователи изначально проходят через стадию аутентификации, при которой сначала оценивается состояние их устройств, затем создаются профили устройств, применяются политика и правила до того, как предоставляется разрешение. Управление мобильным устройством системы используется для восстановления подсчета/регистрации клиентов, которые имеют доступ к сети, ресурсов, к которым они получили доступ, и сессий подключения к сети. Также обеспечивается централизованное управление мобильными устройствами.

Сети Меру делают больший акцент на портативные мобильные устройства, которые сотрудники присоединяют к корпоративной сети. Для того чтобы осуществить безопасное и эффективное применение BYOD, были разработаны руководящие принципы для управления сетевой структурой. Когда мобильные устройства подключаются к сети, они должны пройти через процессы аутентификации для получения доступа. Идея заключается в том, чтобы пользователи проходили аутентификацию с помощью приложения с минимальной помощью IT-отдела. Для того чтобы управлять различными устройствами в сети, используются разнообразные сервисы и ресурсы, такие как шифрование устройства, модернизация т.д. Кроме того, система может назначить пользователям или сотрудникам права и привилегии в зависимости от их роли и поместить их в правой среде сетевой системы. Система BYOD Меру была разработана только для устройств с поддержкой беспроводных ЛВС в организации. Система обеспечивает возможность

интеллектуального и гостевого соединения для доступа на основе идентификационной информации, управления регистрацией и политикой устройства для корпоративных и частных мобильных устройств (Meru Networks, 2012). Возможности интеллектуального и гостевого соединения предоставляют безопасные соединения путем упрощения доступа по стандарту 802.1x и раздачи WiFi-соединений с использованием политик и правил. Новые пользователи могут также легко создать свой собственный профиль WiFi через веб-портал и ввести свой логин и пароль, чтобы получить доступ к сети.

Преимуществом сетевого подхода к BYOD является возможность безопасного и контролируемого доступа к сети и приложениям для широкого спектра совместимых устройств независимо от местоположения пользователя. Политики сетевого доступа могут быть обеспечены для того, чтобы устройства осуществляли дифференциацию и управление доступом к информационным ресурсам. Аутентификация и шифрование могут быть использованы для защиты сетей. Виртуальные локальные сети могут также использоваться, чтобы держать устройства на отдельных сегментах для уменьшения сетевого трафика. Инструменты управления мобильными устройствами могут быть интегрированы с сетевыми инфраструктурами, чтобы обеспечить видимость по всей сети, а также полное управление ресурсами, услугами и пользователями.

Также существуют недостатки сетевого подхода, относящиеся к совместимости различных устройств с различными операционными системами в сети. Кроме того, назначение сетевой службы и доступа к сетевым ресурсам на нужном устройстве может быть проблемой, особенно когда один пользователь имеет несколько устройств в сети. Существует возможность подделки личных данных, особенно если политика требует, чтобы приложения для мобильных устройств и файлов были открытыми и пригодными для работы в системе. Мониторинг и регистрация устройств могут быть спутаны и изменены, или вредоносные устройства могут быть незамеченными. Зараженные устройства, подключающиеся через виртуальную частную сеть, могут поставить под угрозу присоединенную сеть. Точно так же инфицированное устройство на VLAN может заразить все другие устройства в той же сети, что делает их бот-сетями во время DDOS-атак. Кроме того, доступ к мультимедийному контенту может перегрузить сеть, а также проблемой могут стать управление обширным содержанием средств массовой информации и их отслеживание.

Приложения MDM (Mobile Device Management – управление мобильными устройствами) – это программное обеспечение для работы с корпоративными системами при помощи мобильных устройств [2].

Преимущество использования MDM заключается в том, что он обеспечивает единый подход к управлению мобильными устройствами. Политика доступа может быть общей для всех устройств, соединенных через платформы MDM. Программа-конфигуратор используются для того, чтобы обеспечить видимость подключенных устройств. Серверы MDM могут использоваться для отправки профилей конфигурации, таких как пользовательская политика, удаленное стирание данных, удаленная блокировка и удаление пароля устройства. Камера, приложения и использование облачных сервисов могут быть ограничены на устройствах, которые используют системы управления приложениями.

Недостатки подхода MDM: централизованный подход, который используется для управления всеми мобильными устройствами, включая рабочий стол и серверы, может быть очень запутанным, так как используются те же или сопутствующие инструменты, особенно при возникновении проблем совместимости устройств. В данный момент поддерживается ограниченное количество устройств и операционных систем. Корпоративные данные могут также путаться с персональными данными, так как они используются на той же платформе. Вредоносное программное обеспечение может легко быть передано корпоративным сетям, потому что нет никаких средств управления, чтобы остановить устройства, запускающие помещенные в черный список (запрещенные) приложения. Нет никакой возможности для полного доступа к устройствам пользователя и их контроля, чтобы диагностировать проблемы. Владельцы устройств не будут довольны возможностью удаленного стирания данных с их мобильных устройств без их на то согласия.

Виртуализация может быть также довольно эффективным методом управления корпоративными данными в средах BYOD. Она позволяет сотрудникам безопасно использовать их рабочие столы и приложения путем аутентификации с безопасным паролем. В случае утери или кражи мобильного устройства удаленное соединение доступа прекращается, для того чтобы защитить данные [3].

Виртуализация позволяет приложениям и рабочим столам работать на серверах вместо мобильных устройств. Это обеспечивает

безопасный удаленный доступ к ресурсам, не оставляя корпоративных приложений и данных по устройствам пользователей, таким образом разделяя корпоративные и персональные данные. Виртуализированные приложения предлагают больше функциональности, чем мобильные приложения, потому что системные ресурсы не ограничивают их. Приложения могут легко быть обновлены на устройствах, используя централизованный подход. Вредоносное программное обеспечение и вирусы не могут быть отправлены через виртуализированные приложения.

Однако не все мобильные устройства поддерживают системные требования или соответствуют им для запуска виртуальных приложений и рабочих столов. Соединение мобильных устройств с удаленными рабочими столами может подвергнуть корпоративную сеть внешним угрозам. Совместимость и поддержка мобильных устройств для доступа к удаленным рабочим столам могут быть сложной задачей для ИТ. Приложения, которые подразумевают работу с мышью и клавиатурой, будет трудно использовать. Загрузка виртуализированного приложения обычно занимает больше времени, чем загрузка мобильного приложения. Будет трудно вывести на маленький экран приложения, совместимые с большими экранами. Отсутствие доступа в Интернет означает отсутствие возможности работать или получить доступ удаленно.

Из-за недостатков каждого подхода большинство организаций используют эти подходы вместе, чтобы постоянно обеспечивать безопасность BYOD. Стремительно возрастает интеграция инструментов управления мобильными устройствами в инфраструктуру сети для автоматизации безопасного ввода. Телефонно-ориентированные подходы используют технологию контейнеризации, чтобы ограничить пользовательскую возможность получить доступ, копирование, вставку или редактирование данных, сохраненных в контейнерах приложения. Такой подход используется в сочетании с подходами виртуализации, чтобы обеспечить удаленный доступ к контейнерам. Формальная рукописная политика BYOD проводится некоторыми организациями, чтобы работать вместе с MDM и сетевыми подходами для снижения риска потенциальной потери данных. Другие организации, например, работающие в отраслях защиты, выбирают экстремальные методы запрещения всех персональных устройств в их корпоративных сетях, тогда как большинство организа-

ций стремится принять средства управления BYOD, что предлагает некоторую гибкость пользователям при осуществлении корпоративной политики. Данный подход является новым понятием, которое еще не было широко исследовано и опубликовано.

Библиографический список

1. Meru Networks. BYOD best practices, requirements and challenges. – 2012. – URL: www.merunetworks.com/collateral/white-papers/2012-wp-byod-implementation-whitepaper-for-wlan-security.pdf (дата обращения: 12.10.2015).
2. Rhee K., Jeon W., Won D. Security requirements of a mobile device management system // International Journal of Security and Its Applications. – 2012. – Vol. 6. – № 2. – P. 353–358.
3. Analysis of security controls for BYOD (bring your own device) / D. Rivera, G. George, P. Peter, S. Muralidharan, S. Khanum. – Melbourne, Australia: The University of Melbourne, 2013.
4. Scarfo A. New security perspectives around BYOD // Seventh International Conference on Broadband, Wireless Computing, Communication and Applications. – Victoria, BC. – 2012.

РАЗРАБОТКА ОНТОЛОГИЧЕСКОЙ МОДЕЛИ ТЕХНОЛОГИЧЕСКОГО ГРАФА ЗАДАЧИ КАЛЕНДАРНОГО ПЛАНИРОВАНИЯ

**Магистранты Д.И. Ахметзянов, И.А. Посохин,
д-р экон. наук, профессор Р.А. Файзрахманов**
Пермский национальный исследовательский
политехнический университет

В статье кратко описан процесс построения онтологической модели технологического графа задачи календарного планирования. Разработанная онтологическая модель необходима для проектирования БД АРМа «ГДО-планировщик».

Ключевые слова: моделирование, онтологическая модель, технологический граф, календарное планирование.

DEVELOPMENT OF AN ONTOLOGICAL MODEL OF TECHNOLOGICAL GRAPH OF THE SCHEDULING PROBLEM

**Master Students D.I. Akhmetzyanov, I.A. Posohin,
Doctor of Economic Sciences, Professor R.A. Fayzrahmanov**
Perm National Research Polytechnic University

This article briefly describes the process of constructing the ontological model of technological graph of the scheduling problem. Designed an ontological model is needed to design the DB ArmA "PDO-scheduler".

Keywords: modeling, ontological model, process graph, scheduling.

Задача календарного планирования производства (далее – задача КП производства) для любого производственного предприятия является одной из важнейших составляющих всего производственного процесса, так как описывает работу всех звеньев производства по изготовлению и выпуску заданной номенклатуры изделий в установленных объемах и сроках при наилучшем использовании всех производственных ресурсов. Для решения задачи КП инженеру-технологу, как правило, первым делом необходимо спроектировать технологический состав производимого изделия.

Проектирование технологической спецификации (далее – ТС на производимое изделие – процесс, включающий в себя комплекс определенных работ: конструкторских, технологических, нормирую-

щих и других, которые должен выполнить инженер-технолог в процессе формирования технологического состава. В производственных условиях пакет проектных документов под общим названием «Технологическая спецификация» разрабатывается при технологической подготовке производственного процесса, при этом наследуя всю конструкторскую, и является комплектом документации для запуска изделия в производство. Одним из разделов, составляющих ТС, является «Технологический граф». Подобное технологическое описание изделия является важнейшим источником информации при формировании календарного плана производственного процесса. Данный документ необходим при конфекционировании логики расцеховки (маршрута изготовления), выборе методов технологической обработки входимости и моделирования проектируемого изделия в целом.

Содержательная часть существующей логики формирования маршрута изготовления при наличии некоторого общего подхода на практике является процессом относительно уникальным, исходя из квалификации технолога (экспертного мнения) и требований спецификации, а значит, имеет субъективный характер, и использование ее для целей производственной системы класса САПР не представляется возможным.

В действующих САПР, которые предназначены для проектирования ТС, вопросам разработки и формирования технологического графа не уделяется должного внимания, тогда как эти данные являются исходными и неотъемлемыми при формировании КП производственного процесса. Например, в САПР: PLM «Лоцман-Расцеховщик» [1] российской компании «АСКОН» ТС наследуются из родительского (конструкторского), а его редактирование и последующая расцеховка составляются специалистом-экспертом (инженером-технологом) путем непосредственного внесения необходимой информации, которой в данном случае должен обладать пользователь, в соответствующее окно интерфейса технологического описания с последующим сохранением в базе данных. Данная форма представления не удовлетворяет требованиям автоматизированного процесса проектирования ТС, так как происходит односторонний обмен информацией от пользователя к программе, в процессах такого вида не используются элементы интеллектуализации.

Следовательно, актуальной становится задача получения и систематизации знаний, составляющих комплексное описание ТС,

структурирования и представления их в адаптированном виде для информационных технологий на основе интеллектуальных технологий и процессов интеграции [2].

Цель исследования – интеллектуализация процесса формирования описания онтологии ТС, разработка технологического графа производимого изделия, на основе которого будет спроектирована БД АРМа «ПДО-Планировщик».

Методы исследования: методы системного анализа, онтологический подход, системный подход, методы интеллектуализации, поддержки принятия решений, средства и методы интерфейса пользователя.

Научная новизна исследования заключается в разработке онтологии ТС применительно к задачам САПР и создании предпосылок для дальнейшей интеграции ОПИ в информационную систему и САПР в целом.

Практическая значимость исследования заключается в создании качественно нового источника информации для реализации процессов принятия инженерных решений при технологической подготовке производства. Анализ действующих САПР показал, что отсутствует общая база данных элементов описания технологического состава. Существующие ТС не имеют общей формы представления, описания разделов разрознены, не имеют единой логики и последовательности изложения и зависят от мнения экспертной оценки.

Следовательно, необходимо на первоначальном этапе исследования выполнить перенос знаний из различных источников в общую базу знаний информационной системы, называемую экспертной системой (ЭС), в данном случае ЭС «ПДО-Планировщик». Данная система способствует обоснованию полученных результатов, предоставляет технологу необходимую информацию и позволяет уверенно без страха ошибки принять правильное инженерное решение. Для завершения домашнего этапа проектирования интеллектуальной системы необходимо структурировать и систематизировать исходные знания. Данные предметной области «Технологический граф» в разделе «Технологическая спецификация» являются трудноформализуемыми, тесно взаимосвязаны с проектными характеристиками изделия (конструкторскими и технологическими параметрами, характеристиками свойств логики расщеховки), требуют постоянного обновления. Только определенная организация системы знаний способна обеспечить ее легкое определение и актуализацию.

Реализация данного процесса в рамках интегрированной системы автоматизированного проектирования технологического состава возможна на основе онтологического подхода.

Разработанная и использованная онтологии находятся среди новых информационных технологий, назначение которой можно определить как поддержку процессов совместного решения задач, т.е. возможность обеспечить одинаковое понимание всеми пользователями смысла применяемых при решении терминов, их атрибутов и отношений между ними [3] в рамках модели.

Итак, необходимо создать онтологическую модель технологического графа, хранящую и систематизирующую базу знаний (информацию о предметной области). Для реализации задачи следует использовать одну онтологию, так как исходные данные и знания тесно взаимосвязаны.

Для определения информации технологического графа был изучен перечень производственной информации, предложенный АО «Пермский завод «Машиностроитель»». Анализ полученной информации выявил определенные закономерности:

- технологический состав в процентном соотношении на 80–85 % схож с конструкторским составом;
- логика формирования расцеховки напрямую зависит от источника поступления входимости (детали): собственного изготовления, покупная или кооперативная;
- в зависимости от признака входимости наблюдаются схожие элементы маршрута, например, операция «сборка» присуща для каждой входимости, вне зависимости от источника поступления.

Для разработки онтологии использовалась технология METHONTOLOGY [4]. Разработанная онтология используется для проектирования вышеупомянутой информационной системы «ПДО-планировщик». Согласно данной методологии проектирование онтологии состояло из двух основных этапов:

- общая концептуализация;
- концептуализация для каждого классификационного дерева.

Общая концептуализация. На этом этапе были сформированы:

- глоссарий терминов;
- деревья классификации концептов.

Фрагмент глоссария терминов представлен в табл. 1.

Таблица 1

Фрагмент глоссария терминов

Термин	Описание термина
Конструкторская спецификация	Комплект конструкторских документов, который включает в себя ряд данных в графической и текстовой форме, описывающий логику формирования конструкторского состава
Конструкторский состав	Иерархический список, представленный в виде дерева или таблицы, построенный по определенной логике (позиция, количество), исходя из требований конструкторской спецификации
Конструкторская входимость	Единица конструкторского состава, имеющая определенные конструкторские характеристики
Технологическая спецификация	Комплект технологических документов, который описывает операции и требования по ним для производства какого-либо узла, полуфабриката, либо изделия в целом
...	...

Анализ глоссария выявил два основных концепта:

- технологический состав (ТС);
- объект предметной области.

«Технологическим составом» будем называть иерархический состав производимого изделия, наследованный из конструкторского состава с дополнительными технологическими параметрами. «Объектом предметной области» будем называть некоторое описание явления, события, процесса реального мира, именно эту информацию формирует пользователь при проектировании технологического графа. Фрагмент деревьев классификации концептов представлен в табл. 2.

Таблица 2

Фрагмент дерева классификации объектов предметной области

№	Название класса	Родительский класс
1	Конструкторская спецификация	–
2	Конструкторский состав	1
3	Конструкторская входимость	2,5
4	Технологическая спецификация	–
...

Отметим, что для построения дерева классификации технологического графа использовались дерево классификации объектов, а также таблица бинарных отношений для объектов предметной области (табл. 3).

Таблица 3

Фрагмент дерева классификации для технологического графа

№	Название класса	Родительский класс
1	Вершина «Конструкторская спецификация»	–
2	Вершина «Конструкторский состав»	1
3	Вершина «Конструкторская входимость»	2,5
4	Вершина «Технологическая спецификация»	–
...

Концептуализация для каждого классификационного дерева. Данный этап включает в себя разработку: словаря концептов; таблицы бинарных отношений; таблицы атрибутов класса; таблицы констант; таблицы формул; таблицы экземпляров.

При этом на втором этапе фактически мы получаем свою онтологию для каждого дерева классификации из этапа 1.

Рассмотрим элементы онтологии применительно к нашей задаче построения технологического графа.

Словарь концептов. Словарь концептов идентичен с терминами, описанными в глоссарии.

Таблица бинарных отношений. Основным отношением, на наш взгляд, является отношение «наследование». В результате анализа этих взаимоотношений появляется возможность автоматизировать реализацию вершин вида:

- «следующая» и «предыдущая» (линейная структура);
- «вершины одного уровня»;
- «вершина-родитель» и «вершина-дочерний элемент».

Таблица 4

Фрагмент бинарных отношений

Участник 1	Название отношения (прямое)	Название отношения (инверсное)	Тип	Участник 2
Технологическая спецификация	Формирует	Формируется	п,п	Технологический состав
Технологический состав	Формируется	Формирует	п,п	Технологическая спецификация
Конструкторская входимость	Входит в	Состоит из	п,п	Технологический состав
Технологический состав	Состоит из	Входит в	п,п	Конструкторская входимость
...

Таблицы атрибутов класса. Пример таблицы атрибутов класса для объекта предметной области «Технологический состав» представлен в табл. 5.

Таблица 5

Фрагмент атрибутов класса «Технологический состав»

№	Наименование атрибута
1	Шифр
2	Наименование
3	ФИО инженера (автор)
4	Источник поступления
...	...

Таблицы констант. Имеет смысл объединить таблицу констант «Станки» с таблицей констант «Цеха» и «Техоперации», так как всегда определенный станок находится в конкретном цехе и может выполнить только специфичные для него техоперации. Фрагмент констант представлен в табл. 6.

Таблица 6

Фрагмент констант

Наименование	Значение	Принадлежность
Станок – 111.1	Механообработка	178
Станок – 111.1	Резка листов	178
Станок – 211.1	Маркировка	215
Станок – 3.111	Отжиг	314

Очевидно, что при данной структуре возможны только текстовые константы.

Таблицы формул. Формулы при проектировании технологического графа в логику проектирования ТС не закладываются, так как на первоначальном этапе построения календарного плана мы учитываем идеальную ситуацию загрузки станочного парка (все свободно) и считаем общее валовое время всего маршрута.

Таблица экземпляров. Пример экземпляра класса представлен в табл. 7 на примере тоже объекта предметной области «Технологический состав». Важно подметить, что при определенных условиях значения атрибутов экземпляра класса могут иметь отличающиеся значения.

Фрагмент атрибутов класса «Конструкторский состав»

№	Экземпляр класса	Атрибут	Значение атрибута
1	Технологический состав	Шифр	2К.1234.0000000
2	Технологический состав	Наименование	Корпус изделия
3	Технологический состав	ФИО инженера (автор)	Иванов И. И.
4	Технологический состав	Источник поступления	Изготавливаемое
...

В результате был получен технологический граф производимого изделия, представленный на рисунке. Для каждой вершина графа – объекта предметной области построены и полностью заполнены описанные выше таблицы.

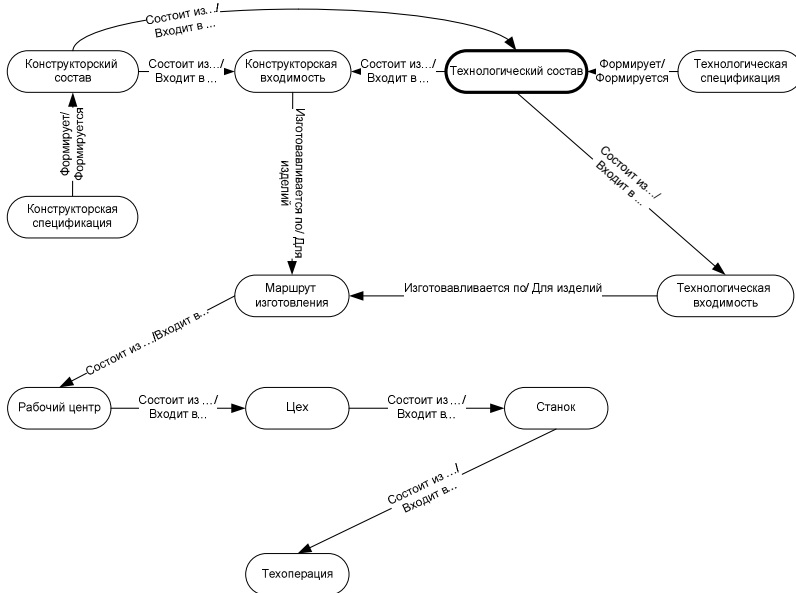


Рис. Технологический граф производимого изделия

Выводы. Произведены структуризация и формализация данных предметной области «Технологический граф» на основе онтологического подхода, что позволяет получать описание проектируемого изделия в наиболее полном объеме, в логически обоснованной последовательности, с использованием единого тезауруса.

Созданная база знаний проблемной области «Технологический граф» является общим фундаментом при проектировании БД для разработки вышеупомянутой информационной системы «ПДО-планировщик». Данная АРМ должна будет формировать правильное инженерное решение при построении КП производственного процесса на изделие или его входимость по требованию. Важно отметить, что дальнейшее усложнение информационной системы (добавление различного математического аппарата при поиске лучшего маршрута) ведет к усложнению разработанной онтологии.

Библиографический список

1. Официальный сайт ПО «Лоцман: PLM Расцеховщик» [Электронный ресурс]. – URL: <http://machinery.ascon.ru/software/tasks/items/?prcid=8&prpid=892> (дата обращения: 02.10.2015).

2. Формализация описания проектируемых изделий на основе онтологического подхода / О.В. Панюшкина, Л.А. Королева, А.В. Подшивалова, К.О. Шевчук // Фундаментальные исследования. – 2013. – № 10.

3. Гаврилова Т.А. Использование онтологий в системах управления знаниями [Электронный ресурс]. – URL: http://www.big.spb.ru/publications/bigspb/km/use_ontology_in_suz.shtml (дата обращения: 25.10.2013).

4. Моделирование и экспертные системы [Электронный ресурс]. – URL: http://sdo.uspi.ru/mathem&inform/lek5/lek_5.htm (дата обращения: 22.10.2013).

5. Гаврилова Т.И., Хорошевский В.Ф. Базы знаний интеллектуальных систем. – СПб.: Питер, 2001.

АРХИТЕКТУРА ПОДСИСТЕМЫ ФОРМИРОВАНИЯ МАРШРУТА РЕЖУЩЕГО ИНСТРУМЕНТА ПРОГРАММНОГО КОМПЛЕКСА РАСКРОЯ «ITAS NESTING»

Студент А.В. Бурылов, ассистент В.С. Шилов
Пермский национальный исследовательский
политехнический университет

Рассматривается подсистема формирования маршрута режущего инструмента. Описаны функции данной подсистемы. Применение принципов и паттернов объектно-ориентированного проектирования позволило разработать простую и гибкую архитектуру для данной подсистемы.

Ключевые слова: гибкая архитектура, паттерны проектирования, принцип инверсии зависимостей, маршрут режущего инструмента.

ARCHITECTURE OF THE CUTTING TOOL PATH GENERATION SUBSYSTEM OF THE PROGRAM NESTING SYSTEM «ITAS NESTING»

Student A.V. Burylov, assistant V.S. Shilov
Perm national research polytechnic university

Article deals with cutter path generation subsystem. Functions of this subsystem are described. Usage of object-oriented design principles and patterns allowed to developed simple and flexible architecture of this subsystem.

Keywords: flexible architecture, design patterns, dependencies inversion principle, cutting tool path.

Автоматизация формирования маршрута режущего инструмента (РИ) требует использования множества разнообразных алгоритмов: эвристик, методов вычислительной геометрии, алгоритмов на графах и т.д. [1]. При проектировании соответствующей подсистемы необходимо обеспечить возможность легкого добавления новых методов построения пути, технологических ограничений, станков и т.д. Следовательно, необходимо создать гибкую архитектуру, позволяющую быстро и независимо изменять различные части подсистемы для реализации нового функционала.

Накопленный опыт использования объектно-ориентированной парадигмы позволил разработчикам определить наиболее удачные подходы, позволяющие создавать системы с гибкой, легко расширяемой архитектурой. К самым фундаментальным из них относятся

паттерны проектирования по GoF («Gang Of Four») [2] и группа принципов проектирования классов SOLID [3]. Выбор подходов при создании архитектуры не всегда очевиден, так как многие из них имеют как достоинства, так и недостатки.

Цель данной работы заключается в разработке архитектуры подсистемы формирования маршрута режущего инструмента с использованием принципов и паттернов объектно-ориентированного проектирования для программного комплекса автоматизированного раскроя материала Itas Nesting [4].

Функции и место подсистемы в комплексе раскроя. Проектируемая подсистема должна предоставить возможность автоматического и интерактивного формирования маршрута режущего инструмента для станков различных типов при использовании стандартной технологии резки [1].

Входными данными для подсистемы являются: карты раскроя, содержащие информацию о контурах деталей и их положении на листовом материале [5]; параметры оборудования (описаны далее); способ резки материала (лазерная резка, резка струей воды, плазменная резка и т.д.).

В результате работы подсистемы будет сформирован маршрут РИ, задающий порядок обработки контуров и определяющий координаты точек входа и выхода инструмента. На основе данной информации другими подсистемами Itas Nesting может быть выполнена визуальная имитация резки и сгенерирована управляющая программа для определенного станка [4].

Создание архитектуры подсистемы. Архитектура подсистемы должна легко расширяться по следующим направлениям:

- добавление новых типов станков и учет дополнительных параметров оборудования при формировании маршрута РИ;
- добавление новых стандартных способов формирования маршрута РИ в автоматическом режиме;
- изменение алгоритмов оптимизации маршрута, добавление новых критериев и технологических ограничений.

При создании архитектуры подсистемы предполагается повторно использовать некоторые существующие в Itas Nesting классы и интерфейсы.

Поскольку полная диаграмма классов является достаточно большой, она будет представлена частями с разбиением классов по

пакетам. На диаграммах будут отображены только наиболее важные поля и методы.

Пакет machine. Данный пакет включает в себя набор классов для моделирования режущих станков. Диаграмма классов представлена на рис. 1. На диаграмме не представлены геттеры/сеттеры для уменьшения занимаемого пространства.

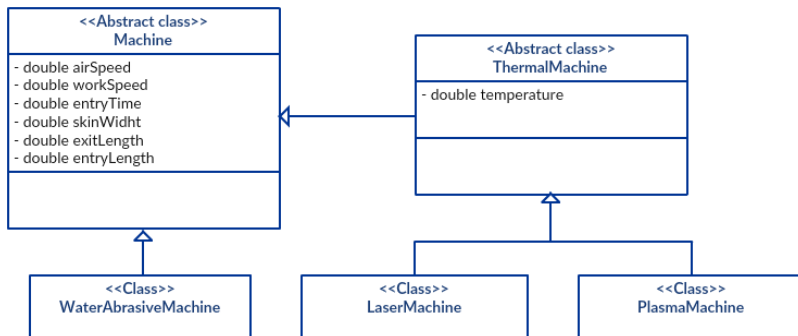


Рис. 1. Диаграмма классов пакета machine

Абстрактный класс Machine имеет следующие поля: скорость холостого хода airSpeed, скорость рабочего хода workSpeed, время врезки entryTime, ширина реза cuttingWidht, длина вреза инструмента entryLength, длина выхода инструмента exitLength. Для станков тепловой резки (ThermalMachine) учитывается температура реза temperature. Явное выделение классов LaserMachine и PlasmaMachine связано с тем, что для станков данных типов процедура формирования маршрута РИ будет отличаться. В дальнейшем с высокой вероятностью у данных классов появятся собственные поля, учитываемые при формировании маршрута РИ.

Пакет path. Содержит основные классы подсистемы: класс маршрута режущего инструмента Path, абстрактный класс генератора маршрута PathGenerator и интерфейс ExtractionStrategy, имеющий метод extract для извлечения множеств потенциальных точек врезки из контуров. На рис. 2 представлена диаграмма классов, непосредственно связанных с Path. Это классы EntryPoint и ExitPoint, представляющие собой точки входа и выхода соответственно. Оба класса наследуют абстрактный класс MoveablePoint, хранящий ссылку на интерфейс MoveProcessor. В данном случае использован паттерн «Стратегия», что

позволяет динамически изменять способ перемещения точек входа и выхода. На диаграмме представлен только SegmentMoveProcessor, управляющий перемещением по отрезку.

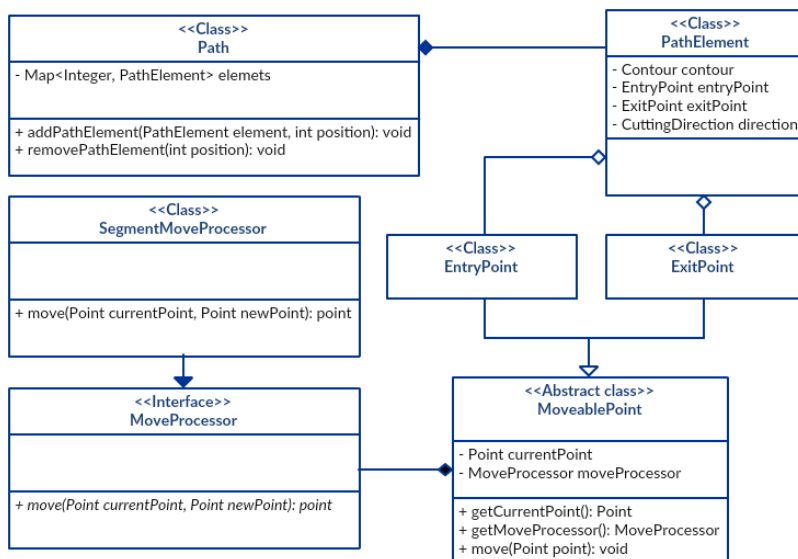


Рис. 2. Диаграмма классов, непосредственно связанных с Path

На рис. 3 представлены классы, формирующие маршрут режущего инструмента (Path). Наследники абстрактного класса PathGenerator могут создавать маршрут РИ в зависимости от способа разбиения контуров на потенциальные точки врезки. На диаграмме представлен только один наследник – класс GDAPathGenerator, строящий маршрут согласно алгоритму, представленному в работе [1].

Представленный на диаграмме класс Contour используется в различных подсистемах комплекса Itas Nesting. Экземпляры класса Contour представляют контур детали и хранят коллекцию геометрических примитивов. Обработка примитивов должна производиться в классах, реализующих интерфейс ExtractionStrategy при вызове метода extract. Результатом будет являться объект класса DiscreteContour, не представленный на диаграмме. Он содержит коллекцию потенциальных точек врезки, т.е. точек, на которых можно выполнить врезку для станка данного типа.

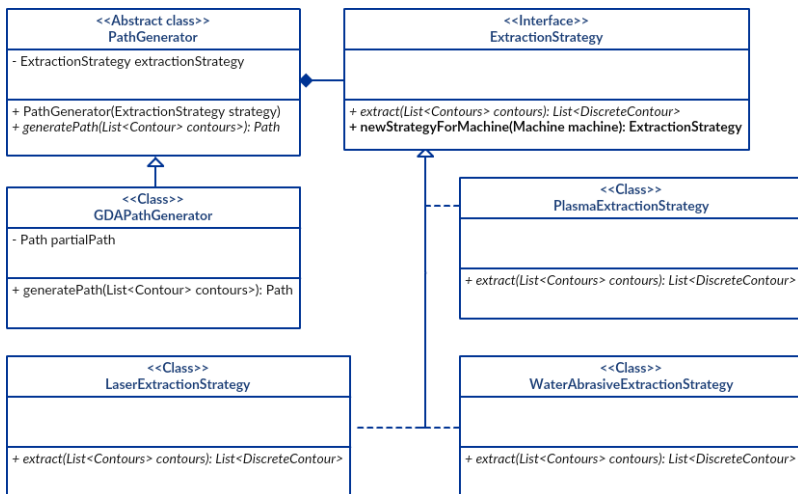


Рис. 3. Набор классов для формирования маршрута РИ

Благодаря применению паттерна проектирования «Стратегия», на диаграмме (см. рис. 3) соблюдается принцип инверсии зависимостей [3], важный для поддержания гибкой объектно-ориентированной архитектуры.

Статический метод `newStrategyForMachine(Machine type)` представляет собой идиому программирования, называемую простой фабрикой. Его цель – полиморфное создание способа разбиения контуров на потенциальные точки для станка заданного типа.

Пакет controllers. В Itas Nesting визуальная работа с деталями на карте раскроя выполняется в терминах режимов: режим поворота, режим перемещения и т.д. Аналогичный подход предлагается использовать и в проектируемой подсистеме. Для этого необходимо создать классы, реализующие интерфейсы `EditorController` и `ModeController` (рис. 4).

Класс `PathGeneratingController`, реализующий интерфейс `EditorController`, – главный управляющий класс подсистемы формирования маршрута режущего инструмента. Задача данного класса – делегировать события мыши классам `AutomaticPathModelController` и `InteractivePathModeController`, представляющим собой автоматический и интерактивный режимы формирования маршрута.

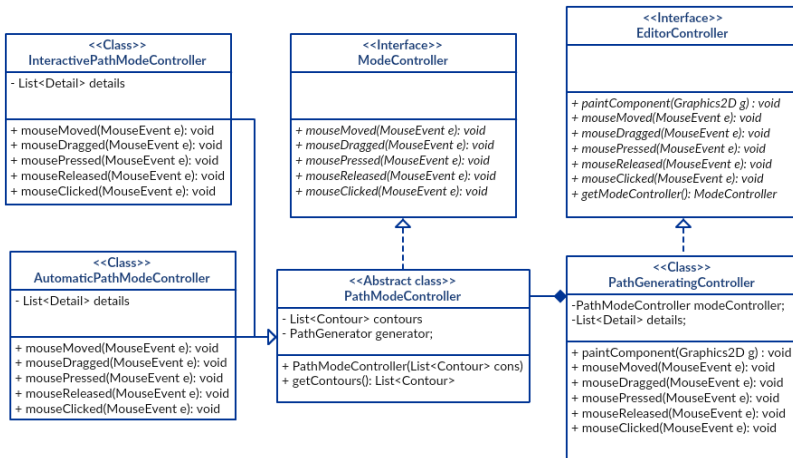


Рис. 4. Диаграмма классов пакета controllers

Запуск процедуры формирования маршрута режущего инструмента выполняется в выбранном режиме при помощи экземпляра класса PathGenerator вызовом метода generatePath(contours).

Разработка интерфейса пользователя не является частью данной работы. Класс, реализующий интерфейс пользователя, должен будет иметь ссылку на PathGeneratingController и делегировать ему события, возникающие при взаимодействии с UI.

Закключение. Таким образом, разработана объектно-ориентированная архитектура подсистемы формирования маршрута режущего инструмента, имеющая следующие свойства:

- каждый класс представляет небольшую сущность и имеет только одну обязанность;
- соблюдается принцип инверсии зависимости во всех частях подсистемы благодаря использованию паттерна «Стратегия»;
- соблюдается принцип открытости-закрытости – для добавления нового функционала не нужно менять поведение существующих классов, достаточно применить наследование и использовать новый класс;
- потенциально изменяемые аспекты архитектуры изолированы в отдельные классы, для их связи с неизменяемыми частями подсистемы использованы принцип слабой связности и делегирование.

На основе созданной архитектуры реализована подсистема формирования маршрута режущего инструмента для программного комплекса раскроя Itas Nesting.

Библиографический список

1. Мурзакаев Р.Т., Шилов В.С., Бурылов А.В. Применение метаэвристических алгоритмов для минимизации длины холостого хода режущего инструмента // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления – 2015. – № 14. – С. 123–136.

2. Приемы объектно-ориентированного проектирования. Паттерны проектирования / Э. Гамма, Р. Хелм, Р. Джонсон, Д. Влиссидес. – СПб.: Питер, 2015. – 368 с.

3. Быстрая разработка программ: принципы, примеры, практика / Р.С. Мартин, Д.В. Ньюкирк, Р.С. Косс; пер. с англ. А.П. Сергеева, Т.А. Шамренко. – М.: Вильямс, 2004. – 752 с.

4. Мурзакаев Р.Т., Шилов В.С., Брюханова А.А. Программный комплекс фигурного раскроя материала Itas Nesting // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2015. – № 13. – С. 15–25.

5. Мезенцев А.С., Шилов В.С. Интерпретация данных о деталях для алгоритмов решения задач двумерного раскроя-упаковки // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2013. – № 8. – С. 137–143.

ИМИТАЦИЯ ДЕЙСТВИЯ СИЛЫ ТЯЖЕСТИ НА ПЛОСКИЕ КОНТУРЫ

Студент А.А. Ветчанинов,
канд. техн. наук, доцент Р.Т. Мурзакаев
Пермский национальный исследовательский
политехнический университет

Рассмотрена имитация действия силы тяжести на плоские контуры, описаны алгоритм нахождения вершины детали, вокруг которой будет происходить поворот под собственным весом, и алгоритм нахождения угла поворота.

Ключевые слова: сила тяжести, центр масс.

SIMULATION OF GRAVITY ON THE FLAT CONTURS

Student A.A. Vetchaninov,
Ph.D. in Technical Sciences, Associate Professor R.T. Murzakaev
Perm National Research Polytechnic University

This article describes simulation of gravity on the flat contours, describes an algorithm for finding the top of the details, which will take place around the turn of and an algorithm for finding the angle of rotation.

Keywords: gravity; the center of mass.

Введение. На сегодняшний день в различных сферах производства возникают проблемы, связанные с задачами упаковки. Задачи упаковки заключаются в оптимизации размещения объектов в заданных областях с выполнением определенных условий [1, 2]. Одним из подходов для решения данных задач является использование двумерных физических движков [3]. Для простоты реализации объект (деталь) представлен набором вершин, а процессы взаимодействия описываются силой тяжести, силой упругости и законом сохранения импульса. Рассмотрим учет силы тяжести.

Учет силы тяжести. Перемещение объекта под действием силы тяжести разобьем на два этапа: свободное падение до нулевого уровня (пола) и поворот детали под собственным весом.

1. Свободное падение. Представим процесс движения детали как дискретный (изменение координат вершин детали происходит каждые 100 мс.). Падение детали до нулевого уровня происходит по алгоритму, который указан на схеме (рис. 1).

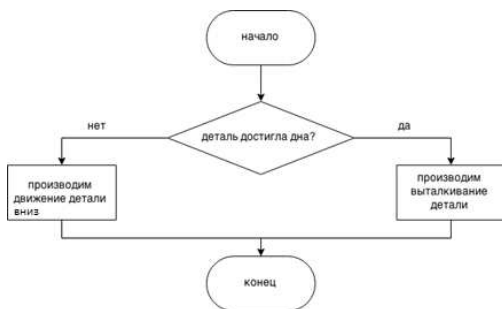


Рис. 1. Свободное падение детали до нулевого уровня

Изменение координаты y при движении детали происходит у всех вершин детали одновременно. Для этого используется матрица переноса

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & n & 1 \end{bmatrix},$$

где n отвечает за смещение по y . Первый столбец матрицы отвечает за координату детали по x , второй – по y , третий – за однородную координату [4]. Отметим, что для большей реалистичности физического процесса n зависит от площади детали.

Поскольку процесс движения детали дискретный, то возможен случай, когда деталь «проскочит» нулевой уровень (вершина детали окажется ниже нулевого уровня). Чтобы предотвратить данную ситуацию, необходимо реализовать механизм выталкивания детали. Для этого использована матрица переноса, где параметр n равен разнице между координатами y нулевого уровня и вершиной, «проскочившей» пол. На рис. 2 наглядно продемонстрирована данная ситуация.



Рис. 2. Выталкивание детали

2. Поворот детали. Для поворота детали под собственным весом необходимо определить угол поворота и вершину, вокруг которой будет происходить поворот. Алгоритм нахождения вершины, вокруг которой будет происходить поворот, показан на рис. 3.

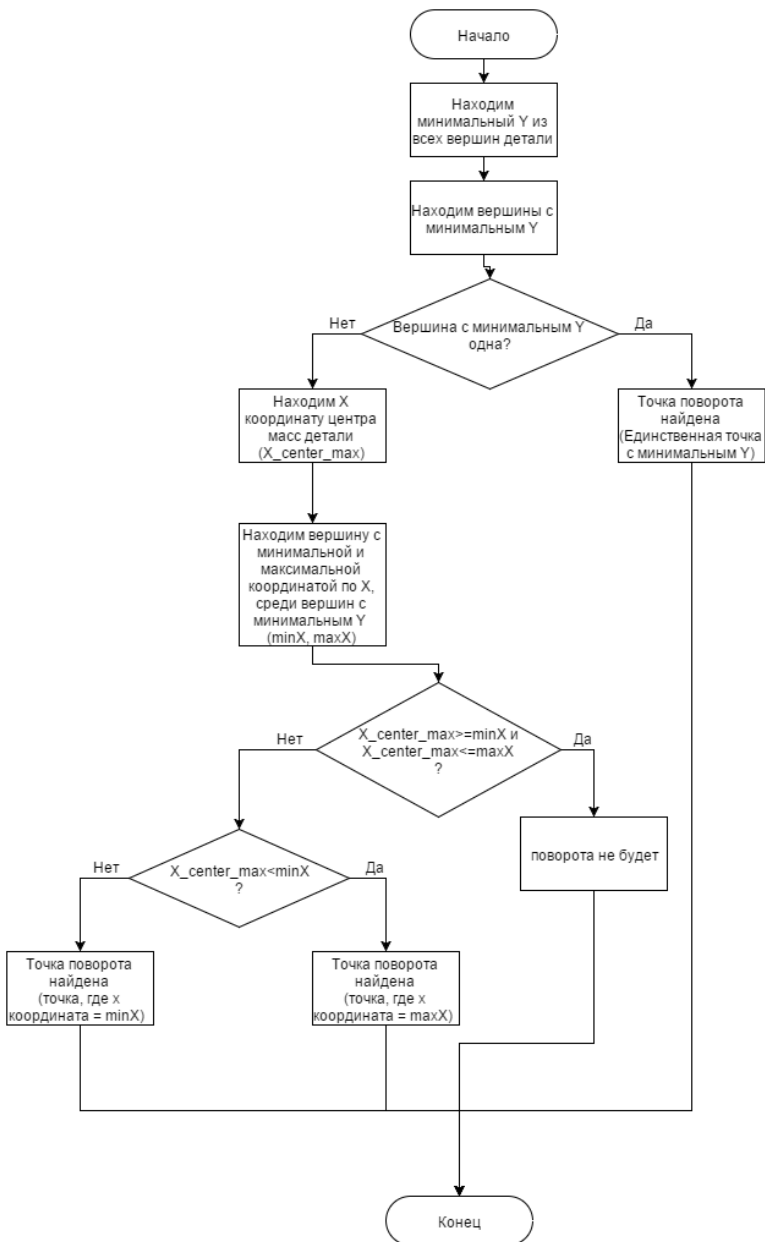


Рис. 3. Схема алгоритма нахождения вершины, вокруг которой будет произведен поворот

Вначале находим минимальную координату по y из всех вершин детали. Далее находим все вершины детали, у которых координата по y равна минимальной. Затем производим проверку на количество найденных вершин: если вершина одна, то поворот детали будет происходить относительно нее, иначе находим координату центра масс детали по x (X_center_mass). Находим вершины с минимальной ($\min X$) и максимальной ($\max X$) координатой по x среди уже найденных вершин (вершины с минимальной y координатой). Далее определяем, где находится X_center_mass : если X_center_mass лежит между $\min X$ и $\max X$, то поворота не будет (деталь находится в равновесном состоянии), если X_center_mass находится левее $\min X$, то поворот будет налево и вершина, вокруг которой будет поворот, и есть вершина с $\min X$ координатой по x , иначе, будет поворот направо и искомая вершина – вершина с $\max X$ координатой по x .

Алгоритм нахождения угла поворота показан на рис. 4. Находим координату центра масс детали по x (X_center_mass). Если X_center_mass совпадает с x координатой вершины, вокруг которой будет происходить поворот, то поворота не будет (деталь находится в равновесном состоянии), иначе, копируем все вершины детали, кроме вершины, вокруг которой будет происходить поворот, в дополнительную коллекцию. Если $X_center_mass < x$ координаты вершины поворота, то удаляем из дополнительной коллекции все вершины, лежащие правее вершины поворота по x (поворот налево), иначе, будет поворот направо – удаляем из дополнительной коллекции все вершины, лежащие левее вершины поворота по x . Далее вычисляем тангенсы углов, образованные нулевым уровнем и прямыми, содержащими точки из дополнительной коллекции и точку поворота. Определяем угол, соответствующий максимальному тангенсу, это и будет искомым углом поворота.

Поворот детали происходит с использованием матрицы [5]:

$$\begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ -m(\cos\theta - 1) + n\sin\theta & -n(\cos\theta - 1) - m\sin\theta & 1 \end{bmatrix},$$

где θ – угол поворота, а m и n – координаты точки поворота по x и y соответственно. Примеры поворота детали показаны на рис. 5.

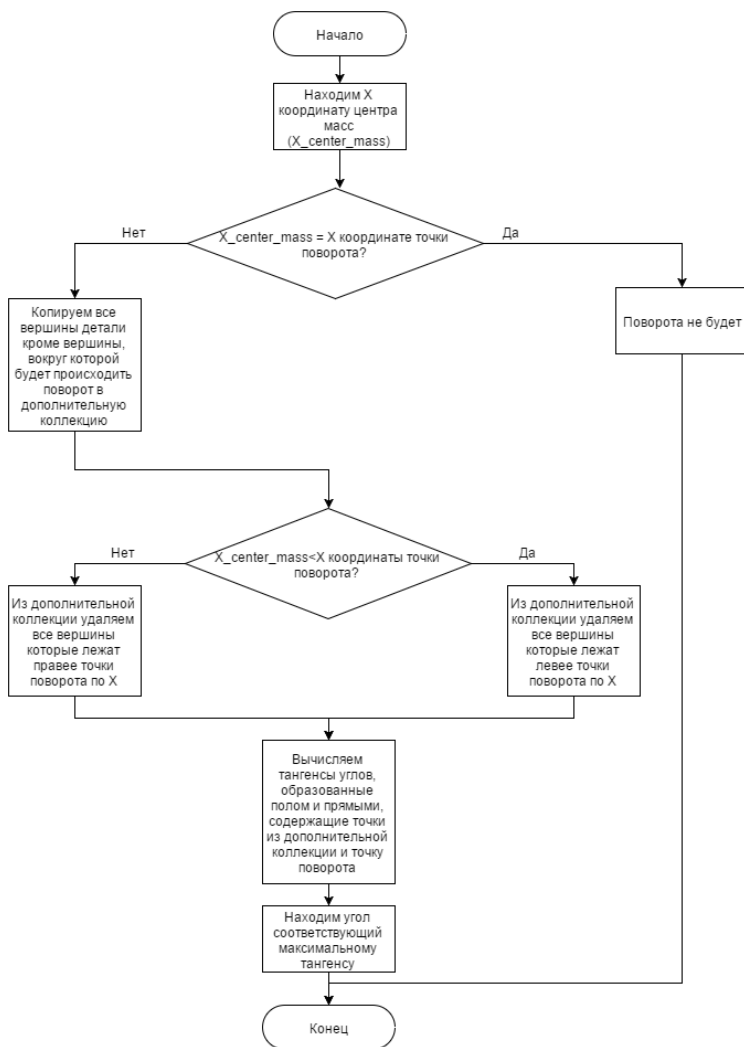


Рис. 4. Схема алгоритма нахождения угла поворота

Случай № 1 – поворот налево, так как центр масс находится левее вершины с самой минимальной x координатой. Случай № 2 – поворота не будет, так как центр масс лежит между точками опоры. Случай № 3 – поворота не будет, так как координаты по x центра масс и опоры совпадают.

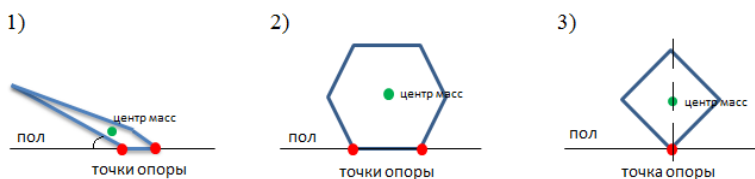


Рис. 5. Примеры поворота детали

Заключение. Таким образом, была произведена имитация действия силы тяжести на плоские фигуры, описан алгоритм поворота детали под собственным весом, который состоит из алгоритма нахождения точки поворота и угла поворота.

Библиографический список

1. Мурзакаев Р.Т., Шилов В.С., Брюханова А.А. Программный комплекс фигурного раскроя материала Itas Nesting // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2015. – № 13. – С. 15–25.
2. Мурзакаев Р.Т., Шилов В.С., Мезенцев А.С. Раскрой листовых материалов с учетом технологических зазоров оборудования для резки деталей // Электротехника. – 2014. – № 11. – С. 23–27.
3. Усанин Ф.Н., Ветчанинов А.А. Моделирование процесса уплотнения геометрических объектов на основе плоской гравитационной модели // Автоматизированные системы управления и информационные технологии: материалы всерос. науч.-техн. конф. – Пермь, Изд-во Перм. нац. исслед. политехн. ун-та, 2015. – С. 124–127.
4. Однородные координаты [Электронный ресурс]. – URL: http://compgraphics.info/2D/affine_transform.php (дата обращения: 09.09.2015).
5. Поворот вокруг произвольной точки [Электронный ресурс]. – URL: http://sernam.ru/book_mm3d.php?id=45 (дата обращения: 09.09.2015).

АНАЛИЗ СЕТЕВОГО И ЛЕНТОЧНОГО ГРАФИКА В ЗАДАЧАХ ПЛАНИРОВАНИЯ ПРОИЗВОДСТВА

**Студенты М.В. Голохвастова, Д.Е. Проданов,
ассистент Н.Ф. Шаякбаров**

Пермский национальный исследовательский
политехнический университет

В статье приведён сравнительный анализ сетевого и ленточного графиков. На его основе выдвигается предположение о проектировании системы, способной объединить достоинства обоих графиков. Приводится описание возможной структуры информационной базы.

Ключевые слова: диаграмма Ганта, информационная база, планирование производства, сетевой график.

ANALYSIS OF NETWORK AND RIBBON GRAPHICS IN PRODUCTION PLANNING PROBLEM

**Students M.V. Golokhvastova, D.E. Prodanov,
assistant N.F. Shayakbarov**

Perm national research polytechnic university

In this article comparative analysis of project network and Gantt chart is given. Based on it put the assumption about system is able to combine the advantages of both methods. Description of the possible information base is given.

Keywords: Gantt chart; the informational base; production planning; project network.

Важнейшим этапом в жизни любого предприятия является этап планирования. На данном этапе важную роль играет ритмичная работа, в течение которой на каждом рабочем месте и участке, в каждом подразделении выполняется в данный момент времени определенное количество продукции. Такая работа является действенной, рациональной и обладает признаком высокой культуры [10]. Но, как показывает практика, достичь такого ритма работы удаётся крайне редко. Ведь нужно не только согласовать выполнение всех работ во времени, но и обеспечить в дальнейшем их производственную пропорциональность. Отклонения от намеченного ритма могут привести к весьма значительным экономическим тратам на предприятии: простои оборудования, привлечение денежных средств для прикрепления дополнительных рабочих ресурсов и пр.

Для снижения вероятности возникновения таких отклонений на предприятиях, как правило, используют системы оперативно-календарного планирования [9, 8, 11]. Эти системы являются базисом, обеспечивающим высокоэффективную работу любого промышленного предприятия [13].

Важным элементом любой системы оперативно-календарного планирования является представление полученных результатов с помощью диаграмм и графиков.

1. Визуализация данных. В настоящее время существует множество методов для эффективного отображения результатов планирования. Однако некоторые методы имеют большую популярность, а другие никак не используются.

Наибольшей популярностью пользуется графическое отображение календарных планов в виде линейных диаграмм и сетевых графиков.

Сетевой график – графическое изображение процессов, соединенных между собой дугами, на котором можно проводить эксперименты и выяснять, к каким результатам приведет то или иное изменение [1, 7]. Пример сетевого графика представлен на рис. 1.

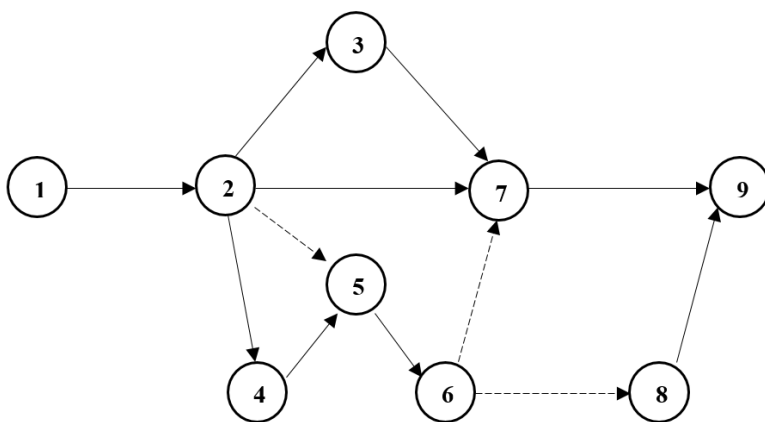


Рис. 1. Пример сетевого графика

Для начала рассмотрим достоинства сетевых графиков [7, 6, 4]. Во-первых, сетевой график имеет наглядное обозначение взаимосвязей между операциями. Во-вторых, он обладает легкостью внесения дополнительных работ. В-третьих, легко может быть выделена

последовательность работ, которая определяет время окончания всего комплекса работ – критический путь. Кроме этого, сетевой график позволяет определить временные запасы работ, лежащих не на критическом пути, что позволяет осуществлять корректировку графика.

Из основных недостатков стоит отметить следующие: сетевой график избыточен для простых проектов и требует высокой квалификации от составителя [1]. Также необходимо знать и учитывать правила построения графиков. В дополнение, некоторые виды сетевых графиков нуждаются в пояснении и требуют многих расчётов.

Линейный график Ганта предназначен для иллюстрации этапов работ проекта. Для каждой работы указываются сроки выполнения, даты начала и окончания работы, связи между работами, а на самой диаграмме все работы показываются в виде полосок, соединенных стрелками. На этих полосках показаны процент выполнения работы и назначенные на нее ресурсы. Визуализация значительно облегчает работу менеджеру проекта. Пример диаграммы Ганта представлен на рис. 2.



Рис. 2. Пример диаграммы Ганта

Данный тип графиков очень часто используется в системах оперативно-календарного планирования в силу того, что он является наглядным и простым. Теперь перейдем к недостаткам диаграмм Ганта. Во-первых, на диаграмме отсутствует наглядное обозначение взаимосвязей между операциями. Во-вторых, из-за того, что проекты не являются статичными и они постоянно меняются, график сложно корректировать. В-третьих, для крупных проектов диаграмма Ганта становится чрезмерно тяжеловесной и теряет всякую наглядность [7, 3].

Подводя промежуточные итоги, представим данные по указанным выше методам в виде сравнительной таблицы.

Сравнительная характеристика диаграммы Ганта и сетевого графика

	Сетевой график	Диаграмма Ганта
Достоинства	<ol style="list-style-type: none"> 1. Наглядно обозначает взаимосвязи между работами. 2. Легко вносить дополнительные работы. 3. Легко может быть выделен критический путь. 4. Позволяет определить временные резервы по каждой из работ. 5. Возможно корректировать график 	<ol style="list-style-type: none"> 1. Наглядность. 2. Простота
Недостатки	<ol style="list-style-type: none"> 1. Избыточна в простых проектах. 2. Необходимы высокая квалификация, знание правил построения графиков. 3. Требует много их расчётов 	<ol style="list-style-type: none"> 1. Отсутствует обозначение взаимосвязей между операциями. 2. График сложно поддается корректировке. 3. В крупных проектах диаграмма становится тяжеловесной

2. Проектирование информационной базы. Целью разработки информационной базы данных является решение задачи календарного планирования производства. Для того чтобы построить информационную базу, необходимо понять, какая производственная информация будет использована.

Самым главным элементом будет продукция, выпускаемая предприятием. Описание технологии выпускаемых изделий можно представить в виде иерархических ориентированных сетей, определяющих конструкцию изделия [5]. Начальной вершиной таких сетей является готовое изделие, а последующие ветвления представляют собой сборочные единицы и детали. При добавлении времени к технологическим операциям мы сможем получить полный образ изготовления изделия.

Таким образом, мы можем выделить главные сущности информационной базы – «Изделие» и «Технологическая операция». Рассмотрим их более подробно.

Сущность «Изделие» нужна для детального описания изделия на предприятии. Среди её основных атрибутов выделим следующие: уникальный «Код» изделия, «Обозначение», «Тип», показывающий, относится ли изделие к детали или сборочной единице, «Версия», которая нужна для отслеживания версионности изделия, «Статус»

показывает, на каком этапе находится изделие в данный момент, «Идентификатор родителя» необходим для построения иерархичной ориентированной сети.

Каждое изделие будет изготавливаться по отдельному маршруту, поэтому следующей сущностью для описания будет «Маршрут». Связь между сущностями будет иметь вид «один ко многим», т.е. одно изделие будет иметь много маршрутов.

Следовательно, выделим сущность «Маршрут», который будет связан с определенным изделием. Рассмотрим атрибуты, входящие в данную сущность. «Код» является идентификатором сущности, «Статус» показывает завершенность изготовления изделия, «Ответственный» – атрибут для обозначения данных ответственного сотрудника, «Дата утверждения».

Далее «Маршрут» включает в себя последовательность из технологических операций. Связь между сущностями будет иметь вид «многие ко многим», в последующем она заменится промежуточной сущностью.

Сущность «Технологическая операция» представляет собой справочник, в котором перечисляются все операции, применяемые на предприятии. Выделим атрибуты данной сущности. К ним относятся: «Код операции», «Наименование», «Разряд» – описывает уровень подготовки работника, «Идентификатор родителя».

Каждая технологическая операция связана с конкретными нормативами выполнения, поэтому следует выделить одноимённую сущность. Связь между сущностями – «один ко многим».

«Нормативы» будут представлены трудовыми затратами и затратами материалов на конкретную операцию. Трудовые нормативы будут представлены следующими атрибутами: «Время начала выполнения операции», «Время окончания выполнения операции», «Основное», «Вспомогательное время», «Подготовительно-заключительное время», «Время переналадки» и пр. [12]. Затраты материалов будут представлены количественной величиной требуемого материала – атрибут «Количество».

Для выполнения технологических операций необходим персонал, владеющий определенной профессией, поэтому между сущностями «Технологическая операция» и «Персонал» необходима связь «многие ко многим», вследствие чего необходимо разбить данную связь дополнительными сущностями (профессии).

Сущность «Персонал» в информационной базе тоже будет присутствовать. Информация о сотрудниках, такая как «Фамилия», «Имя», «Отчество», «Профессия», «Разряд», «Расписание смены», определяет атрибуты данной сущности.

Выводы. Все методы по-своему эффективны, у них есть достоинства и недостатки, каждый метод хорошо применим при разных направленностях и особенностях проекта. Но некоторые из них более популярны среди менеджеров проектов по объективным причинам, которые можно отследить по сравнительному анализу. Один из главных критериев работы – визуализация.

Информационная база данных является связующим звеном между сетевым графиком и ленточным. Правильная организация структуры базы данных позволит использовать достоинства двух графиков: в случае сетевого – описать более подробно производственный процесс, а именно построение маршрутов изделий с возможностью оперативного изменения операций и нахождения критического пути, в случае ленточного графика – отображать процесс во временном интервале с процентом выполнения и необходимыми ресурсами.

Библиографический список

1. Андреева А.О. «Методы эффективного планирования проекта» [Электронный ресурс] // Экономика и менеджмент инновационных технологий. – 2015. – № 3. – URL: <http://ekonomika.snauka.ru/2015/03/7579> (дата обращения: 13.11.2015).
2. Аралбаева Ф.З., Русяева О.А. Критерии эффективного проектного менеджмента // Вестник Оренбург. гос. ун-та. – 2006. – Вып. 10–2. – С. 258–264.
3. Бродецкий Г., Ермолов С. Возможности использования скрытого резерва оптимизации решений на сетевых графиках логистических проектов // ИТКОР. Экономико-математические методы и модели в логистике. – 2009. – С. 136–143.
4. Гасанбеков С.К., Лубенец Н.А. Сетевое планирование как инструмент управления проектами // Известия МГТУ «МАМИ». – М., 2014. – Вып. 1(19). – Т. 5. – С. 21–25.
5. Долгова Ю.И. Информационная база для планирования производства на машиностроительном предприятии [Электронный ресурс] // Устойчивое инновационное развитие: проектирование и управление. – 2010. – Т. 6. – № 2(7). – Ст. 3. – URL: <http://www.gypravlenie.ru/wp>

content/uploads/2010/06/3_Dolgova_Informationnaya-baza.pdf (дата обращения: 14.11.2015).

6. Исследование операций в экономике: учеб. пособие для вузов / Н.Ш. Кремер, Б.А. Путко, И.М. Тришин, М.Н. Фридман; под ред. проф. Н.Ш. Кремера. – М.: ЮНИТИ, 2002. – 407 с.

7. Новицкий Н.И. Сетевое планирование и управление производством: учеб.-практ. пособие. – М.: Новое знание, 2004. – 159 с.

8. Планирование производства [Электронный ресурс]. – URL: <http://www.parus.ua/ru/308/#2> (дата обращения: 13.11.2015).

9. Рубахина В. СПРУТ-ОКП: Оперативно-календарное планирование – путь к эффективному управлению предприятием [Электронный ресурс]. – URL: <http://www.sprut.ru/company/press/articles/ritm-76> (дата обращения: 13.11.2015).

10. Сергеева И. Учебно-методический комплекс по дисциплине «Оперативно-производственное планирование» [Электронный ресурс] / Институт экономики и управления. – URL: <http://www.sergeeva-i.narod.ru/opp/page4.htm> (дата обращения: 13.11.15).

11. Система оперативного календарного планирования от Preactor International как APS-система [Электронный ресурс]. – URL: <http://www.intenta-it.ru/resheniya/program-solutions/preactor-aps/> (дата обращения: 13.11.2015).

12. Трудовые нормативы [Электронный ресурс] // Руководство пользователя СПРУТ-ОКП. – URL: http://www.sprut.ru/files/SprutOKP/documentation/System/index.html?top_d61400.htm (дата обращения: 13.11.2015).

13. Фролов Е. Оперативное планирование производства [Электронный ресурс] // Директор: настольный журнал ИТ-руководителя. – 2013. – № 5. – URL: <http://www.osp.ru/cio/2013/05/13035711/> (дата обращения: 13.11.2015).

МИНИ-КОМПЬЮТЕРЫ

Студент А.С. Иванов

Южно-Уральский государственный национальный
исследовательский университет, г. Челябинск

В статье осуществляется обзор портативной вычислительной техники, а также перечисляются разработки сегодня в этой сфере. В статье не рассматриваются смартфоны и планшетные компьютеры.

Ключевые слова: мини-компьютеры, операционная система, ОЗУ, процессор.

MINI-COMPUTERS

Student A.S. Ivanov

South-Ural State National Research University, Chelyabinsk

This article includes an overview of portable computing, and lists current developments in this field. Article does not affect smart phones and tablet computers.

Keywords: Mini-computer, operating system, RAM, CPU.

Современные тенденции в развитии вычислительной техники направлены на уменьшение энергопотребления, размеров и повышение производительности. К этим целям разработчики стремились еще со времени появления первых компьютеров, но сегодня компьютеры смогли достигнуть размеров банковской карты.

Рассмотрим несколько примеров таких разработок. В основном мини-компьютеры собираются на одной печатной плате, на которой установлены: процессор, оперативная память (ОЗУ), устройства ввода-вывода, постоянная память (ПЗУ), а также другие дополнительные модули. В основном такие мини-компьютеры используются в качестве систем для разработчиков, однако они также могут применяться в системе образования и в повседневной жизни. Например, сегодня все чаще мини-компьютеры выпускаются на операционной системе Android. Рассмотрим одно из таких устройств.

CX-919 – это мини-компьютер под операционной системой Android, оснащенный четырехъядерным процессором Rockchip RK3188 с частотой 1,6 ГГц, USB-портом, для подключения внешних устройств (компьютерная мышь, клавиатура), модулем Wi-Fi, 2 Гб оперативной памяти, видеопроцессором Mali 400 и выходом HDMI для

подключения CX-919 к монитору, телевизору или другим подобным устройствам. Внешний вид устройства представлен на рис. 1, а его плата – на рис. 2.



Рис. 1. Внешний вид CX-919

Стоит отметить, что данное устройство и ему подобные в основном предназначены для потребительского сектора и не обладают большим функционалом. Стоимость таких устройства составляет порядка 30–50 \$ или 3–5 тыс. рублей.



Рис. 2. Плата устройства CX-919

Однако многие пользователи не готовы переходить с привычной для них системы Windows на что-то другое. Эту проблему частично решила компания Intel. В 2012 г. она выпустила линейку процессоров

Intel Atom, которые поддерживают ОС Windows. Рассмотрим пример такого мини-компьютера:

MeegoPad T07 – это компьютер размером с флешку, оснащенный четырёхядерным процессором Intel Atom x5-Z8300 с частотой от 1,44 до 1,84 ГГц на каждом из четырех ядер, 4 Гб оперативной памяти, одним выходным разъемом HDMI, разъемом питания, слотом для внешней карты памяти, двумя полноразмерными USB-портами, а также модулями Wi-Fi и Bluetooth. MeegoPad T07 работает на операционной системе Windows 10.

Таким образом, подключив MeegoPad T07, мышь и клавиатуру, пользователь может работать в Microsoft Office, Microsoft PowerPoint, Paint, Google Chrome и в других полноценных программах на Windows. Стоимость такого устройства составляет 110 \$ или 6–8 тыс. рублей. Внешний вид такого устройства представлен на рис. 3.



Рис. 3. MeegoPad T07

Другая кампания, которая сделала шаг в этом же направлении, – Frontier Developments. В 2011 г. был представлен одноплатный компьютер Raspberry Pi. Изначально разработчики планировали использовать его в качестве системы для обучения информатике. Но устройство получило широкое применение и популярность. На сегодня последней разработкой компании Frontier Developments является мини-компьютер Raspberry Pi Zero. Он был представлен 26 ноября 2015 г. и оснащен процессором Broadcom BCM2835 с частотой 1 ГГц, оперативной памятью объемом в 512 Мб, 2 микро USB-портами и разъемом мини-HDMI. Особенностью Raspberry Pi является поддержка

нескольких операционных систем: FreeBSD, системы на ядре Linux, Windows, Android, Firefox OS и др. Однако главными особенностями Raspberry Pi являются размер и цена. Так, стоимость Raspberry Pi Zero составляет всего 5 \$. Внешний вид Raspberry Pi Zero представлен на рис. 4.

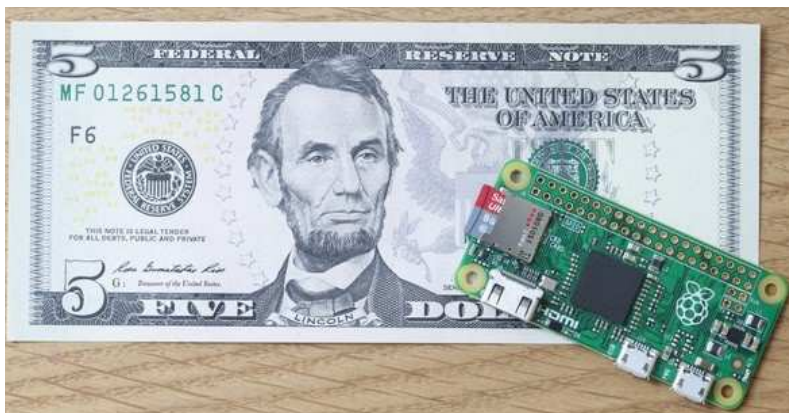


Рис. 4. Сравнение размера Raspberry Pi Zero и 5 долларовой купюры



Рис. 5. Intel Edison

В заключение стоит отметить, что гонка за компактностью устройств продолжается. Так, на конференции CES 2014 компания Intel продемонстрировала новую платформу под названием Edison, кото-

рая имеет размер карты памяти формата SD (рис. 5). Intel Edison оснащен процессором «Tangier» с частотой 500 МГц, оперативной памятью объемом 1 Гб, Wi-Fi и Bluetooth-модулем, одним разъемом микроUSB. Новика предназначена в большей мере для разработчиков, однако компания Intel будет поощрять самые интересные проекты. Стоимость устройства составляет 20 \$.

Вычислительная техника развивается удивительно быстрым темпом. Первые компьютеры занимали целые комнаты, а сейчас они могут поместиться в кармане.

Библиографический список

1. Процессор Intel Atom [Электронный ресурс]. – URL: <http://www.intel.ru/content/www/ru/ru/processors/atom/atom-processor.html>
2. Raspberry Pi. [Электронный ресурс]. – URL: <https://www.raspberrypi.org/> (дата обращения: 09.09.2015).
3. MeegoPad T07 [Электронный ресурс]. – URL: <http://www.3dnews.ru/923969> (дата обращения: 09.09.2015).
4. CX-919 [Электронный ресурс]. – URL: <http://www.android-smart-tv.ru/category/android-tv-box-2/cx-919> (дата обращения: 09.09.2015).

МОДЕЛИ РЕАЛЬНЫХ СИСТЕМ ПРИ ОБУЧЕНИИ ТЕХНИЧЕСКИХ СПЕЦИАЛИСТОВ

Канд. техн. наук М.И. Коныхов

Филиал Казанского национального исследовательского технического университета, г. Нижнекамск

В статье рассмотрены вопросы моделирования реальных систем при обучении технических специалистов.

Ключевые слова: моделирования, обучение.

MODELS OF REAL SYSTEMS WHILE TRAINING TECHNICIANS

Ph.D. in Technical Sciences M.I. Konukhov

Branch of Kazan National Research Technical University, Nizhnekamsk

In this article the aspects of modeling real systems when training technical professionals.

Keywords: simulation, training.

Подготовка студентов технических специальностей требует наглядности. При обучении технических специалистов возникает необходимость в использовании полунатурного моделирования. А это, естественно, требует применения новых технических средств обучения. Основными причинами низкой эффективности существующих технических средств обучения являются малая наглядность для технических специалистов и невозможность учета обучающим всех показателей при оценивании выполнения упражнений обучаемым.

Для исключения этих недостатков и повышения уровня обученности специалистов считается перспективным применение комплексов, полностью имитирующих поведение технических систем, которые включают в свой состав вычислительные системы, специализированные и комплексные тренажеры [1–2].

Информационную поддержку управлению имитационным комплексом можно осуществлять с помощью автоматизированной системы управления профессиональной подготовленностью, которая должна стать неотъемлемой составной частью каждого технического средства комплекса и способствовать управлению ходом процесса обучения на каждом уровне подготовки специалистов.

Эта система должна будет решать следующие основные задачи [3]: оценивание качества выполнения тренировочных задач; статистический анализ результатов обучения и выявление тенденций развития навыков;

выработка рекомендаций обучаемому и преподавателю по дальнейшему прохождению курса обучения.

Структурно комплекс представляет собой иерархическую систему. Верхний уровень системы осуществляет координацию всей работы и контроль за процессом подготовки. Нижний уровень – специализированные подсистемы, обеспечивающие имитацию поведения технической системы.

Функциональная организация таких систем совпадает. Однако задачи, решаемые на разных уровнях, различны. В частности, она должна осуществлять: формирование интегральной оценки общего уровня подготовленности каждого обучаемого по результатам всех форм обучения и определение степени готовности к выполнению более сложных упражнений; формирование банка данных на каждого будущего специалиста, его хранение и пополнение после каждого занятия; формирование информации о готовности обучаемого к переходу на следующий этап обучения.

Для решения этих задач система должна содержать следующие модули: систему сбора информации, систему выработки рекомендаций, базу данных и базу знаний. В системе выработки рекомендаций, помимо информации о текущем состоянии имитируемой системы и условиях внешней среды, формируемых в системе индикации, используются значения переменных, характеризующих внешнюю среду, и результаты функционирования системы на разных стадиях, предшествующих текущей, хранящиеся в базе данных. Из базы знаний в систему поступает информация о математических моделях процессов, протекающих в имитируемой системе, а также необходимые алгоритмы обработки.

База данных может быть представлена иерархической, сетевой или реляционной структурой. База знаний будет содержать процедурную и декларативную информацию и может быть построена на основе логической, сетевой, фреймовой или продукционной модели.

Результаты работы системы используется преподавателем для осуществления управления ходом подготовки специалиста.

Библиографический список

1. Козин Р.Г. Математическое моделирование: учеб. пособие. – М.: Изд-во МИФИ, 2006. – 85 с.
2. Шукшунов В.Е. Тренажерные системы. – М.: Машиностроение, 1981. – 254 с.
3. Соловов А.В. Электронное обучение: проблематика, дидактика, технология. – Самара: Новая техника, 2006. – 462 с.

РАЗРАБОТКА СИСТЕМЫ ДЕТЕКТИРОВАНИЯ УЛЫБАЮЩИХСЯ ЛИЦ НА ИЗОБРАЖЕНИЯХ

Студент Н.Г. Макагонов

Пермский национальный исследовательский
политехнический университет

В статье рассмотрены аспекты разработки системы детектирования улыбающихся лиц. Поиск производится с помощью обученных каскадов Хаара, открытой библиотеки OpenCV. В результате, достигнут приемлемый уровень качества распознавания улыбающихся лиц.

Ключевые слова: OpenCV, эмоции, каскад Хаара.

DEVELOPMENT SYSTEM OF RECOGNITION SMILING FACES IN THE IMAGES

Student N.G. Makagonov

Perm National Research Polytechnic University

In this article the aspects of develop of a smiling faces recognition system are considered. The detecting is performed using the trained Haar cascades, open source library OpenCV. As a result the required level of quality of recognition smiling faces is achieved.

Keywords: OpenCV, emotions, Haar cascade.

Человеческие эмоции с древних времен являются одним из методов обмена информацией. Зачастую эмоции оказываются даже более информативным средством, чем привычная словесная форма. Мимолетное выражение лица может заменить целый набор предложений. Разные эмоции в разных культурах могут иметь совершенно разные смыслы. Однако всеобъемлющий процесс глобализации практически полностью стандартизировал и этот аспект человеческой жизни. Актуальность эмоций в настоящее время не подвергается сомнению и лишний раз неопровержимо доказывается тем фактом, что в социальных сетях эмоции нашли свое отражение в виде стилизованных изображений человеческого лица, выражающих какие-либо эмоции, так называемых «смайликов». Однако современные вычислительные системы, направленные на общение с человеком, практически не используют эмоции. Одной из наиболее универсальных и часто используемых эмоций является улыбка. Данная статья посвящена практическому исследованию возможности распознавания

человеческой улыбки без использования высокопроизводительных вычислительных систем

Для реализации подобных исследований существует несколько программных продуктов, предоставляющих возможность получения и обработки видеoinформации. Наиболее известными и популярными из них являются инженерный пакет MatLab и открытая библиотека компьютерного зрения OpenCV. MatLab является мощным пакетом, при помощи которого может решаться широкий круг инженерных задач, таких как матричный анализ, реализация алгоритмов на основе нечеткой логики, разработка и моделирование систем автоматического управления, систем массового обслуживания, телекоммуникационных систем, и т.д. Также MatLab имеет специальный пакет расширений ImageProcessingToolbox, предоставляющий достаточно обширный круг возможностей по обработке и анализу изображений [1]. Однако для использования данного пакета требуется обязательное наличие дорогостоящей лицензии. Библиотека OpenCV является открыто распространяемым программным продуктом, что является ее бесспорным преимуществом. Также данный программный продукт специализируется конкретно на компьютерном зрении, вследствие чего имеет не уступающие, а зачастую превосходящие возможности в области обработки и анализа изображений. Вследствие этого была выбрана библиотека OpenCV.

Принцип предлагаемого метода, решения задачи детектирования улыбающегося человеческого лица, заключается в разбиении данной задачи на 2 подзадачи: нахождение лица либо нескольких лиц на исходном изображении, а также проверка принадлежности данного лица к классу улыбающихся лиц. Первая подзадача является классической задачей компьютерного зрения и имеет множество решений, основанных на различных принципах, таких как детектирование цвета кожи, контурный анализ и т.д. Однако наиболее перспективными и обладающими хорошим соотношением скорости распознавания к качеству распознавания являются алгоритмы на основе каскадов признаков. В библиотеке OpenCV реализовано 2 наиболее распространенных из них, это алгоритм Виолы и Джонса [2, 3] и алгоритм LBP (Local Binary Patterns) [4].

Данные алгоритмы были специально разработаны для распознавания лиц и относятся к разряду алгоритмов на основе машинного обучения, т.е. для работы требуют наборы правильных и неправильных

примеров. На их основе создается так называемый каскад, являющийся специфически составленным набором особенностей, характерных для правильных примеров и не характерных для неправильных. Алгоритм Виолы и Джонса показывает более точные результаты распознавания, однако преимуществом LBP является скорость как в плане обучения, так и в плане распознавания. В данной разработке был использован алгоритм Виолы и Джонса, так как для получения удовлетворительных результатов с помощью алгоритма LBP требуется более сложная и тонкая настройка параметров обучения. Поскольку детектирование человеческого лица (как было изложено ранее) является уже решенной задачей, то библиотека OpenCV снабжена прекрасно обученным каскадом для детектирования лиц. Таким образом, решение предложенной задачи сводится к решению только второй подзадачи, т.е. определению наличия улыбки на лице, находящемся в области, указанной первым каскадом.

На рис. 1 виден конечный результат работы первого каскада: лицо обнаружено и выделено эллипсом.



Рис. 1. Пример детектирования лица

Решение второй подзадачи сводится к обучению аналогичного каскада. Однако так как данный каскад будет предназначен для анализа изображений, полученных в результате детектирования другим

каскадом, то ложный набор обучающей выборки должен содержать только изображения, отображающие лица, не выражающие улыбку, и найденные с помощью первого каскада, а также изображения, являющиеся ложными срабатываниями первого каскада. Данный принцип обусловлен тем, что второй каскад в отдельности не планируется использовать. Таким образом, второй каскад будет не пригоден без использования первого. При этом в тандеме с первым он не только пригоден для решения поставленной задачи, но и выполняет фильтрацию ложных срабатываний первого каскада за счет наличия в обучающей выборке ложных срабатываний первого каскада.

Для обучения каскадов, распознающих лица, обычно требуется от 1000 до 6000 положительных примеров, в данной работе было использовано немногим более 1000 положительных примеров. Поиск и обработка такого количества изображений невозможны вручную, в связи с чем была написана программа, загружающая изображения по определенному запросу из Интернета и вырезающая из данных изображений области, детектированные с помощью каскада для поиска лиц.

Также следует заметить, что обучающая выборка должна являться репрезентативной, т.е. в нее должны входить изображения лиц различных национальностей, рас, полов. При этом необходимо наличие в обучающей выборке изображений с естественными помехами, такими как платок, борода, очки. За счет вышеуказанных принципов достигается минимизация ложноотрицательных срабатываний каскада.



Рис. 2. Пример детектирования улыбающихся и не улыбающихся лиц

На рис. 2 виден конечный результат работы тандема из двух каскадов: все детектированные лица выделены эллипсами. 1, 2 – эллипсы соответствуют лицу без улыбки, 3 – эллипс соответствует улыбающемуся лицу.

Исследование уровня ложных срабатываний показало точность на уровне 85 %, данное исследование проводилось на тестовой выборке, множество изображений которой не пересекалось с множеством изображений обучающей выборки. Тестовая выборка содержала 50 % изображений с улыбающимися лицами.

В результате была создана программа, реализующая поиск лиц на изображении и распознавание улыбок на них. В дальнейшем планируется создание аналогичного каскада с помощью алгоритма LBP с целью уменьшения времени детектирования, а также усовершенствование данной системы в плане повышения точности распознавания и добавления возможности распознавания других эмоций.

Библиографический список

1. Макагонова М.А., Макагонов Н.Г. Использование нейросетевых методов при классификации объектов // Материалы междунар. науч.-практ. конф. – Пермь: Изд-во РГТЭУ, 2012. – Т. 2. – С. 86–90.
2. Viola P., Jones M. Rapid object detection using a boosted cascade of simple features // Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on. – IEEE, 2001. – Т. 1. – С. I-511-I-518 vol. 1.
3. Lienhart R., Maydt J. An extended set of haar-like features for rapid object detection // Image Processing. 2002. Proceedings. 2002 International Conference on. – IEEE, 2002. – Т. 1. – С. I-900-I-903 vol. 1.
4. Ahonen T., Hadid A., Pietikainen M. Face description with local binary patterns: Application to face recognition // Pattern Analysis and Machine Intelligence, IEEE Transactions on. – 2006. – Т. 28. – №. 12. – С. 2037–2041.

ПРОЕКТИРОВАНИЕ И РЕАЛИЗАЦИЯ ИНФОРМАЦИОННОЙ БАЗЫ ДАННЫХ ДЛЯ ПЛАНИРОВАНИЯ МАШИНОСТРОИТЕЛЬНОГО ПРОИЗВОДСТВА

Магистранты И.Т. Михайлова, М.А. Морозов
М.В. Голохвастова

Пермский национальный исследовательский
политехнический университет

Представлена информация о планировании производства в целом, рассмотрено изготовление основной продукции как отдельный производственный процесс. В работе описаны аспекты проектирования и реализации базы данных для системы планирования производства.

Ключевые слова: база данных, производственный процесс, планирование производства.

DESIGN AND IMPLEMENTATION OF AN INFORMATION DATABASE FOR PLANNING OF MACHINE-BUILDING PRODUCTION

Master Students I.T. Mikhailova, M.A. Morozov, M.V. Golokhvastova
Perm National Research Polytechnic University

This article presents information of manufacturing planning in general, describes the manufacture of basic products as the one of the production processes. Described aspects of design and implementation of the database for the manufacture planning system.

Keywords: database, manufacturing process, manufacturing planning.

Введение. В связи с быстрым развитием информационных технологий широкое распространение получили многочисленные программные комплексы для планирования и управления производством.

Основной целью системы планирования производства является составление оптимального плана производства, который представляет собой план производства отдельных конечных изделий. При создании плана учитываются не только необходимые материалы для производства деталей изделия, но и занятость разнообразных цехов производства в целом и станков в частности. Кроме того, каждый этап производства детали изделия занимает различное время, которое также учитывается при создании плана производства [1].

Таким образом, система планирования производства должна затрагивать как аспекты производства изделий, так и общей работы предприятия в целом. Следовательно, при разработке системы планирования необходимо учитывать не только данные, затрагивающие непосредственно изготовление деталей, но и большой перечень данных о работе всего предприятия.

Важным этапом в разработке подобных систем являются проектирование и разработка информационной базы. Информационная база данных должна содержать данные, необходимые и достаточные для построения плана производства. В первую очередь необходимо определиться со структурой и составом информационной базы данных для планирования производства [2].

Проектирование информационной базы. Поскольку основным процессом в производстве является изготовление основной продукции, т.е. деталей и изделий, то целесообразно использовать в качестве основы информационной базы выпускаемое изделие. На предприятии изделия зачастую представляют собой иерархические ориентированные сети, определённые в соответствии с конструкцией изделия (ЕСКД ГОСТ 2. 108-68).

Этапы производства изделия определяются очередностью прохождения определённых станков или же центров обработки деталей, в которых происходят определённые технологические операции: обработка или производство отдельно взятой детали, или сборка уже созданных деталей в определённое изделие. Стоит заметить, что очередность этапов производства изделия может быть как линейной, так и иметь ветвление и параллельные технологические операции, например, обработка разных деталей для последующего создания из них цельного изделия [3].

При учёте времени каждой операции, в том числе времени перемещения детали или изделия от одного центра обработки к другому, формируется полное представление о времени всего маршрута изготовления изделия. В дальнейшем на основе данных о времени прохождения отдельных этапов маршрута возможна оптимизация плана для уменьшения общего времени изготовления детали [3].

Но основываться при разработке информационной базы только на время производства не целесообразно, так как важную роль играют и другие параметры: необходимый материал, занятость оборудования и потребность в трудовых ресурсах.

Таким образом, нужно учитывать нормы использования ресурсов для каждого этапа производства изделия, т.е. нормы расхода материала, оборудования, инструмента, а также специфические нормы, которые различаются от производства к производству. Кроме того, целесообразно оперировать не только определёнными технологическими операциями, но и рабочими местами, на которых выполняются данные операции. Для удобства можно назвать представленный перечень данных «технологией производства».

Соответственно, можно определить, какие разделы данных необходимо иметь в информационной базе:

1. Справочник материалов, применяемых на предприятии.
2. Конструкторская спецификация, хранящая основные данные о составе изделия (ЕСКД ГОСТ 2.108-68).
3. Технология производства.

Указанных данных достаточно для успешного решения задачи планирования производства, но вполне возможны дополнения или изменения представленного состава данных для различных целей производства.

К примеру, во многих случаях имеет смысл добавить данные учёта производства, такие как наличие в данный момент необходимых материалов на складах, данные о прохождении определённых технологических операций или занятости станков, или центров обработки в реальном времени.

Предложенная информационная база данных системы планирования производства применима к любому машиностроительному предприятию, поскольку базируется на распространённом представлении состава изделия. Также повышается эффективность планирования и управления на предприятии, поскольку структуризация этапов планирования позволяет автоматизировать данный процесс. Кроме того, предложенная информационная база данных может быть использована для мониторинга процесса планирования в реальном времени.

Реализация БД на основе СУБД MySQL. После того как спроектирована БД, можно приступить к её реализации. В качестве СУБД была выбрана MySQL. Это реляционная система управления базами данных, т.е. вся информация хранится в виде таблиц, логически связанных друг с другом. Для доступа к этой информации используется язык запросов SQL. Была выбрана именно эта СУБД, потому что она

достаточно быстрая и надёжная, а также простая и свободно распространяемая система.

Но для работы с MySQL существует возможность не только текстового режима, но и графического. Наиболее популярным приложением по работе с СУБД MySQL является phpMyAdmin. Этот интерфейс помогает упростить работу с БД в MySQL.

В результате анализа производственного процесса производства, а именно изготовления основной продукции, были разработаны таблицы для хранения определённой информации (таблица).

Структура БД

№ п/п	Название	Краткое описание
1	Длительность доставки	Предназначен для хранения времени доставки между участками «отправителями» и «получателями»
2	Элемент варианта маршрута	Содержит последовательность выполняемых логических операций в рабочем центре подразделения
3	Логическая операция (ЛО)	Хранит информацию о ЛО (наименование ЛО и минуты выполнения и т.д.)
4	Подразделения	Содержит информацию о названиях и типах всех подразделений
5	Рабочее место	Предназначен для хранения информации по рабочим местами (какие операции выполняются в данном рабочем месте, в каком подразделении оно и находится, а также наименование оборудования рабочего места и т.д.)
6	Рабочий центр (РЦ)	Содержит информацию о РЦ (код и наименование РЦ, количество оборудования и т.д.)
7	Рабочий центр подразделений (РЦ подразделений)	Хранит информацию о том, какие рабочие центры содержатся в конкретном подразделении
8	Вариант маршрута	Содержит информацию о вариантах маршрута изделия (код и наименование маршрута, дата и т.д.)

Исходя из вышеописанной таблицы, была сформирована логическая модель базы данных (рисунок).

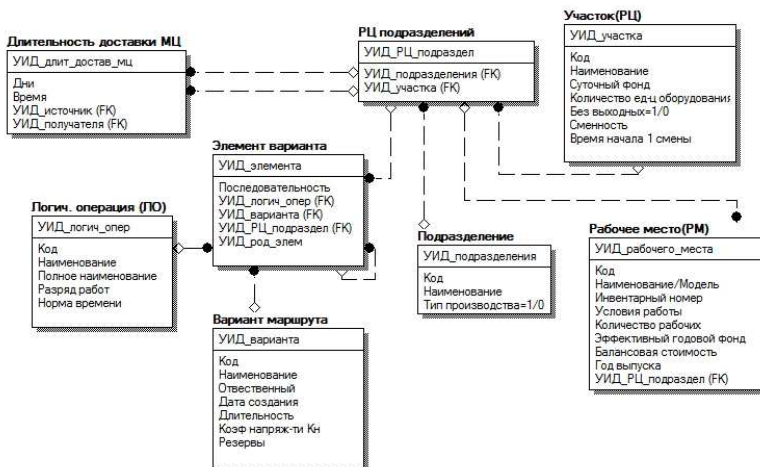


Рис. Логическая модель БД

Заключение. Как говорилось ранее, важным элементом при разработке систем планирования производства является база данных. В предлагаемой БД решена задача изготовления основной продукции. Здесь представлен самый верхний уровень производственного процесса на предприятии. Разработанную базу данных можно считать универсальной, так как её функции можно применять к любому предприятию машиностроения [4]. В дальнейшем планируются развитие базы данных и проектирование самой системы планирования производства.

Библиографический список

1. Овшинов С.А. Оперативно-календарное планирование в серийном и единичном производстве // Вестник ВолГУ. Сер. 3. Экономика. Экология. – 2010. – № 1. – С. 13–18.
2. Полетаев В.А. Автоматизированная поддержка производственного цикла создания изделия машиностроения // Вестник КузГТУ. – 2009. – № 5. – С. 40–44.
3. Долгова Ю.И. Информационная база для планирования производства на машиностроительном предприятии. – 2010. – № 2(7). – С. 23–29.
4. Долгова Ю.И. База данных «ЮНА-АСУП» по планированию машиностроительного производства // Информационные технологии в проектировании и производстве. – 2013. – № 4(152). – С. 52–53.

РЕАЛИЗАЦИЯ АЛГОРИТМА ОБРАТНОГО ПРЕОБРАЗОВАНИЯ УПРАВЛЯЮЩИХ ПРОГРАММ ДЛЯ СТАНКОВ С ЧПУ

Студент А.Н. Поляков, аспирант Д.П. Сорока
Пермский национальный исследовательский
политехнический университет

В статье предложен способ преобразования текста управляющей программы в карту раскроя в рамках программного пакета Itas Nesting. Рассмотрены алгоритм лексического анализа текста управляющей программы и способы воссоздания контуров и деталей с учетом вложенности. Предложенный алгоритм построчно читает управляющую программу и создает упорядоченный список действий инструмента. Полученные действия по пересечению концов примитивов объединяются в контуры. Затем проверяются вхождение контуров друг в друга и их принадлежность к разным деталям.

Ключевые слова: G-коды, технология резки, листового материал, станки с ЧПУ.

IMPLEMENTATION OF REVERSE TRANSFORM ALGORITHM FOR CNC CONTROL PROGRAMS

Student A.N. Poliakov, Graduate Student D.P. Soroka
Perm National Research Polytechnic University

This article describes method of transforming control program into cutting layout within Itas Nesting software package. Lexical analysis for control program and contour rebuilding with inclusion possibility algorithm reviewed. These method reads control program by lines and creates ordered list of instrument actions. Actions then join in contours by primitives ends intersections. Then algorithm checks entrance of contours in each other and defines belonging to different details.

Keywords: cutting technology, sheet material, nesting, CNC machines, G-code.

Описание проблемы и постановка задачи. Создание правильной управляющей программы для станка является не менее важной частью технологического процесса по резке листового материала, чем использование эффективных алгоритмов укладки деталей [1]. Кроме ее непосредственного формирования часто возникает ситуация, когда необходимо провести обратные процессы – создание карты раскроя по уже готовой управляющей программе [2].

Подобная ситуация может возникнуть в случае утраты карты раскроя и необходимости отредактировать имеющуюся к ней управ-

ляющую программу в автоматизированном режиме. Также могут появиться какие-либо сторонние управляющие программы.

Проблемой автоматического формирования управляющей программы уже занимаются такие программные пакеты, как WryKrug или Lantek Expert. Однако модули обратного преобразования реализованы очень слабо или в большинстве случаев отсутствуют совсем.

Поэтому в рамках программного комплекса раскроя листовых материалов Itas Nesting [3] необходимо реализовать функционал, который позволит анализировать текст управляющей программы на G-кодах и восстанавливать карту раскроя с последующим редактированием.

Для решения поставленной задачи необходимо совершить три шага:

- лексический анализ программы на G-кодах;
- воссоздание контуров деталей;
- определение вложенности контуров и воссоздание деталей.

Лексический анализ программы на G-кодах. Первым шагом анализа является разбиение входного текстового файла на строки, которые по своей сути и являются отдельными командами.

Например, G01 X0. Y0. Z100 F200 производит резку по прямой в точку (0,0,100) со скоростью 200.

После этого каждая строка разбивается на множество подстрок, разделенных пробелом. Каждая такая подстрока несет в себе определенную информацию в зависимости от того, с какой буквы она начинается. Разберем уже приведенный пример G01 X0. Y0. Z100 F200: G – тип исполняемой команды.

В данном случае 01 – линейная интерполяция; X – смещение по оси x; Y – смещение по оси y; Z – смещение по оси z; F – скорость инструмента.

Отсюда можно заключить, что в первую очередь следует обращать внимание на подстроки G, так как они определяют тип команды, т.е. тип Action, в который преобразовывается текст.

То же касается команд M, но в отличие от G такие команды распространяются на весь станок, а не на инструмент.

В таблице приведены основные G-команды.

Наиболее распространенные G-коды

Код	Назначение
G00	Холостой ход инструмента
G01	Линейная интерполяция
G02	Круговая интерполяция по часовой стрелке
G03	Круговая интерполяция против часовой стрелки
G40	Отмена компенсации радиуса инструмента
G41	Компенсировать радиус инструмента слева от траектории
G42	Компенсировать радиус инструмента справа от траектории
G90	Задание абсолютных координат опорных точек траектории
G91	Задание координат инкрементально последней введенной опорной точки
M30	Конец программы, со сбросом модальных функций

Далее, когда выясняется тип команды, считываются остальные подстроки, если они есть, и оттуда получается вся остальная технологическая информация. Таким образом, последовательный проход по массиву строк дает на выходе упорядоченный массив Action [4], т.е. внутренний формат данных, с которым теперь можно продолжить работу более гибко.

Воссоздание контуров деталей. Следующим шагом является создание из полученного массива примитивных действий массива контуров деталей на листе. Исходя из того, что на данный момент основными являются гидроабразивные станки, которые не способны прерывать обработку текущего контура, было решено определять начало и конец контура по холостым ходам. Все действия, которые заключены между двумя холостыми ходами, являются законченным контуром.

Каждый контур состоит из геометрических частей и может являться внутренним или внешним. Каждый Action описывает действия инструмента, и в нем достаточно данных, чтобы создать математическую модель геометрического примитива.

В списке Action между двумя холостыми ходами обрабатываются все позиции, и создается экземпляр контура. Такая операция производится с каждым промежутком, и получается массив контуров. Тем не менее, нам не известно, внутренние они или внешние.

Определение вложенности контуров и воссоздание деталей. Среди огромного массива контуров теперь необходимо определить, какие являются внутренними, а какие – внешними. При этом высокая вероятность того, что вложенность может быть достаточно высокой степени, например, как представлено на рис. 1.

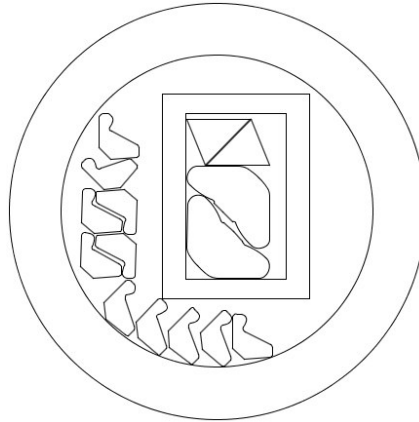


Рис. 1. Пример вложенности деталей

Для решения этой проблемы необходимо создать два массива. В одном найти степень вложенности каждого контура, а во втором для каждого контура определить, какие контуры находятся внутри него.

На рис. 2 представлена схема содержимого массива степени вложенности. Сверху – номер контура, снизу – степень вложенности.

0	1	2	...	n
0	1	1	...	2

Рис. 2. Массив степеней вложенности

0	1,3,6,12
1	3,6,15,16
2	
...	...
n	25

Рис. 3. Ассоциативный массив вложенности

Нетрудно заметить, что все контуры, степень вложенности которых четная, – внешние, а нечетная – внутренние. На рис. 3 представлена схема содержимого ассоциативного массива вложенности.

Слева – номер контура, справа – номера контуров, входящих в него. Затем с помощью этих двух массивов происходят рекурсивный обход всех контуров и связывание их в детали.

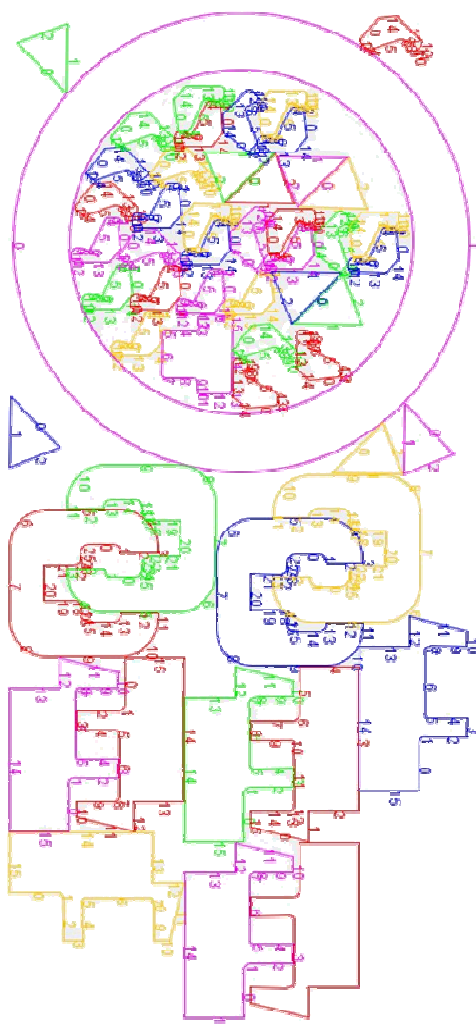


Рис. 4. Результат работы модуля

Берется первый контур из ассоциативного массива, и для всех входящих в него контуров применяется ветвление. Если степень вложенности контура больше текущего на один, то от него функция снова вызывается рекурсивно, и на выходе из рекурсии проверяется следующее условие. Если степень вложенности текущего контура четная, то этот проверяемый контур со степенью вложенности на

1 больше добавляется в массив «дырок». После этого для каждого внешнего контура и всех его «дырок» конструируется деталь.

Демонстрация работы В качестве тестовой программы была выбрана управляющая программа на G-кодах, которая была составлена в самом комплексе Itas-Nesting [5].

В ходе работы модуля была восстановлена карта раскроя, полностью поддающаяся редактированию, она представлена на рис. 4. Для наглядности детали были покрашены в разные цвета и выведены номера примитивов внутри контуров деталей.

Заключение. Таким образом, был разработан модуль по восстановлению карты раскроя из текста управляющей программы. Получен опыт разработки модуля на языке программирования Java. В будущем он будет совершенствоваться для работы в менее щадящих условиях, а именно с УП под разные станки, с заведомо некорректными УП, с УП, реализованными в иных системах координат. Задание на производственную практику полностью выполнено.

Библиографический список

1. Решение задачи ортогональной упаковки листовых материалов методами линейного раскроя / Р.А. Файзрахманов, Р.Т. Мурзакаев, В.С. Шилов, А.С. Мезенцев // Вестник Пермского государственного технического университета. Электротехника, информационные технологии, системы управления. – 2014. – № 10. – С. 29–41.

2. Ловыгин А.А. Современный станок с ЧПУ. CAD-CAM программирование: учеб. пособие. – М: Эльф ИПР, 2006.

3. Мурзакаев Р.Т., Шилов В.С., Брюханова А.А. Программный комплекс фигурного раскроя материала Itas Nesting // Вестник Пермского государственного технического университета. Электротехника, информационные технологии, системы управления. – 2015. – № 13. – С. 15–25.

4. Poliakov A.N., Poluianov D.A. Development of software module for control program for CNC machines forming // Инновационные технологии: теория, инструменты, практика: материалы VI Междунар. интернет-конф. молод. ученых, аспирантов, студентов. – Пермь: Изд-во Перм. нац. исслед. политехн. ун-та, 2014. – С. 284–288.

5. Шилов В.С. Разработка программного комплекса автоматизированного раскроя листового материала // Материалы XI студ. регион. конкурса инновац. проектов по программе «УМНИК»; 27–28 апреля 2015 г. – Пермь, 2015.

ПОИСК УЧАСТКОВ СОВМЕЩЕННОГО РЕЗА В РЕДАКТОРЕ КАРТ РАСКРОЯ

Студенты **В.А. Рутин, А.Р. Шакирзянов, ассистент В.С. Шилов**
Пермский национальный исследовательский
политехнический университет

Рассмотрено применение технологии совмещенного реза при раскрое листового материала. Предложены алгоритмы поиска возможных участков общего реза и смещения деталей в положение, удовлетворяющее технологическим ограничениям общего реза.

Ключевые слова: общий рез, совмещенный рез, редактор карт раскроя, раскрой материала.

SEARCHING OF COMBINED CUT SECTION IN CUTTING LAYOUT EDITOR

Students **V.A. Rutin, A.R. Shakirzyanov, assistant V.S. Shilov**
Perm National Research Polytechnic University

The article deals with the application of combined cut technology when cutting the sheet material. The algorithms of search for possible areas of cut and of offset the total parts in the position of satisfying technological limitations total cut were proposed.

Keywords: combined cut, cutting layout editor, cutting and packing.

В условиях промышленного производства даже небольшая экономия сырья приводит к значительному снижению итоговых затрат [1, 2]. При раскрое листового материала применяются различные методы экономии, в том числе технология совмещенного реза [3]. Использование совмещенного реза позволяет одновременно вырезать два геометрических объекта по смежной стороне, что сокращает используемую площадь и приводит к экономии материала. Применение данной технологии требует строгого соблюдения технологических ограничений. Рассмотрим решение задачи нахождения участков общего реза на примере редактора карт раскроя программного комплекса «Itas Nesting» [4].

Целью статьи является описание разработки класса, позволяющего пользователю задать участок общего реза для двух деталей.

Выделим следующие задачи:

– проверка допустимости использования технологии совмещенного реза для двух деталей при перемещении указателя мыши;

– сдвиг и группировка деталей, содержащих участок общего реза, при нажатии на кнопку мыши.

Описание деталей и технологии общего реза. Для выполнения поставленной задачи будем отталкиваться от того, что все детали состоят из трёх примитивов: отрезок, дуга, окружность (рис. 1) [5].

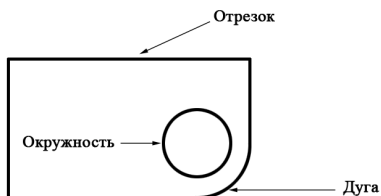


Рис. 1. Пример детали с указанием геометрических примитивов

Технология совмещенного (общего) реза заключается в группировке двух деталей, которые имеют хотя бы одну общую сторону [2]. Общей стороной будем считать отрезок. Так как между двумя деталями имеется линия общего реза, то отсутствует необходимость прорезать этот участок дважды. Участок общего реза позволяет размещать детали с припуском на рез, равным одной ширине реза, что экономит материал и сокращает общий путь реза.

На рис. 2 возможный участок общего реза для двух прямоугольных деталей обозначен пунктирной линией красного цвета. Обозначим условия, при которых применим совмещенный рез:

- 1) два каких-либо отрезка параллельны;
- 2) длины отрезков равны;
- 3) детали выровнены.

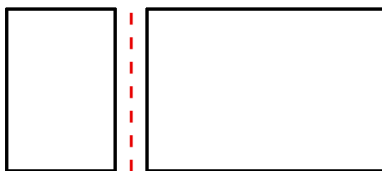


Рис. 2. Участок общего реза для двух прямоугольных деталей

В соответствии с перечисленными требованиями опишем алгоритм поиска участков общего реза. Общий рез будем искать для отрезков выпуклых деталей.

Алгоритм определения применимости технологии совмещенного реза. Определять возможность использования общего реза для деталей будем при перемещении указателя мыши. Согласно архитектуре редактора при каждом движении курсора вызывается метод `MouseMove` [5], следовательно, основные вычисления будем производить в этом методе. Заведем дополнительные поля класса для хранения информации о последних найденных участках совмещенного реза. Разобьем задачу на подзадачи:

1. Найти отрезки деталей, которые пользователь желает сгруппировать.
2. Выбрать параллельные отрезки одинаковой длины.
3. Сообщить пользователю о возможности или невозможности применения технологии, т.е. покрасить детали в соответствующий цвет.

Для решения первой задачи найдем список отрезков, рядом с которыми находится курсор мыши. Построим вокруг курсора мыши окружность с определенным радиусом обойдем все отрезки на карте раскроя. Если при этом минимальное расстояние от отрезка до точки меньше или равно отношению радиуса к масштабу, то добавим отрезок в требуемый список.

Среди найденных на предыдущем шаге отрезков выберем два параллельных одинаковой длины. Используя полный перебор, проверим выполнимость параллельности отрезков и равенства их длин [6]. Будем считать, что длины отрезков одинаковы, если

$$\text{abs}(a - b) \leq \text{eps}, \quad (1)$$

где a , b – длины отрезков, eps – относительная погрешность. Сравнивать числа с плавающей точкой через eps необходимо, чтобы исключить влияние погрешности измерений на ответ, например, когда длины деталей отличаются на крайне малое число порядка 10^{-5} мм. Если существуют два параллельных отрезка одинаковой длины, то запишем информацию о них в соответствующие поля класса, иначе обнулیم эти поля. Если найдены два отрезка, подходящие под описанные выше ограничения, покрасим соответствующие детали в зеленый цвет. Таким образом, пользователь будет знать, что детали можно сгруппировать и применить к ним технологию общего реза.

Заметим, что метод `MouseMove` вызывается даже при небольшом движении мыши, соответственно, перед каждым поиском возможных участков совмещенного реза обнулیم поля класса, в которых содержится информация о ранее найденных участках. Блок-схема алгоритма представлена на рис. 3.

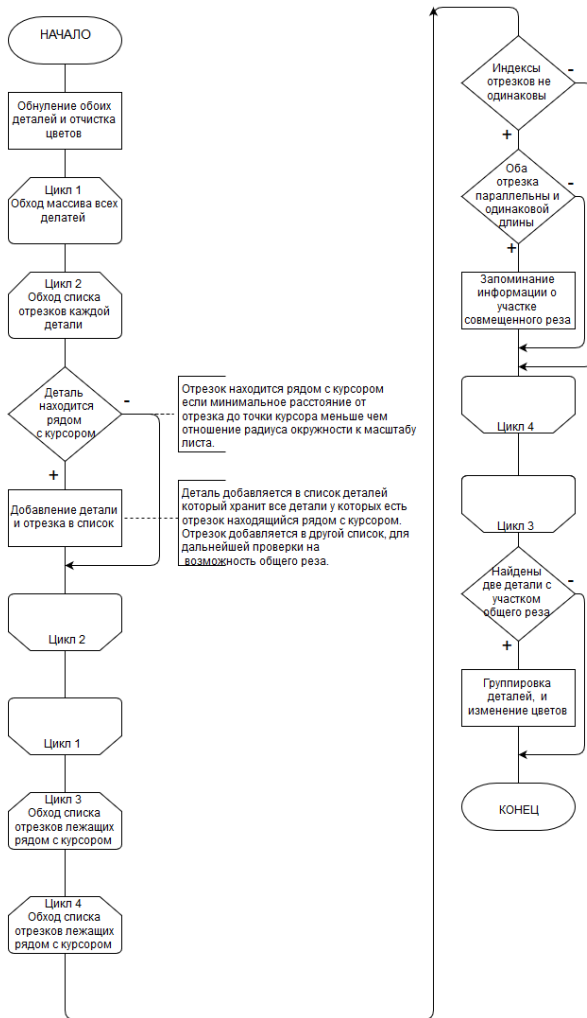


Рис. 3. Блок-схема алгоритма поиска двух деталей, для которых допустим общий рез

Алгоритм перемещения и группировки двух деталей, имеющих участок общего реза. Перед тем, как использовать совмещенный рез, необходимо сдвинуть детали таким образом, чтобы их взаимное расположение удовлетворяло технологическим ограничениям [2]. Перемещать детали и объединять их в группу будем при нажатии левой

кнопки мыши. Если во время клика в полях класса установлены значения о найденных отрезках, через которые можно провести общий рез, сдвинем детали в корректное положение, иначе – закончим выполнение алгоритма.

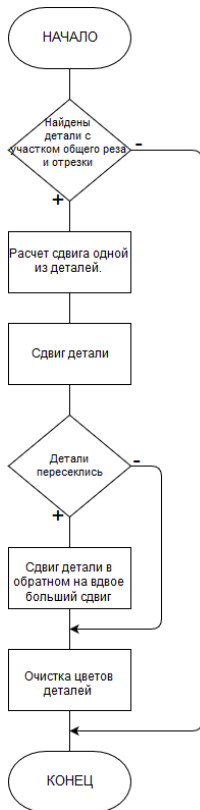


Рис. 4. Блок-схема алгоритма перемещения двух деталей, для которых допустим общий рез

Будем перемещать одну из деталей таким образом, чтобы между отрезками, которые являются участком общего реза, оставалось расстояние, равное одной ширине реза. Опишем формулы сдвига детали в декартовой системе координат по осям X и Y :

$$\text{shift}X = \text{dist} - \text{gap} \cdot \sin\alpha, \quad (2)$$

$$\text{shift}Y = \text{dist} - \text{gap} \cdot \cos\alpha, \quad (3)$$

где $\text{shift}X$ и $\text{shift}Y$ – сдвиг деталей по осям X и Y , dist – минимальное расстояние между деталями, gap – ширина реза, α – угол между отрезками и осью X . Сдвинем одну из деталей на рассчитанный вектор. При этом если детали пересеклись, то вернем перемещаемую деталь в первоначальное расположение и сместим ее на вектор, обратный данному. Блок-схема алгоритма представлена на рис. 4.

Заключение. Рассмотрена реализация класса для работы с технологией совмещенного реза. Предложены алгоритм определения допустимости применения общего реза и алгоритм смещения деталей в положение, при котором соблюдаются технологические ограничения. Предложенные алгоритмы использованы в редакторе карт раскроя программного комплекса «Itas Nesting».

Библиографический список

1. Файзрахманов Р.А., Архипов А.В. Проектирование автоматизированных информационных систем на основе объектно-ориентированного подхода: учеб. пособ. – Пермь: Изд-во Перм. гос. техн. ун-та, 2011. – 222 с.

2. Romanova T., Stoyan Y., Pankratov A. Mathematical models and solution algorithm for nesting problem of arbitrary shaped objects // 8th Conference of the special interest group on cutting and packing (ESICUP). – Copenhagen, Denmark, 2011.

3. Murzakaev R.T., Shilov V.S., Mezentsev A.S. On the nesting layout in view on technical gaps for joint cutting // Russian electrical engineering. – 2014. – № 11. – С. 668–672.

4. Мурзакаев Р.Т., Шилов В.С., Брюханова А.А. Программный комплекс фигурного раскроя листового материала Itas Nesting // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2015. – № 13. – С. 15–25.

5. Рутин В.А. Алгоритмы функционирования редактора карт раскроя // Инновационные технологии: теория, инструменты, практика. – Пермь: Изд-во Перм. нац. исслед. политехн. ун-та, 2014. – № 1. – С. 322–328.

6. De Berg M. Computational geometry: Algorithms and Applications // Springer Berlin Heidelberg. – 2008. – С. 219–241.

ИСПОЛЬЗОВАНИЕ АКТИВНЫХ МЕТОДОВ ОБУЧЕНИЯ ПРИ РАЗРАБОТКЕ ИНТЕРАКТИВНЫХ ОБУЧАЮЩИХ СИСТЕМ

Курсант Ю.В. Собакина, преподаватель А.В. Набатов
Пермский военный институт внутренних войск МВД России

В статье описаны преимущества применения активных методов обучения и возможность их реализации с помощью интерактивных обучающих систем.

Ключевые слова: активный метод обучения, метод кейсов, интерфейс, мультимедийное приложение.

USE OF ACTIVE METHODS OF TRAINING WHEN DEVELOPING OF THE INTERACTIVE TRAINING SYSTEMS

Cadet Y.V. Sobakina, Teacher A.V. Nabatov
Perm Military Institute of Internal Troops of the Ministry
of Internal Affairs of Russia

In this article advantages of application of active methods of training and possibility of their realization by means of the interactive training systems are described.

Keywords: active method of training, method of cases, interface, multimedia application.

Современный человек уже не мыслит свое существование без техники. Будь он школьник, студент (курсант) либо служащий небольшого предприятия, каждому из них требуются результаты научных разработок. В связи с этим изменяется и подход к обучению молодежи сегодня.

При изучении дисциплин, а также при приеме на работу нового сотрудника педагога и работодателя все чаще стали использовать более современные методы обучения и тестирования знаний. Данные методы позволяют активизировать учебный процесс, побудить обучаемых к творческому участию в нем. Задача таких методов обучения заключается в обеспечении развития и саморазвития личности, при этом используя его индивидуальные особенности и способности. Особое место в таких процессах отводится теоретическому мышлению. Оно, как правило, больше используется при первом типе обучения, который включает: проблемные лекции, проблемно-активные практические занятия и лабораторные работы, семинары и дискуссии, курсовое и дипломное проектирование, практику, стажировку.

Однако все они ориентированы на самостоятельную деятельность обучаемого, и в них отсутствует имитация реальных обстоятельств в условной ситуации [1–2]. Для этого свое место в процессе обучения нашли активные методы обучения второго типа, они же имитационные. Их подразделяют на игровые и неигровые. Это методы, за счет которых происходит моделирование реальных объектов и ситуаций, поэтому они имеют высокую эффективность в учебном процессе, и сегодня их применение практикуется в ведущих учебных заведениях всего мира.

Таким образом, активные методы обучения:

- способствуют вовлечению обучаемых в решение проблем, максимально относящихся к их профессиональной деятельности;
- совершенствуют практические навыки и умения любого заинтересованного лица в этой сфере деятельности путем расширения и углубления профессиональных знаний;
- развивают мышления обучаемых.

Одним из них является метод кейсов, который основывается на решении конкретных задач (ситуаций). Здесь обучаемый должен проанализировать ситуацию, разобраться в сути проблем, предложить варианты возможных решений и выбрать лучшее из них. Успех такого метода обучения будет зависеть от следующих критериев:

- достаточный объем информации;
- участие заинтересованных (владеющих достоверной информацией) лиц в разработке метода для конкретной ситуации;
- наличие захватывающей ситуации, позволяющей применить разнообразные методы анализа при поиске решения.

Метод кейсов может быть представлен как в мультимедийном, так и в печатном варианте.

Печатный вариант может быть объемом от нескольких предложений до множества страниц. При этом следует иметь в виду, что большие кейсы вызывают у обучаемых некоторые затруднения по сравнению с малыми, особенно при работе впервые. Поэтому информацию необходимо обязательно дополнять наглядным материалом в виде рисунков (фотографий), таблиц, различного рода диаграмм (гистограмм). Исходя из опыта, можно сказать, что с печатной информацией или информацией, представленной на электронных носителях, работать гораздо легче, чем с аудио- или видеовариантами.

В последнее время все популярнее становятся мультимедийные представления кейсов, которые позволяют избежать вышеназванных трудностей, сочетая в себе преимущества текстовой информации и интерактивного видеоизображения [3–4].

При разработке мультимедийного приложения необходимо помнить о пользовательском интерфейсе, который обеспечивает взаимодействие пользователя с программой. Сама программа не должна быть перегружена лишней информацией. Как правило, она должна состоять из описательной части, которая включает необходимую информацию об изучаемой дисциплине (объекте, процессе) и соответственно практическое задание с набором необходимых документов для его решения, а также средства контроля знаний. Помимо этого в приложения включается и вспомогательная информация для более углубленного изучения.

Исходные данные, для построения таких систем берутся из литературных источников, Интернета, руководств по эксплуатации объектов или систем и т.д. Весь учебный материал представляется обучаемому в виде электронных документов, которые могут быть просмотрены с помощью различных программ в зависимости от того, как и с помощью чего данное приложение разрабатывалось. На сегодняшний день для разработки графических и мультимедийных данных используются специализированные программы, такие как Autodesk Maya, Autodesk 3ds Max, Autodesk Softimage и многие другие.

При разработке интерактивных обучающих систем нужно учитывать реальные функциональные возможности системы с требованиями заказчика и видами задач, которые эта система будет позволять решать.

В конечном итоге такое программное обеспечение можно будет использовать:

- для теоретической подготовки будущих специалистов, обслуживающего персонала;
- наглядного отображения теоретического и практического материала;
- проведению начального и основного контроля;
- отображения в наглядной статистике результатов обучения.

Использование такого метода обучения позволяет решить ряд важнейших задач. Во-первых, это уменьшение времени на поиск соответствующего объема информации, во-вторых, обучаемый учится

самостоятельно принимать обоснованные решения для сложных задач, рассматривая их с разных сторон.

Активные методы (особенно кейс метод) обучения имеют огромные перспективы в профессиональной подготовке как будущих специалистов, так и уже обученного персонала в технической, экономической и других областях науки.

Библиографический список

1. Андюсов Б.Е. Кейс-метод как инструмент формирования компетентностей // Директор школы. – 2010. – № 4. – С. 61–69.

2. Набатов А.В. Метод Парето при реализации метода кейсов в технологии обучения в технических вузах // Педагогическое мастерство и педагогические технологии: материалы V Междунар. науч.-практ. конф.; Чебоксары, 20 сентября 2015 г. / ред.: О.Н. Широков [и др.]. – Чебоксары: Интерактив плюс, 2015. – № 3(5). – С. 100–101.

3. Федоринова З.В. Использование метода case-study для гуманитаризации образования в техническом вузе // Вестник Томск. гос. пед. ун-та. – 2012. – Вып. 4. – С. 90–92.

4. Кейс-метод. Окно в мир ситуационной методики обучения (case-study). – URL: <http://www.casemethod.ru> (дата обращения: 05.10.2015).

ПРОВЕРКА ВАЛИДНОСТИ УПРАВЛЯЮЩЕЙ ПРОГРАММЫ

Студенты Н.В. Соловьева, Н.А. Синицын, ассистент В.С. Шилов
Пермский национальный исследовательский
политехнический университет

Статья посвящена методикам проверки валидности управляющей программы. Рассмотрены ошибки, встречающиеся при написании управляющей программы, и методы их решения.

Ключевые слова: G-код, ЧПУ, валидность, УП.

CHECKING THE VALIDITY OF CONTROL PROGRAM

Students N.V. Solovyeva, N.A. Sinitsyn, assistant V.S. Shilov
Perm National Research Polytechnic University

The paper deals with checking the validity of control program. The paper discusses the errors that occur while writing a control program, and recommended solutions.

Keywords: G-code, CNC, validity, CP.

Задача раскрытия материала включает множество подзадач [1, 2], одной из них является имитация резки [3]. Имитация резки представляет собой компьютерную модель, отображающую весь процесс резки станка с числовым программным управлением (ЧПУ). Она основывается на построении путей обхода режущего инструмента, которые описываются программно с использованием G-кода.

G-код – общее название языка программирования, регламентированного стандартом ISO 6983-1:1982 и ГОСТ 20999-83. В технической литературе, изданной в Советском Союзе, G-код именуется как код ИСО 7-бит (ISO 7-bit) [4].

Текст программы для станка с ЧПУ заключается в операторные скобки «%», и уже в них прописывается код программы. Обычно в самом начале пишется название программы, следующее после символа «О». Управляющая программа, написанная с помощью G-кода, записывается в виде последовательных кадров. В общем случае команды в кадре могут следовать в произвольном порядке, но для удобства прочтения программы рекомендуется записывать команды в следующем порядке:

- подготовительные команды;
- команды управления перемещением;
- команды выбора режимов обработки материалов;
- технологические команды.

Несмотря на то, что управляющая программа создается на стадии построения пути резки, у оператора также может быть возможность вносить изменения и писать УП. При этом необходимо учитывать, что даже человек с многолетним опытом работы не застрахован от ошибок.

Существует несколько типов ошибок, наиболее часто встречающихся в УП:

- перемещение реза в рабочем режиме;
- попытка резки при выключенном станке;
- начало резки при отсутствии листа;
- синтаксические ошибки.

Наличие подобных ошибок может привести не только к браку деталей, которые могут выполняться из дорогостоящего материала, но и к поломке реза, что, в свою очередь, повлечет за собой приостановку работы, а также дорогостоящий ремонт оборудования.

В связи с этим целесообразно проводить проверку валидности кода управляющей программы на соответствие общепринятым стандартам и основным технологиям резки перед началом резки [5].

Таким образом, целью работы является разработка методики проверки управляющей программы на ошибки.

Существует несколько способов ликвидации ошибок в управляющей программе до непосредственного начала процесса резки:

1. Оператор может после написания программы провести проверку написанного текста вручную. Но, во-первых, не каждый оператор будет проверять им же написанную программу. Во-вторых, даже если оператор проверит код, это не дает стопроцентной гарантии того, что ошибки будут найдены.

2. Имитация резки – визуальное отображение процесса резки со всеми производимыми переходами реза посредством специального программного обеспечения. Данный способ так же не исключает возможности ошибок, так как оператор по каким-либо причинам может обойти данный этап.

3. Автоматическая проверка при отправке программы на станок на наличие основных ошибок. Данное решение требует разработки программного модуля, обладающего достаточно сложной логикой работы, однако такой подход гарантирует осуществление валидации программы.

Выделим основные проверки, которые должны производиться в модуле валидации УП:

- если в детали имеются отверстия, в первую очередь производится резка именно этих отверстий;
- детали, находящиеся в отверстиях других деталей, должны вырезаться в первую очередь;
- при определенном расположении деталей, после обработки нескольких деталей часть листа может отпасть. В такой ситуации необходимо учитывать, чтобы на этой части не оставалось необработанных деталей. Пример такого расположения деталей изображен на рисунке. Если последовательность резки будет выполняться в порядке, представленном на рисунке, то деталь под номером 5 будет испорчена.

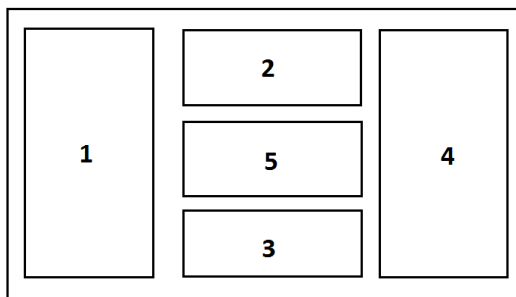


Рис. Пример неудачного расположения деталей при формировании карты раскроя

Таким образом, существуют различные способы, позволяющие осуществить проверку управляющей программы на валидность до начала процесса резки, что, в свою очередь, может сократить количество бракованных изделий и увеличить скорость выполнения обработки деталей. Наиболее эффективной представляется автоматическая проверка перед отправкой программы на станок. В данной статье были описаны только общие для всех технологий резки ошибки.

Библиографический список

1. Решение задачи ортогональной упаковки листовых материалов методами линейного раскроя / Р.А. Файзрахманов, Р.Т. Мурзакаев, В.С. Шилов, А.С. Мезенцев // Вестник Пермского государственного технического университета. Электротехника, информационные технологии, системы управления. – 2014. – № 10. – С. 29–41.
2. Исследование бизнес-процесса учета делового остатка при раскрое листовых материалов / Р.А. Файзрахманов, Р.Т. Мурзакаев, В.С. Шилов, А.В. Буркова // Вестник Пермского государственного технического университета. Электротехника, информационные технологии, системы управления. – 2013. – № 7. – С. 143–148.
3. Леонов Е.Р., Казанцев А.С., Мурзакаев Р.Т. Разработка имитатора резки станка с ЧПУ // Инновационные технологии: теория, инструменты, практика. – Пермь: Изд-во Перм. нац. исслед. политехн. ун-та, 2014. – № 2. – С. 245–249.
4. Intuwiz Software. G-код – основные понятия [Электронный ресурс]. – URL: <http://www.intuwiz.ru> (дата обращения: 26.11.2015).
5. Сертиком. Отличие валидации от верификации [Электронный ресурс]. – URL: <http://www.smk56.ru> (дата обращения: 26.11.2015).

ВОПРОСЫ БИОМЕТРИЧЕСКОЙ АУТЕНТИФИКАЦИИ В ОПЕРАЦИОННЫХ СИСТЕМАХ WINDOWS И LINUX

**Студенты П.А. Фефилова, Р.А. Андреев,
канд. техн. наук, доцент А.Н. Кокоулин**
Пермского национального исследовательского
политехнического университета

В статье исследованы архитектура и возможности современных систем биометрической аутентификации и сканеров отпечатков пальцев. Рассмотрены вопросы интеграции систем биометрической аутентификации в подсистему безопасности операционных систем Windows и Linux.

Ключевые слова: биометрические сканеры отпечатков пальцев, Windows, Linux.

ASPECTS OF BIOMETRIC AUTHENTICATION IN OPERATION SYSTEMS WINDOWS AND LINUX

**Students P.A. Feofilova, R.A. Andreev,
Ph.D. in Technical Sciences, Associate Professor A.N. Kokoulin**
Perm National Research Polytechnic University

This article reviews architecture and capabilities of modern biometric authentication systems and fingerprint scanners. Aspects of integration biometric authentication systems into security subsystem of OS Windows and Linux are considered.

Keywords: Biometric fingerprint scanners, Windows, Linux.

Сканеры отпечатков пальцев можно разделить на три группы по используемым ими физическим принципам:

- оптические;
- кремниевые;
- ультразвуковые.

Для пользователя отличие данных групп состоит лишь в том, какие действия необходимо производить со сканером: прикладывать палец (оптический и ёмкостной) или проводить им по сенсору (тепловой).

Идентификация по отпечаткам пальцев основана на распознавании образа, когда папиллярные узоры сравниваются с зарегистрированными данными. Процесс идентификации выполняется в три этапа:

1. Формируется изображение отпечатка пальца. Захват изображения может производиться с помощью встроенной камеры считыва-

теля либо с помощью регистрации разности потенциалов электрического поля между бугорками и впадинами папиллярного узора. Возможно применение комбинаций методов. В результате получается цифровой черно-белый снимок узоров отпечатка пальца.

2. Изображение отпечатка пальца преобразуется в математическую модель, в которой уникальные признаки, такие как дуги, завитки, петли и расстояния между ними, сохраняются в виде цифрового кода.

3. Производится сравнение идентифицируемой цифровой модели с шаблонами в базе данных и выполняется поиск соответствий.

Процедуры биометрической аутентификации позволяют аутентифицировать пользователя путем измерения его физиологических параметров и характеристик, которые чрезвычайно трудно подделать [1].

Основные достоинства биометрических методов:

- высокая степень достоверности аутентификации по биометрическим признакам (из-за их уникальности);
- неотделимость биометрических признаков от дееспособной личности;
- трудность фальсификации биометрических признаков. Помимо отпечатков пальцев используются такие биометрические признаки, как геометрическая форма кисти руки, форма и размеры лица, особенности голоса, узор радужной оболочки и сетчатки глаз.

Но как все это работает в операционных системах? И что можно сделать еще при помощи сканера отпечатков пальца?

Как в Windows, так и в Linux при помощи сканера можно войти в систему, т.е. отпечаток пальца и есть ваш пароль (рис. 1). В ОС Windows XP за это отвечала GINA. GINA или Graphical Identification and Authentication dll отвечает за пользовательский интерфейс входа в систему в Windows NT. GINA загружается и используется процессом Winlogon, отвечающим за аутентификацию пользователей. Именно она ответственна за выдачу окошек с надписями «Нажмите CTRL-ALT-DEL» и т.п. В большинстве случаев пользователю вполне достаточно возможностей стандартной реализации GINA – msgina.dll, но иногда возникают ситуации, когда нужно реализовать собственный алгоритм входа в систему или выполнять блокировку станции при возникновении определенного события. Именно для таких случаев программисты Microsoft оставили возможность написания собственной GINA DLL. Однако написание собственной GINA с нуля –

очень трудоемкая задача. Гораздо проще написать DLL, экспортирующую нужные по стандарту GINA функции, большинство из которых являются просто обертками над функциями уже установленной ранее GINA (например, стандартной msgina.dll).

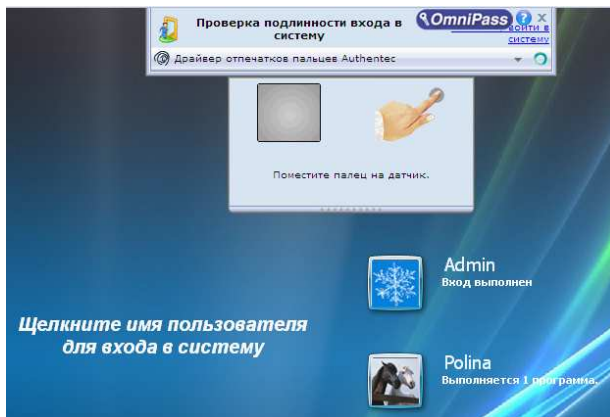


Рис. 1. Пример входа в Windows XP по отпечатку пальца

Но в системах, начиная с Windows Vista, GINA больше не применяется. Теперь технология интеграция с WinLogon осуществляется с помощью Credential Provider. Credential Provider (поставщик учётных данных) – основной механизм для аутентификации пользователя, в настоящее время единственный метод для пользователей, который нужен для проверки подлинности. Данный механизм применяется для входа в систему и других сценариев проверки подлинности. Поставщик учётных данных дает возможность входа в систему (например, по отпечатку пальца). В Windows10 механизм провайдеров учетных записей более важен, чем когда-либо; он будет использоваться для аутентификации в приложения, веб-сайты и т.д. Также в Windows 10 добавлена новая функция Windows Hello, которая является частью Windows 10 и предназначена для входа на ваши устройства, в приложения, веб-службы и сети. Windows Hello работает с технологией учетных данных, называемой Microsoft Passport, – более простым, удобным и безопасным способом, чем использование пароля, так как в этом случае используется «биометрическая проверка подлинности», т.е. вы входите через распознавание лица, радужной оболочки глаз или отпечатка пальца (либо с помощью PIN-кода).

Можно использовать 2 типа механизма Credential Provider's: «system credential providers» и «third-party credential providers» (рис. 2). В системные провайдеры учетных записей входят механизмы, уже встроенные в Windows (PIN, отпечатки пальцев, распознавание по лицу и радужной оболочке). Под неоригинальными провайдерами учетных записей подразумеваются написанные собственноручно инструкции, которые легко можно интегрировать в систему [2].

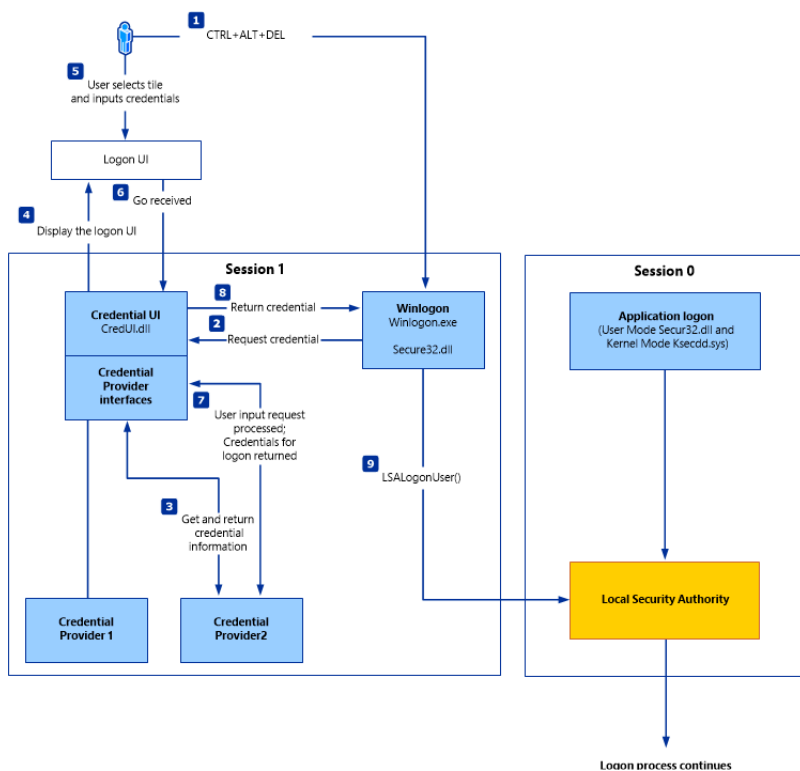


Рис. 2. Схема работы Credential Provider

Кроме этого, в Windows есть специальная служба для взаимодействия с биометрическими устройствами – Windows Biometrics Framework. Теперь нет необходимости использовать специальное программное обеспечение, поставляемое производителями биометрических устройств. Все операции по взаимодействию возьмет на себя служба Windows Biometric Service. В составе операционной системы присутствует

специальное приложение, позволяющее управлять процессом считывания отпечатков пальцев и устанавливать соответствующие политики безопасности. Так, для каждого пользователя можно закрепить «определенный палец». Успешное считывание отпечатка будет предоставлять пользователю вход в Windows или домен (рис. 3).

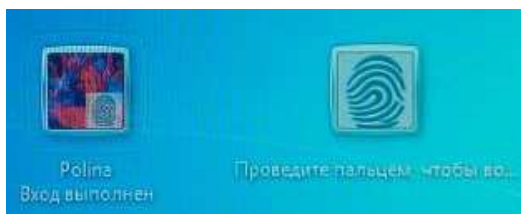


Рис. 3. Пример входа в Windows 7 по отпечатку пальца

Все возможности платформы доступны разработчикам в виде набора API-функций. Для многих это будет приятным сюрпризом, так как создавать приложения для взаимодействия с устройствами станет намного проще, и, самое главное, есть все шансы добиться универсальности. Поскольку работа с устройством происходит через посредника (Windows Biometric Service), у программиста отпадет необходимость заботиться о поддержке устройств определенных производителей. Главное, чтобы Windows могла работать с ними. Стоит заметить, что работа с устройством не напрямую, а через службу обеспечивает, как минимум, еще один плюс – безопасность. Клиентское приложение не имеет прямого доступа к устройству, а если это так, то и вероятность изменить конфиденциальные данные существенно снижается.

В ОС Linux, к сожалению, нет никакой поддержки для сканеров отпечатка пальца в ядре. Соответственно, драйверов для любого подобного устройства под Linux не существует. Но все же есть способ использовать некоторые модели устройств с помощью `sys-auth/libfprint`. Это библиотека, разработанная в Манчестерском университете, которая работает со сканером, не используя никаких специальных модулей ядра, через `libusb`. Все основные функции для работы с устройствами унифицированы, и для каждого поддерживаемого типа устройства в библиотеке существует собственный модуль, который разработчики называют «драйвером». Для определения подлинности отпечатка пальца разработчики библиотеки создали алгоритм, подробности которого автору неизвестны,

но суть примерно такова: изображение преобразуется в векторный формат, потом на преобразованном изображении выделяются точки, характеризующие отпечаток пальца. При проверке происходят такие же преобразование и сравнение полученных точек с сохраненным образцом. Со списком поддерживаемых в настоящий момент моделей устройств можно ознакомиться на сайте проекта libfprint (<http://www.freedesktop.org/wiki/Software/fprint/>).

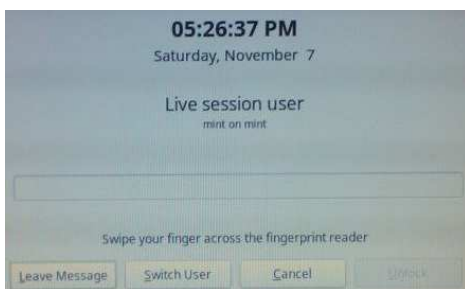


Рис. 4. Пример входа в Linux Mint по отпечатку пальца

Библиотека libfprint сама по себе не имеет никаких функций авторизации пользователей, для обеспечения такой возможности были разработаны несколько модулей PAM, использующих эту библиотеку. В состав пакета, кроме модуля PAM, входит программа для сохранения и проверки образцов отпечатков [3].

На сегодняшний день ОС Windows гораздо лучше адаптированы для использования биометрических сканеров отпечатков пальцев, чем ОС Linux. В Linux отпечаток пальца – это только ваш пароль, а в системах Windows при помощи специальных программных продуктов и отпечатка можно совершать различные действия (смена пользователя, шифрование данных, запуск приложений и т.д.). При этом далеко не все сканеры будут работать в системе Linux.

Библиографический список

1. Задорожный В. Идентификация по отпечаткам пальцев // PC Magazine/Russian Edition. – 2004. – № 1. – С. 114–119.
2. TechNet [Электронный ресурс]. – URL: [http://technet.microsoft.com/en-us/library/dn169014\(v=ws.10\).aspx](http://technet.microsoft.com/en-us/library/dn169014(v=ws.10).aspx) (дата обращения: 25.11.2015).
3. Archlinux [Электронный ресурс]. – URL: <https://wiki.archlinux.org/index.php/Fprint> (дата обращения: 25.11.2015).

МОДЕЛИРОВАНИЕ КОЛЕБАНИЙ ТРОСА ДЛЯ ТРЕНАЖЕРА ПОРТАЛЬНОГО КРАНА

Старший преподаватель А.Ф. Хабибулин,
магистрант Ф.И. Шкляев

Пермский национальный исследовательский
политехнический университет

В статье приведен алгоритм моделирования троса в компьютерном тренажере крановщика портального крана с использованием интегрирования Верле. Представлен расчет модели троса для каждого шага итерации.

Ключевые слова: моделирование троса, компьютерный тренажер, интегрирование Верле, портальный кран, колебания.

MODELING OF CABLE OSCILLATIONS FOR GANTRY CRANE SIMULATOR

Senior Lecturer A.F. Khabibulin,
Master Student P.I. Shklyayev

Perm National Research Polytechnic University

In this article presented the computer simulation algorithm of a cable of gantry crane simulator using Verlet integration. The calculations of the model cable for each iteration step are presented.

Keywords: cable modeling, simulator, Verlet integration, gantry crane, oscillations.

Введение. Использование компьютерных тренажеров – один из основных инструментов при обучении операторов перегрузочных машин. Портальный кран является разновидностью сложных перегрузочных машин. Цель использования тренажера портального крана состоит в подготовке операторов к реальным условиям работы, поэтому компьютерная модель должна максимально точно отражать реальность. Одним из физических процессов, который необходимо моделировать в тренажерном комплексе оператора портального крана [1], является колебание системы «трос – груз». Для решения данной задачи было решено разбить трос на материальные точки и использовать метод интегрирования Верле.

Методика моделирования связанных материальных точек. Метод интегрирования Верле – численный метод, используемый для интегрирования уравнений движения материальной точки [2]. Часто используется для вычисления траекторий частиц в моделях молекулярной динамики [3] и в компьютерных играх [4].

Интегрирование Верле используется для вычисления будущей позиции точки по текущей и прошлой позициям без использования скорости:

$$\vec{x}(t + \Delta t) = 2\vec{x}(t) - \vec{x}(t - \Delta t) + \vec{a}(t)\Delta t^2 + O(\Delta t^4), \quad (1)$$

где \vec{x} – радиус-вектор; \vec{a} – ускорение; $O(\Delta t^4)$ – ошибка относительно Δt^4 .

При моделировании связанных между собой точек с заданным расстоянием между ними используются ограничения двух видов: жесткие и упругие.

Под жесткими ограничениями понимается абсолютно нерастяжимый стержень с постоянной длиной между двумя точками (рис. 1), условие связи:

$$|\vec{x}_2 - \vec{x}_1| = L_1^* = \text{const}, \quad (2)$$

где \vec{x}_2, \vec{x}_1 – радиус-векторы; L_1^* – заданное расстояние между точками.

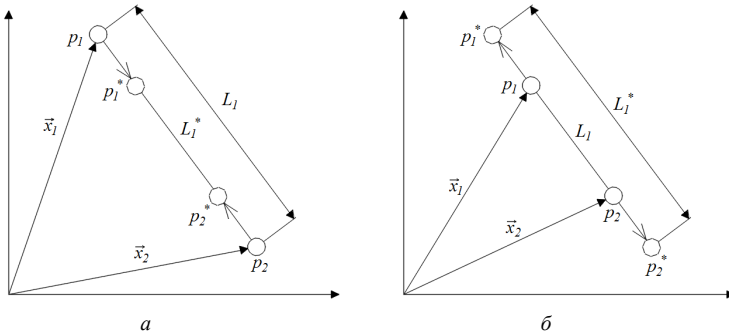


Рис. 1. Накладывание жестких ограничений на расстояние между точками:

L_1^* – заданное расстояние между точками; L_1 – текущее расстояние

между точками: $a - L_1 > L_1^*$; $b - L_1 < L_1^*$

Условие (2) выполняется путем следующих вычислений: находим вектор \vec{x}_{12} между двумя точками (3), вычисляем его фактическую длину (4), относительную разность между текущей и требуемой длиной вектора (5) и величину смещения (6):

$$\vec{x}_{12} = \vec{x}_2 - \vec{x}_1, \quad (3)$$

$$L = |\vec{x}_{12}|, \quad (4)$$

$$df = \frac{L - L^*}{L}, \quad (5)$$

$$d\vec{l} = df \cdot \vec{x}_{12}. \quad (6)$$

Новые положения точек p_1 и p_2 высчитываются по формулам (7) и (8):

$$\vec{x}_1^* = \vec{x}_1 + \frac{d\vec{l}}{2}, \quad (7)$$

$$\vec{x}_2^* = \vec{x}_2 - \frac{d\vec{l}}{2}, \quad (8)$$

где \vec{x}_1^*, \vec{x}_2^* – новые векторы положения частиц; $d\vec{l}/2$ – приращение к старому положению точки.

Под упругой связью понимается растяжимый стержень с определенной длиной и наличием массы у частиц (рис. 2). Данный тип ограничений описывается теми же формулами, что и жесткие ограничения, но с некоторыми поправками (применяем соотношение масс), – изменяются формулы (6), (7) и (8).

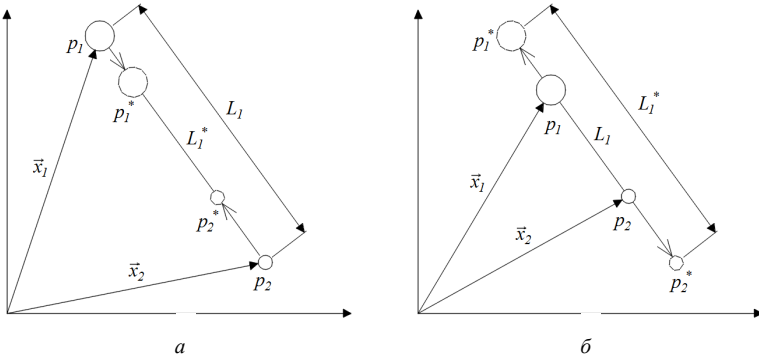


Рис. 2. Накладывание упругих ограничений на расстояние между точками,

L_1^* – заданное расстояние между точками; L_1 – текущее расстояние

между точками: $a - L_1 > L_1^*$; $b - L_1 < L_1^*$

$$d\vec{l}_1 = df \cdot \vec{x}_{12} \frac{m_2}{m_1 + m_2}, \quad (9)$$

$$d\vec{l}_2 = df \cdot \vec{x}_{12} \frac{m_1}{m_1 + m_2}, \quad (10)$$

$$\vec{x}_1^* = \vec{x}_1 + d\vec{l}_1, \quad (11)$$

$$\vec{x}_2^* = \vec{x}_2 - d\vec{l}_2, \quad (12)$$

где m_1, m_2 – массы первой и второй материальных точек соответственно.

Как видно из формул (9) и (10), соотношение масс влияет на приращение к старому положению точки. Чем больше масса одной точки по отношению к другой, тем меньше её приращение.

Таким образом, при моделировании троса он делится на множество материальных точек. После применения сил ко всем точкам находятся их новые положения по формуле (1), после чего применяются ограничения на расстояния между ними.

Моделирование колебаний троса с использованием данного метода:

1) Разобьём трос на три материальные точки, соединенные между собой (рис. 3).

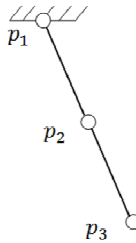


Рис. 3. Представление моделируемого троса в виде системы материальных точек (точка p_1 закреплена)

Каждая частица описывается уравнением движения материальной точки (рис. 4): $\ddot{\vec{x}} = \vec{a}(\vec{x}, t)$.

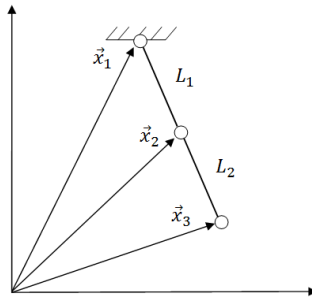


Рис. 4. Представление системы частиц в векторном виде,
 $\vec{x}_1, \vec{x}_2, \vec{x}_3$ – радиус-вектор первой, второй и третьей частиц соответственно,
 L_1, L_2 – расстояние между точками p_1-p_2 и p_2-p_3 соответственно

2) Вычисляем новые радиус-векторы по формуле (1) для всех точек, задействованных в движении (рис. 5). В данном случае это точки p_2 и p_3 (точка p_1 является точкой крепления троса). В формуле (1) за $\vec{a}(t)$ берем ускорение свободного падения, так как в данном случае на трос действует только сила тяжести.

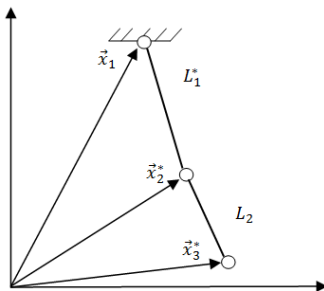


Рис. 5. Система частиц после первого шага алгоритма

3) Применяем ограничение для связи между частицами p_1 и p_2 .

Поскольку при применении жестких ограничений масса частиц не учитывается, то такой тип ограничений не подходит для моделирования колебаний троса, так как инертность троса и подцепленного к нему груза важна в обучении операторов. Для решения поставленной задачи будем использовать упругие ограничения. Величина смещения находится по формулам (3)–(5), (9)–(12). В связи p_1 – p_2 происходит смещение только точки p_2 (рис. 6), поскольку точка p_1 является точкой крепления. Вектор смещения $d\vec{l}_2$ для связи p_1 – p_2 будет равен

$$df_1 \cdot \vec{x}_{12} \frac{m_1}{m_1 + m_2}. \text{ Тогда } \vec{x}_2^n = \vec{x}_2^* + d\vec{l}_2.$$

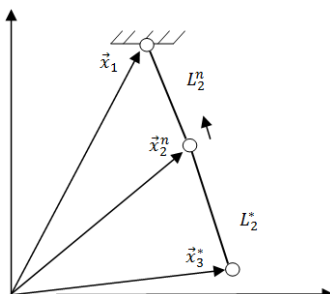


Рис. 6. Смещение частицы p_2

4) Применяем ограничение для связи между частицами p_2 и p_3 .
В связи p_2 - p_3 происходит смещение обеих частиц (рис. 7).

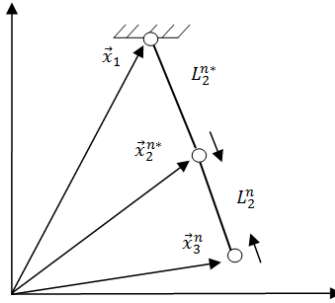


Рис. 7. Смещение частиц p_2 и p_3

Векторы смещения $d\vec{l}_2$ и $d\vec{l}_3$ для связи p_2 - p_3 будут равны:

$$d\vec{l}_2 = df \cdot \vec{x}_{23} \frac{m_3}{m_2 + m_3},$$

$$d\vec{l}_3 = df \cdot \vec{x}_{23} \frac{m_2}{m_2 + m_3}.$$

Тогда

$$\vec{x}_2^n = \vec{x}_2^* + d\vec{l}_2,$$

$$\vec{x}_3^n = \vec{x}_3^* + d\vec{l}_3.$$

Пункты 3–4 повторяются несколько раз, тем самым удовлетворяя условию (1). Повторение происходит до тех пор, пока df_1 и df_2 не станут меньше установленного значения k .

По достижении условия (1) начинается следующая итерация расчета с п. 2.

Заключение. Представленный алгоритм позволяет моделировать как один, так и несколько тросов, находящихся в одной системе. Такие параметры, как масса и длина троса, могут настраиваться благодаря использованию ограничений и материальных точек. Данный метод был применен к решению задачи моделирования троса в проекте «Компьютерный тренажер крановщика порталного крана».

Библиографический список

1. Файзрахманов Р.А., Хабибулин А.Ф. Проектирование и разработка тренажерного комплекса оператора портального крана // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2014. – 1(9). – С. 80–92.

2. Verlet L. Computer «experiments» on classical fluids. I. Thermodynamical properties of Lennard-Jones molecules // Physical review. – 1967. – 98 с.

3. Numerical Recipes. The Art of Scientific Computing / W.P. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery. – New York: Cambridge University Press. – 1256 с.

4. Jakobsen T. Advanced character physics // Game Developers Conference. – 2001. – С. 383–401.

РАЗРАБОТКА ТЕХНОЛОГИИ СОЗДАНИЯ МОБИЛЬНЫХ ПРИЛОЖЕНИЙ НА ОСНОВЕ ИНФРАСТРУКТУРЫ APACHE CORDOVA

Магистрант Д.А. Шестаков

Пермский национальный исследовательский
политехнический университет

Руководитель технологического отдела И.А. Братчиков

Группа компаний ИВС, г. Пермь

В статье рассмотрены аспекты разработки технологии создания мобильных приложений на основе инфраструктуры Apache Cordova.

Ключевые слова: мобильные приложения, Apache Cordova, OData.

DEVELOPMENT OF THE TECHNOLOGY FOR BUILD MOBILE APPLICATIONS BASED ON INFRASTRUCTURE APACHE CORDOVA

Master Student D.A. Shestakov

Perm National Research Polytechnic University

Manager of technological department I.A. Bratchikov

ICS Group of Companies, Perm

In this article describes aspects of development of the technology for build mobile applications based on infrastructure Apache Cordova.

Keywords: mobile applications, Apache Cordova, OData.

В современных условиях большую распространенность получили информационные системы, основанные на трёхуровневой архитектуре. Трёхуровневая архитектура – модель программного комплекса, состоящего из трёх компонентов: клиента, сервера приложений и сервера баз данных.

Основная бизнес-логика в таких системах обрабатывается на стороне сервера приложений. Он отвечает также за обмен информацией между клиентским слоем и слоем данных. За хранение информации и обмен данными с сервером приложений отвечает слой данных. Клиентский слой представляет собой пользовательский интерфейс. Это может быть браузер, мобильное приложение, настольный клиент и пр. На этот уровень выносятся только простейшая бизнес-логика: простая обработка данных, переданных сервером приложений, проверка вводимых значений.

В группе компаний ИВС была разработана платформа Flexberry. Целью данной платформы является решение бизнес-задач посредством создания программного обеспечения (ПО). Flexberry предоставляет для разработчиков набор инструментов и компонент, на основе которых может осуществляться разработка приложений. Платформа Flexberry позволяет сгенерировать на основе UML-диаграмм классов систему, соответствующую трехуровневой архитектуре и обладающую базовым функционалом.

На данный момент в платформе Flexberry отсутствуют полноценные CASE-средства, позволяющие осуществлять автоматизированную генерацию приложений для мобильных устройств под выбранную целевую платформу с заданным типовым набором функций.

Поскольку на сегодняшний день разработка программных решений группы компаний ИВС во многом ориентирована на использование платформы Flexberry, предполагается расширение возможностей платформы Flexberry по генерации кроссплатформенных мобильных приложений.

Процесс разработки генератора мобильных приложений можно разделить на четыре этапа:

1. Проектирование тестового приложения.
2. Разработка тестового приложения.
3. Выделение фреймворка приложения.
4. Создание генератора приложений.

На этапе проектирования были выделены ключевые особенности разрабатываемого приложения:

1. Генерируемые приложения должны запускаться на всех основных мобильных ОС.
2. Генерируемые приложения должны взаимодействовать с сервером приложений по протоколу OData. OData (Open Data Protocol) – это открытый веб-протокол для запроса и обновления данных. Протокол позволяет выполнять операции с ресурсами, используя в качестве запросов HTTP-команды, и получать ответы в форматах XML или JSON.

На следующем этапе было необходимо разработать тестовое приложение. Для создания приложения было принято решение вести разработку с использованием инфраструктуры Apache Cordova. Разработка приложений с помощью Apache Cordova ведется на языке JavaScript, что позволяет использовать в работе JavaScript фреймворки и библиотеки.

Одна из выделенных особенностей приложения – это работа с протоколом OData. Для упрощения разработки было принято решение использовать клиентскую JavaScript-библиотеку для работы с протоколом OData. Были рассмотрены все популярные библиотеки (Breeze.js, datajs, JayData, OpenUI5 и пр.). В результате сравнительного анализа библиотек была выбрана библиотека Breeze.js.

Перед непосредственной разработкой тестового приложения был сформулирован ряд требований к нему:

- данные необходимо получать с сервера по протоколу OData;
- должна быть реализована возможность работы с приложением в условиях отсутствия интернет-соединения, т.е. данные необходимо хранить локально на устройстве и при возможности синхронизировать с данными, расположенными на сервере;
- данные представляют собой массив объектов. Каждый объект содержит три поля;
- на главной форме приложения данные должны быть представлены в виде таблицы;
- должны быть реализованы следующие способы работы с данными:
 - добавление нового объекта;
 - удаление объекта;
 - редактирование существующего объекта.

Было принято решение разрабатывать приложение в виде Single Page Application (SPA) как наиболее подходящий вариант для инфраструктуры Apache Cordova. Были рассмотрены популярные JavaScript-фреймворки, предназначенные для разработки SPA. В результате был выбран фреймворк AngularJS как наиболее соответствующий сформулированным требованиям.

В результате разработки был получен прототип мобильного приложения со следующей структурой (рис. 1): add.html – файл, содержащий разметку формы добавления объекта; add.js – файл, содержащий контроллер формы добавления объекта; edit.html – файл, содержащий разметку формы редактирования объекта; edit.js – файл, содержащий контроллер формы редактирования объекта; main.html – файл, содержащий разметку главной списковой формы; main.js – файл, содержащий контроллер главной списковой формы; app.js – файл, содержащий главный модуль Angular-приложения и его настройки; datacontext.js – файл, содержащий методы для работы с данными; angular.js, angular-

route.js – файлы фреймворка AngularJS; breeze.angular.js – файлы библиотеки Breeze.js; config.xml, packages.config – файлы настройки проекта Apache Cordova; index.html – файл, содержащий общую часть разметки для всех форм приложения.

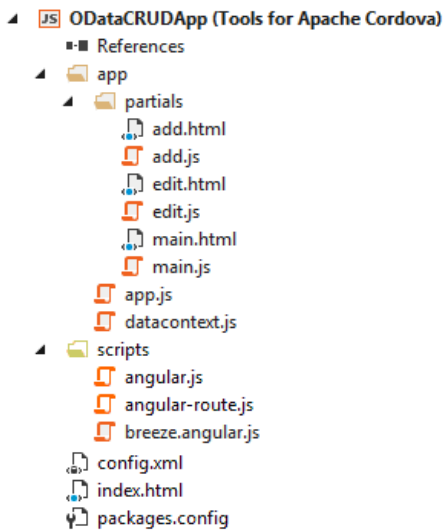


Рис. 1. Структура прототипа приложения

Разработанное приложение имеет следующий интерфейс (рис. 2).

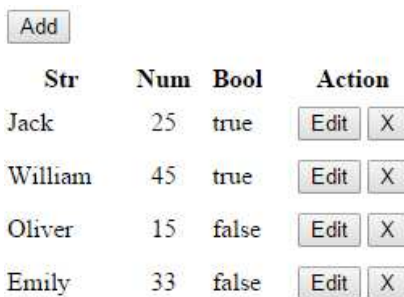


Рис. 2. Главная списковая форма приложения

При запуске приложения производится проверка доступности интернет-соединения. Если интернет-соединение доступно, производится синхронизация данных локального хранилища с сервером. В случае

отсутствия интернет-соединения остается возможность работать с данными, загруженными при последнем соединении с сервером.

В результате проделанной работы был разработан прототип мобильного приложения на основе инфраструктуры Apache Cordova с использованием JavaScript фреймворка AngularJS и библиотеки Breeze.js. Приложение позволяет обмениваться данными с сервером по протоколу OData. При отсутствии интернет-соединения есть возможность работы с информацией, загруженной при последнем подключении к серверу. Приложение поддерживает функции чтения, добавления, редактирования и удаления данных.

Для создания генератора кроссплатформенных мобильных приложений необходимо решить еще ряд задач:

- определить необходимость доработки средств проектирования платформы Flexberry для генерации Cordova-приложений;
- создать спецификацию требований для разрабатываемого компонента генерации;
- выполнить реализацию и тестирование компонента генерации Cordova-приложений;
- создать набор документации для разработанной технологии;
- выполнить апробацию разработанной технологии.

Библиографический список

1. Apache Cordova [Электронный ресурс]. – URL: <http://cordova.apache.org/#about> (дата обращения: 14.11.2015).
2. Libraries | OData – The Protocol for REST APIs [Электронный ресурс]. – URL: <http://www.odata.org/libraries> (дата обращения: 14.11.2015).
3. AngularJS – Superheroic JavaScript MVW Framework [Электронный ресурс]. – URL: <https://angularjs.org> (дата обращения: 14.11.2015).
4. Breeze.js | Breeze [Электронный ресурс]. – URL: <http://www.getbreezenow.com/breezejs> (дата обращения: 14.11.2015).

ГРУППИРОВКА ДЕТАЛЕЙ ПРИ РЕШЕНИИ ЗАДАЧ ФИГУРНОГО РАСКРОЯ ЛИСТОВОГО МАТЕРИАЛА

Ассистент В.С. Шилов

Пермский национальный исследовательский
политехнический университет

В статье описывается алгоритм формирования битов деталей, реализованный автором в программном комплексе автоматизированного раскроя листового материала Itas Nesting. Под битом понимается группа деталей, положение которых друг относительно друга зафиксировано и не меняется при формировании карты раскроя. Показана эффективность объединения деталей в биты с использованием предложенных алгоритмов.

Ключевые слова: раскрой-упаковка, фигурный раскрой, бит, группировка деталей.

DETAIL GROUPING IN NESTING PROBLEMS

Assistant V.S. Shilov

Perm National Research Polytechnic University

The paper deals with the algorithm of forming the detail bits from single details, implemented by the author in the software for automated cutting Itas Nesting. Bit is a group of details, whose relative position fixed and does not change during the formation of the cutting layout. The effectiveness of detail combining into bits using the proposed algorithms was shown.

Keywords: cutting-packing; nesting; bit; detail grouping.

Фигурный раскрой является наиболее сложным подклассом задач раскроя листового материала [1, 2]. Это связано с трудностью как формализации условия взаимного непересечения деталей, так и построения конструктивных алгоритмов решения данного класса задач.

В [3] введено понятие коэффициента прямоугольности задания на раскрой и показано, что если значение этого коэффициента превосходит 0,8, то алгоритмы ортогонального раскроя оказываются более эффективными, чем алгоритмы фигурного раскроя. Кроме того, отмечается, что объединение деталей в так называемые биты позволяет значительно повысить величину коэффициента прямоугольности задания на раскрой. Под битом понимается группа деталей, положение которых друг относительно друга зафиксировано и не меняется при формировании карты раскроя. На рис. 1 приведен пример построения битов деталей. Алгоритмы построения битов в значи-

тельной степени изучены [4, 5, 6]. В данной статье описывается алгоритм, примененный автором в программном комплексе автоматизированного раскроя материала Itas Nesting [7]. Описанный алгоритм имеет достаточно простую реализацию и позволяет значительно улучшать решения, получаемые с помощью алгоритмов ортогонального раскроя; в качестве такого алгоритма автором используется групповой декодер [8].

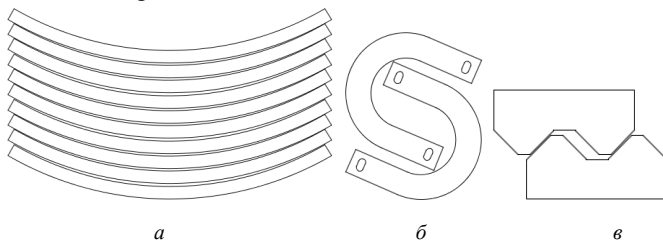


Рис. 1. Пример построения битов деталей

Определим критерий построения битов. Объединение деталей в бит должно уменьшать коэффициент прямоугольности задания на раскрой. Это утверждение позволяет сформулировать следующий критерий:

$$K = \frac{S_b}{n \cdot S_0} \rightarrow \min, \quad (1)$$

где S_b – площадь описанного вокруг бита прямоугольника; S_0 – площадь описанного вокруг детали прямоугольника; n – количество деталей в бите.

Следовательно, необходимо минимизировать отношение площади описанного вокруг бита прямоугольника к суммарной площади всех описанных вокруг деталей прямоугольников.

Здесь и далее в работе описанным прямоугольником является такой прямоугольник, в котором деталь полностью лежит внутри него, а прямоугольник имеет минимально возможную площадь. Для построения такого прямоугольника используется алгоритм, описанный в [9].

Сформулируем теперь алгоритм построения бита деталей. На первом шаге необходимо произвести «выравнивание» детали, т.е. выполнить такой ее поворот, чтобы стороны описанного вокруг детали прямоугольника были параллельны осям координат. Анализ большого числа чертежей показал, что достаточно рассмотреть варианты

стыковки двух деталей, допускающие ортогональные повороты на 0, 90, 180, 270 градусов. Необходимость рассмотрения других углов поворотов может возникнуть, если детали оказываются достаточно сложными; в такой ситуации применение методов ортогонального раскроя к данной задаче будет нецелесообразным. Помимо этого необходимо рассматривать различные сдвиги деталей относительно друг друга вдоль оси OX (рис. 1, *в*). Для заданных углов поворота и сдвига вдоль оси OX стыковка производится выполнением следующих шагов:

1. Разместить детали с учетом заданных углов поворота и сдвига вдоль оси OX , при этом вторую деталь разместить значительно выше первой (так, чтобы детали гарантированно не пересекались).

2. Выполнить сдвиг второй детали таким образом, чтобы детали начали соприкасаться, но не пересекаться. Определение длины вектора такого сдвига описано в [10].

Как уже отмечалось, вторая деталь может быть сдвинута на некоторый вектор вдоль оси OX . Величина вектора такого сдвига может быть определена перебором: достаточно с некоторым шагом рассмотреть все сдвиги, при которых проекции деталей на ось OX будут пересекаться. Из всех вариантов необходимо выбрать тот, при котором критерий (1) достигает минимума. Если значение данного критерия оказалось больше или равно единице, то объединение деталей в биты не позволит повысить коэффициент прямоугольности задания на раскрой, а значит, детали целесообразно рассматривать по отдельности.

Также может быть ситуация, когда целесообразно создавать бит более чем из двух деталей (рис. 1, *а*). Признаком такой ситуации является то, что углы поворота обеих деталей равны и сдвиг по оси OX равен нулю. В таком случае следует создать бит из того количества данных деталей, какое есть в задании на раскрой, если это позволит улучшить значение критерия (1). Формирование более сложных структур из одинаковых деталей в данной работе не рассматривается, так как зачастую это оказывается нецелесообразным. Алгоритмы дополнительного уплотнения на этапе формирования карт раскроя справляются с этой задачей более эффективно.

Предложенный алгоритм реализован в программном комплексе автоматизированного раскроя листового материала. На рис. 2 приведены примеры карт раскроя, полученных без использования предложенного алгоритма, на рис. 3 – примеры карт, для формирования которых использовалось объединение деталей в биты.

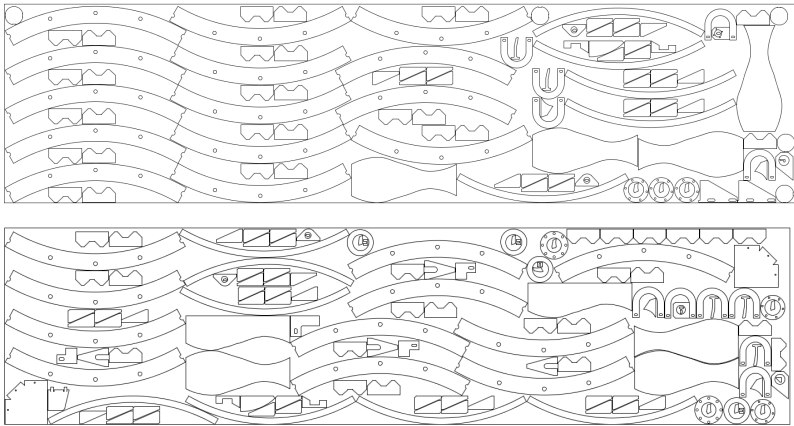


Рис. 2. Пример карт раскроя, полученных без объединения деталей в биты

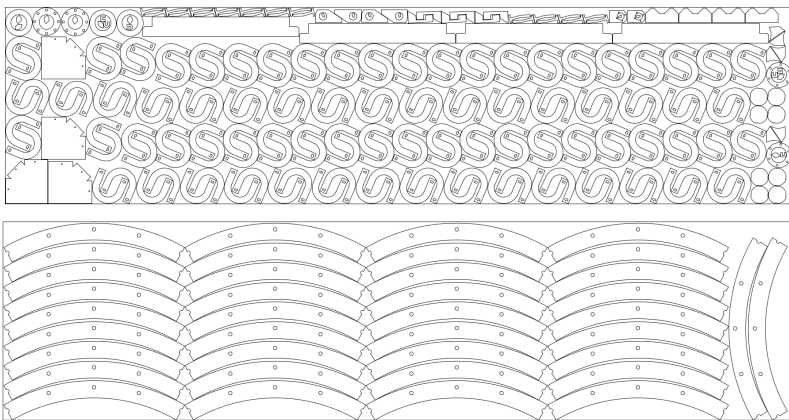


Рис. 3. Пример карт раскроя, полученных с объединением деталей в биты

В статье описан алгоритм формирования битов деталей – групп деталей, положение которых друг относительно друга зафиксировано и не меняется при формировании карты раскроя. Предложенный алгоритм реализован и используется в программном комплексе автоматизированного раскроя листового материала Itas Nesting. Применение описанных алгоритмов совместно с методами ортогонального раскроя позволяет использовать материал в среднем в 1,5 раза эффективнее.

Библиографический список

1. Решение задачи ортогональной упаковки листовых материалов методами линейного раскроя / Р.А. Файзрахманов, Р.Т. Мурзакаев, В.С. Шилов, А.С. Мезенцев // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2014. – № 10. – С. 29–42.
2. Romanova T., Stoyn Y., Pankratov A. Mathematical models and solution algorithm for nesting problem of arbitrary shaped objects // 8th Conference of the special interest group on cutting and packing (ESICUP), Copenhagen, Denmark, 2011.
3. Петунин А.А. Методологические и теоретические основы автоматизации проектирования раскроя листовых материалов на машинах с числовым программным управлением: дис. ... д-ра техн. наук: 05.13.12. – Екатеринбург, 2009. – 348 с.
4. Фроловский В.Д. Оптимальное группирование геометрических объектов при проектировании карт раскроя материалов // Программные продукты и системы. – 2000. – № 3. – С. 47–48.
5. Руднев А.С. Вероятностный поиск с запретами для задачи упаковки кругов и прямоугольников в полосу // Дискретный анализ и исследование операций. – Новосибирск, 2009. – Т.16. – № 4. – С. 61–86.
6. Lamousin H., Waggenspack W. Nesting of two-dimensional irregular parts using a shape reasoning heuristic // Computer-Aided Design. – 1997. – Vol. 29. – №. 3.
7. Мурзакаев Р.Т., Шилов В.С., Брюханова А.А. Программный комплекс фигурного раскроя листового материала Itas Nesting // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2015. – № 13. – С. 15–25.
8. Application of the Group Decoder for Solving the Orthogonal Materials Cutting Problem [Электронный ресурс] / R.A. Fayzrakhmanov, R.T. Murzakaev, A.S. Mezentsev, V.S. Shilov // World Applied Sciences Journal. – 2013. – № 28(10). – P. 1361–1365. – URL: [http://www.idosi.org/wasj/wasj28\(10\)13/4.pdf](http://www.idosi.org/wasj/wasj28(10)13/4.pdf) (дата обращения: 07.07.2014).
9. Murzakaev R.T., Shilov V.S., Mezentsev A.S. On the nesting layout in view on technical gaps for joint cutting // Russian electrical engineering. – 2015. – № 11. – P. 668–672.

Секция 2

**АВТОМАТИЗАЦИЯ И УПРАВЛЕНИЕ ТЕХНОЛОГИЧЕСКИМИ
ПРОЦЕССАМИ И ПРОИЗВОДСТВАМИ**

ИССЛЕДОВАНИЕ МЕТОДОВ АНАЛИЗА ПРОСТРАНСТВЕННЫХ ДАННЫХ НА КЛИЕНТСКОЙ ЧАСТИ ВЕБ-ПРИЛОЖЕНИЙ

Магистрант К.Б. Архипов

Пермский национальный исследовательский
политехнический университет

Руководитель ресурсно-технологического центра П.Е. Радостев

Группа компаний ИВС, г. Пермь

Выполнен краткий анализ подходов для разработки картографических веб-приложений. Выполнен обзор основных средств геопроецирования на клиентской части, а также рассмотрены альтернативные технологии, позволяющие осуществлять его на сервере. Обоснована актуальность разработки именно клиентских средств для анализа пространственных данных. Предложена концепция модуля, позволяющего конвейеризировать инструменты геопроецирования.

Ключевые слова: геоинформационные системы, веб-разработка, клиентская разработка, веб-картография, анализ пространственных данных, геопроецирование.

RESEARCH OF SPATIAL ANALYSIS APPROACHES AT FRONT-END PART OF WEB-APPLICATIONS

Master Student K.B. Arkhipov

Perm National Research Polytechnic University

Head of resource and technology department P.E. Radostev

ICS Group of Companies, Perm

This article gives a brief analysis of existing approaches for cartographic web-applications development. The overview of main tools for spatial analysis and alternative server technologies was put through. The actuality of developing client tools for spatial analysis was justified and proposed the concept of the module for spatial tools queueing.

Keywords: geographic information systems; web-development; front-end development; web-cartography; spatial data analysis; geoprocessing.

С развитием Интернета разработка корпоративных систем управления стала активно смещаться в область разработки веб-приложений, в этом плане ГИС-приложения не стали исключением. В основе работы веб-ГИС лежат ГИС-сервисы, обеспечивающие доступ к пространственным данным, а также берущие на себя функции их обработки, анализа, поиска и визуализации.

Как правило, ГИС-сервисы реализованы в виде компонент серверного ПО [1]. Примерами таких серверов являются, ArcGIS for Server компании Esri или открытые GeoServer, MapServer, QGIS Server и т.п., поэтому обычно при разработке картографических веб-приложений выполняются следующие действия:

- ГИС-ресурсы создаются при помощи настольных ГИС-приложений (QGIS, ArcGis);
- подготовленные ресурсы публикуются при помощи ГИС-серверов;
- к опубликованным ресурсам по принципу REST обращаются клиентские скрипты и визуализируют полученные данные.

Особенность такого подхода в том, что именно серверное ГИС ПО и реализует все основные функции, в том числе и пространственный анализ, а клиентская часть самостоятельно способна лишь обращаться к серверной за подготовленными ГИС-ресурсами. Примером такого подхода является GeoServer, который берет на себя функции геопроцессинга при помощи реализации стандарта WPS.

Стандарт WPS (веб-сервис геопроцессинга) нужен для описания правил для входящих и исходящих данных для сервисов геопроцессинга, таких как пересечение полигонов и др. WPS определяет веб-интерфейсы доступа к сервису геопроцессинга. Геопроцессинг может включать любой алгоритм, расчёт или модель, оперирующие векторными или растровыми данными с пространственными, такие как вычитание одного набора пространственных данных из другого (например, разница между областями распространения гриппа по сезонам года) и более сложные, например, глобальная модель изменения климата [2].

Несмотря на широту функционала, доступного при помощи данной технологии, такой подход нагружает сервер, так как практически каждое действие пользователя обрабатывается сервером. По этой причине он совершенно не приемлем в приложениях, предназначенных для массового пользования.

В стремлении разгрузить сервер развитие технологий движется в сторону насыщения клиентской части веб-приложений большей функциональностью, появляется класс клиентских приложений, получивший название «богатый клиент». Активно развиваются различные открытые библиотеки для организации клиентской части веб-ГИС. На сегодняшний день наиболее популярными среди них являются «Leaflet» и «Open layers». Они берут на себя кастомизации,

вывод данных, создание пользовательских слоев и дополнительного функционала картографической системы. Также частично они способны выполнять геопроецессинг. Однако возможности одних лишь картографических библиотек не сопоставимы с возможностями таких сервисов, как WPS.

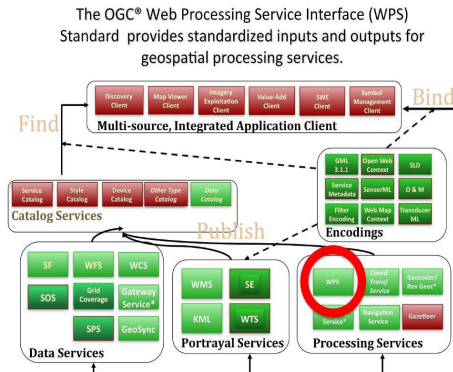


Рис. Позиционирование WPS относительно других стандартов OGC

С другой стороны, геопроецессинг не входит в обязанности картографических библиотек, так как их задачей является управление картой, а не комплексная обработка данных. Иными словами, обработка пространственных данных – это исконно серверная прерогатива, точнее, так было до недавнего времени.

Но современный javascript перестает быть просто языком для реализации скриптинга на веб-страницах. Ввиду его производительности сегодня он также применяется в качестве серверного языка при помощи платформы Node.js. Многие библиотеки для node.js можно использовать не только на серверной стороне, но и на клиентской, что позволяет безболезненно выносить туда часть функционала. Таким образом, появляется все больше инструментов, позволяющих производить на стороне клиента сложную работу с данными.

В рамках данной статьи были произведены поиск и анализ библиотек, позволяющих производить пространственный анализ при помощи javascript. В результате было найдено 2 библиотеки – turf.js и jsts.js.

Turf – это JavaScript-библиотека для пространственного анализа. Она включает стандартные пространственные инструменты, вспомогательные функции для приведения данных к формату GeoJSON, а также инструменты классификации и статистики. Turf может быть

использован на клиентской части сайта либо на серверной при помощи Node.js.

Turf является независимой от картографического фреймворка библиотекой, в которой собраны различные средства для пространственного анализа. Можно считать, что в Turf реализован функциональный подход, так как нет никакой привязки к классам и методам. Всего библиотека насчитывает 60 инструментов, которые можно классифицировать как:

1) геометрия – создание из обычных координат объектов feature стандарта GeoJSON;

2) джоин – функции, которые позволяют объединять геометрии по различным критериям;

3) данные – функции, осуществляющие работу с данными;

4) измерение – функции, производящие измерения с геоданными;

5) интерполяция – функции для осуществления интерполяции над наборами данных;

6) решетка – функции для создания решеток по заданной области;

7) классификация – функции, реализующие различные методы классификации и кластеризации;

8) агрегация – функции, направленные на преобразование и исследование двух входных коллекций feature;

9) трансформация – функции, предназначенные для изменения текущего набора данных или создания нового на основе существующего;

10) разное – функции, не относящиеся ни к одной категории.

Turf осуществляет работу с данными исключительно в формате GeoJSON. Также использование Turf предполагает, что входные данные будут соответствовать стандарту WGS84.

Топологический модуль JSTS является JavaScript-библиотекой содержащей пространственные предикаты и функции для приведения геометрий в соответствие со стандартом Simple Features Specification для SQL, опубликованным OGC. Топологический модуль JSTS также является портом Java-библиотеки JTS, в который встроен парсер геометрий для OpenLayers.

Основной целью проекта является предоставление приложениям OpenLayers полноценной библиотеки для обработки и анализа геометрий, однако, начиная с версии 0.11.0, жесткая привязка к OpenLayers была убрана, что позволяет использовать данный продукт в качестве независимой библиотеки для пространственного анализа.

В отличие от Leaflet данная библиотека имеет частичную привязку к картографическому фреймворку OpenLayers. По этой причине в ней в качестве геометрий выступают объекты, соответствующие по структуре объектам OpenLayers. Также в отличие от Leaflet в ней реализован объектно-ориентированный подход. Все классы хранятся в соответствующих пространствах имен, а весь функционал содержится в методах классов.

В JSTS есть множество пространств имен, отвечающих за различные задачи. Например, есть такое пространство имен, как геометрии (`jsts.geom`), включающие в себя реализацию доступных геометрий в библиотеке, а также методы для взаимодействия с ними. Геометрия, в свою очередь, имеет такие подпространства имен, как Точка или Полигон (`jsts.geom.point`, `jsts.geom.polygon`). Помимо подпространств оно также имеет собственные методы, которые универсальны для всех вложенных подпространств и их методов.

Основным ограничением является привязка к объектной модели OpenLayers, что затрудняет адаптацию данной библиотеки для других фреймворков. Объектно-ориентированный подход может затруднять конвейеризацию функций.

Анализ существующих клиентских библиотек пространственного анализа показал отсутствие комплексных инструментов для обработки пространственных данных. Это доказывает актуальность разработки метаинструмента, позволяющего составлять комплексные инструменты, решающие определенную задачу пользователя, из имеющихся атомарных. Необходимо разработать программный модуль, способный последовательно выполнять заданные функции с возможностью обмена данными между элементами последовательности.

Основные требования к модулю:

1) универсальность входных данных. Данные могут быть как в виде объекта, так и в виде строки json;

2) предварительный контроль данных. Еще до выполнения модели должна быть реализована возможность предварительно проверить правильность предоставленной модели;

2.1) предварительный контроль типов входных данных, указанных в модели и параметров, которые принимает выполняемая функция: должна присутствовать возможность контролировать корректность данных в соответствии с нотацией GeoJSON 1.0;

2.2) проверка на наличие всех обязательных параметров, которые принимает выполняемая функция;

2.3) наличие выполняемого метода;

2.4) проверка на правильность источника входных данных. Доступные значения: внешний «outer», указывающий на экземпляр объекта где-то вне модели, «rpid» – порядковый номер элемента в массиве, выходной параметр которого является входным параметром у исполняемой функции;

3) основной контроль данных. Отличие от предварительного в том, что осуществляется прямо во время выполнения модели, и в случае источника данных «rpid» значения, сопоставляемые с допустимыми, берутся исключительно из результатов выполнения предыдущих методов;

4) обработка исключений. Все всплывающие исключения должны записываться в массив ошибок;

5) расширяемость функционала. Должна присутствовать возможность внедрения пользовательских функций;

6) внутренний функционал должен быть инкапсулирован, чтобы избежать повторения имен;

7) должна присутствовать возможность нанесения промежуточных результатов работы модели на карту.

Основным инструментом для разработки модуля является язык JavaScript.

Для разработки плагинов к картографическим библиотекам будет использоваться JavaScript в связке с JS API соответствующих библиотек.

Для того чтобы модуль могли использовать и дорабатывать все желающие, планируется размещение модуля и соответствующих ему плагинов для картографических библиотек в открытом репозитории на «GitHub».

Предполагается апробация разрабатываемого модуля на веб-проектах группы компаний ИВС в связке с ГИС-компонентом ИВС, основанном на библиотеке «Leaflet».

Библиографический список

1. Веб-ГИС [Электронный ресурс]. – URL: <http://gis-lab.info/qa/webgis.html> (дата обращения: 21.11.2015).

2. Веб-ГИС [Электронный ресурс]. – URL: <http://docs.opengeospatial.org/is/14-065/14-065.html> (дата обращения: 30.11.2015).

СОВРЕМЕННЫЕ ПОДХОДЫ К УПРАВЛЕНИЮ ПОКАЗАТЕЛЯМИ ТЕХНОЛОГИЧНОСТИ

**Аспирант, начальник бюро систем автоматизированного
проектирования Д.А. Глухов**

Пермский национальный исследовательский политехнический
университет, ПАО «Мотовилихинские заводы» г. Пермь

Инженер-конструктор 1-й категории Я.А. Катаев

ЗАО «Специальное конструкторское бюро»

ПАО «Мотовилихинские заводы», г. Пермь

Кандидат технических наук, доцент В.Р. Туктамышев

Пермский национальный исследовательский
политехнический университет

Начальник бюро перспективных технологий А.О. Гришарин

ПАО «Мотовилихинские заводы», г. Пермь

В статье представлен обзор классического подхода к определению перечня показателей технологичности, в том числе на основе ГОСТ 14.201-83. Использование концепции «таргет-костинга» позволяет сформировать целевые показатели себестоимости и технологичности. Использование PLM-системы позволяет значительно повысить эффективность процесса управления технологичностью и повысить конкурентоспособность предприятия.

Ключевые слова: себестоимость, технологичность конструкции, показатели технологичности, PLM-система.

MODERN APPROACHES TO MANAGING PROCESSING INDEX

**Graduate Student, Chief of bureau of computer-aided
design D.A. Glukhov**

Perm National Research Polytechnic University,
PJSC "Motovilikhinskie Zavody", Perm

Design Engineer 1 category Y.A. Kataev

JSC "Special Design Bureau" PJSC "Motovilikhinskie Zavody", Perm

Ph.D. in Technical Sciences, Associate Professor V.R. Tuktamyshev

Perm National Research Polytechnic University

Chief of Bureau of promising technologies A.O. Grisharin

PJSC "Motovilikhinskie Zavody", Perm

The article provides an overview of the classical approach to determining the list of indicators manufacturability, including on the basis of GOST 14.201-83. Using the concept of "target costing" allows to generate targets for cost and manufacturability. Using PLM-systems can significantly increase the efficiency of the process control technology and improve the competitiveness of enterprises.

Keywords: Cost price; manufacturability of the design; the processing; PLM-system.

Для обеспечения успешной работы машиностроительных предприятий в современных рыночных условиях особое значение имеет процесс управления себестоимостью проектируемых и изготавливаемых изделий. Традиционно различают следующие виды себестоимости: технологическую, цеховую, производственную, полную.

В структуре себестоимости, в части переменных затрат, на конечную величину себестоимости значительное влияние оказывает технологичность конструкции. Для оперативного управления себестоимостью изделия на этапах проектирования, постановки на производство и изготовления инженерным службам важно правильно организовать процесс управления технологичностью конструкции.

Технологичность конструкции – это одна из комплексных характеристик технического устройства (изделия, прибора, аппарата), которая отражает удобство его производства, ремонтпригодность и эксплуатационные качества. Понятие «управление технологичностью конструкции» предусматривает такую организацию процесса проектирования, которая, при соблюдении всех эксплуатационных качеств, обеспечивает минимальную трудоемкость и материалоемкость изготовления, а также минимальный цикл производства изделия. Технологичность конструкции изделия оценивают совокупностью показателей технологичности. Обычно для оценки технологичности изделия используют показатели технологичности. Перечислим базовые показатели.

Суммарная материалоемкость изделия. Показатель характеризует общее количество материалов заготовок, использованное для изготовления изделия.

Удельная материалоемкость изделия. Показатель характеризует отношение величины суммарной материалоемкости изделия к базовой потребительской характеристике, например максимальному перевозимому весу груза.

Суммарная трудоемкость изделия. Показатель характеризует общее количество трудоемкости, затрачиваемое на изготовление изделия.

Удельная трудоемкость изделия. Показатель характеризует отношение величины суммарной трудоемкости изделия к базовой потребительской характеристике, например максимально перевозимый вес груза.

Коэффициент использования материалов. Коэффициент показывает, какую часть из всего затраченного материала составляет «беловой» вес детали.

Коэффициент унификации изделия. Показатель характеризует отношение заимствованных деталей к общему количеству деталей в изделии.

Коэффициент стандартизации изделия. Показатель характеризует отношение количества применяемых стандартных и стандартизированных деталей к общему количеству деталей в изделии.

Коэффициент унификации конструкционных элементов. Показатель характеризует количество размеров конструкционного элемента, применяемых во всех оригинальных деталях изделия. Рассчитывается для каждого вида конструкционного элемента (резьбы, фаски, паза, отверстия и т.д.).

Коэффициент унификации марок и профилей применяемых материалов. Показатель характеризует повторяемость марок и профилей материалов, применяемых в изделии.

Для классического управления себестоимостью характерен следующий подход. Цена выпускаемого изделия определяется по формуле: $\text{СЕБЕСТОИМОСТЬ} + \text{ПРИБЫЛЬ} = \text{ЦЕНА}$. В настоящее время для управления себестоимостью активно применяется метод «таргет-костинга» (англ. *Target Cost Management*). Концепция таргет-костинга предполагает иной подход к формированию цены и управлению себестоимостью. Себестоимость продукции рассчитывается по формуле: $\text{ЦЕНА} - \text{ПРИБЫЛЬ} = \text{СЕБЕСТОИМОСТЬ}$. Плановая цена изделия определяется на основе маркетинговых исследований, желаемая прибыль компании, как правило, закладывается в виде определенного показателя. Исходя из заданной цены и прибыли, рассчитывается целевой показатель для себестоимости.

В настоящее время процесс управления себестоимостью следует рассматривать в контексте всего жизненного цикла изделия от разработки ТТЗ (определения базовых потребительских и технических свойств изделия) до утилизации. Особенно важно уделять внимание вопросам управления себестоимостью на этапах формирования технического задания, проектирования, постановки изделия на производство установочной партии, изготовления установочной партии и подготовки производства изготовления серийного изделия.

Установленный целевой показатель себестоимости изделия является основанием для расчета целевых показателей технологичности. Для определения показателей технологичности также необходимо знать технические возможности предприятия, на котором планируется изготовление изделия, технические возможности основных

предприятий-смежников и плановую серийность изготовления изделия. Конкретный перечень и значения целевых показателей должны фиксироваться в техническом задании на стадии проведения опытных конструкторских работ.

Процесс достижения целевых показателей технологичности называется отработкой на технологичность. Для достижения целевых значений показателей требуется организовать процесс совместной работы всех специалистов, задействованных в процессе разработки и постановки изделия на производство, и в первую очередь конструкторов, технологов, производственников, снабженцев.

Для сложных изделий процесс контроля показателей технологичности может быть очень трудоемок. Это связано с большим количеством контролируемых параметров. Номенклатура деталей и сборочных единиц в изделиях сложного машиностроения может достигать несколько десятков и даже сотен тысяч позиций.

В настоящее время практически все конструкторские бюро и машиностроительные предприятия страны уже внедрили или внедряют PDM/PLM-системы. Особенно актуально применение таких систем в случае проектирования и(или) изготовления сложных изделий. Работы по разработке и постановке изделий на производство проводятся в электронном виде в системах данного класса. Применение PLM-систем открывает новые возможности по управлению себестоимостью и технологичностью изделий.

Специалистами Пермского национального исследовательского политехнического университета были проведены работы по изучению возможностей современных PLM-систем в части управления технологичностью конструкции изделий. Исследования проводились, кроме прочего, на базе комплекса программного обеспечения Siemens PLM (Teamcenter и NX), также были рассмотрены отечественные комплексы. Перечислим основные преимущества использования PLM-систем для организации процесса управления технологичностью изделия.

Важным модулем любой PLM-системы является модуль «Управление требованиями». Модуль позволяет в структурированном виде заносить данные ТЗ либо декомпозированные требования ТЗ уже в виде требований к конкретным подсистемам изделия. Модуль должен использоваться для формирования плановых показателей технологичности. Модуль также используется для организации автоматизированного контроля достижения заданных показателей.

Основа любой инженерной PLM-системы – это база электронных макетов изделий. Кроме того, PLM-система содержит следующие основные базы данных:

- электронных чертежей и сканированных бумажных копий подлинников конструкторской документации изделий, проектировавшихся и изготавливавшихся на предприятии ранее;
- показателей технологичности изделий и узлов, проектировавшихся ранее;
- чертежей и моделей металлургических заготовок;
- технологических процессов изготовления, содержащих материальные и трудовые нормативы.

Также инженерная PLM-система содержит:

- справочники и ограничительные перечни материалов, стандартных изделий, покупных изделий;
- справочники и ограничительные перечни средств технологического оснащения (СТО), в том числе нестандартных средств оснащения (НСО);
- справочники и ограничители покупного инструмента (режущего основного, вспомогательного, измерительного и т.д.);
- справочник технологического оборудования;
- справочник и ограничитель конструкторско-технологических элементов.

Базы данных используются в процессе расчетов целевых показателей технологичности и в процессе отработки на технологичность деталей, сборочных единиц, систем и изделий в целом. Использование в качестве документации электронных геометрических моделей деталей и сборочных единиц позволяет вести все работы по отработке на технологичность в электронном виде в рамках PLM-системы.

Проведение всех работ по проектированию, отработке на технологичность, постановке изделий на производство в рамках единой PLM-системы дает возможность создания автоматизированной системы по контролю достижения целевых показателей технологичности. Это достигается за счет того, что большая часть показателей технологичности является автоматически вычисляемой, и того, что вся исходная информация для этих расчетов ведется в единой системе. Вычисляемыми показателями являются: «суммарная материалоемкость изделия», «удельная материалоемкость изделия», «суммарная трудоемкость изделия», «удельная трудоемкость изделия», «коэффициент унификации изделия», «коэффициент стандартизации изделия», «коэффициент ис-

пользования материалов», «коэффициент унификации марок и профилей применяемых материалов». «Коэффициент унификации конструктивных элементов» также является частично вычисляемым. С учетом возможности автоматического расчета показателей технологичности возможна организация процесса контроля достижения целевых показателей технологичности в режиме «реального времени».

Одними из основных модулей PLM-система являются модули «Управление потоками работ» и «Электронное согласование». В некоторых PLM-системах модули объединены в один функциональный модуль. Модули позволяют управлять жизненными циклами электронных объектов (документов) системы и проводить электронное согласование. Это позволяет выстраивать логику работы системы в соответствии с принятыми на предприятии регламентами и значительно сократить цикл самого согласования.

PLM-системы содержат в своем функционале модуль «Управление заданиями», который позволяет в автоматизированном режиме планировать и контролировать работу всех специалистов, задействованных в процессе отработки на технологичность изделия.

Использование PLM-систем в качестве инструмента организации управления технологичностью изделия позволяет автоматизировать значительное количество рутинных операций, сделать весь процесс управляемым и прозрачным. Значительно сокращаются потери в процессе работ по отработке на технологичность, в целом процесс становится значительно более эффективным. Эти результаты были подтверждены в ходе моделирования процесса отработки на технологичность на базе PLM-системы Teamcenter.

Взаимосвязанное применение концепции «Target Costing'a» и организованного процесса отработки на технологичность в PLM-системе дает значительные преимущества предприятиям в современных конкурентных условиях, позволяет повысить их конкурентоспособность.

Библиографический список

1. ГОСТ 14.201-83 Обеспечение технологичности конструкции изделий. Общие требования. – М.: Стандартиформ, 2009. – С. 8.
2. Ананьев С.Л. Технологичность конструкции. – М.: Дом техники, 1959. – 452 с.
3. Прялин М.А. Оценка технологичности конструкции. – Киев: Техника, 1985. – 117 с.

РАЗРАБОТКА ВЕБ-РЕДАКТОРА UML-МОДЕЛЕЙ

Магистрант Р.И. Глухов

Пермский национальный исследовательский
политехнический университет

**Канд. техн. наук, руководитель отдела развития
технологии А.С. Елисеев**

Группа компаний ИВС, г. Пермь

В статье описан архитектурный подход к построению многокомпонентного программного обеспечения. Выполнен обзор существующих веб-редакторов, а также библиотек, необходимых для разработки клиентской части редактора и обмена UML-моделями. Кроме того, описана архитектура веб-редактора, продемонстрирован разработанный интерфейс для рисования диаграмм, а также модули импорта и экспорта диаграмм.

Ключевые слова: веб-редактор, UML, метамодель, модель данных, svg, flexberry.

DEVELOPMENT OF UML-MODELS WEB-EDITOR

Master Student R.I. Glukhov

Perm national research polytechnic university

Ph.D. in Technical Sciences,

head of technology development A.S. Eliseev

ICS Group of Companies, Perm

This article describes an architectural approach to building a multi-component software. Further, the article gives an overview of existing web-editors, as well as the libraries required to develop editor client-side and exchange of UML-models. Additionally, the article describes the architecture of the web-editor, shown developed interface for drawing diagrams, as well as import and export modules charts.

Keywords: web-editor, uml, meta-data model, svg, flexberry.

В последнее время преобладающим подходом к проектированию программного обеспечения является объектно-ориентированный подход. Фундаментом подхода являются понятия класса, объекта, метода и атрибута. Система, построенная согласно данному подходу, представляет собой совокупность взаимодействующих между собой объектов. Задачей разработчика является разработка эффективной иерархии классов для реализации приложения, основываясь на трех основных принципах ООП: полиморфизме, инкапсуляции и наследовании [1].

С течением временем сложность разрабатываемых систем начала сильно возрастать. После того как в области информационных технологий были осознаны проблемы сложных задач, а также был накоплен некоторый опыт их решения, начали появляться различные методологии и технологии, призванные повысить эффективность разработки программного обеспечения. Одной из методологий является применение шаблонов проектирования. Это позволяет расширить концепцию повторного использования результатов разработки на архитектурном уровне. Однако недостаточно обдуманное применение шаблонов разработчиками низкого уровня приводит к перегруженности архитектуры, даже если она построена как совокупность известных шаблонов проектирования. Другим способом снижения риска ошибок на стадии разработки архитектуры является ее моделирование. В этом случае можно говорить о визуализации разрабатываемой архитектуры посредством некоторого программного средства и методологии моделирования.

Model Driven Architecture (MDA, архитектура управляемая моделью) – это архитектурный подход к построению многокомпонентного ПО. Он основан на разработке независимого от платформы и языка программирования представления системы (модели) с последующим переходом к исходному коду системы через автоматизированную генерацию кода [7]. Основной целью разработчиков MDA являлось создание архитектурного подхода, позволяющего снизить риск, вызванный различиями между технологиями разработки программных систем и платформами. Ядром MDA являются несколько стандартов – UML, MOF, CWM и XMI. Язык UML (Universal Modeling Language) используется для описания всех моделей. На текущий момент существует большое количество инструментов для редактирования UML. Совокупность метамodelей CWM (Common Warehouse Model) представляет наиболее часто используемые в базах данных и инструментах бизнес-анализа метаданные. CWM содержит большое количество различных метамodelей, описывающих функционирование бизнеса. MOF (Meta Object Facility) – общий абстрактный язык для описания метамodelей; на его основе построены формальные описания метамodelей для CWM и UML. Последний стандарт XMI (XML Metadata Interchange) играет служебную роль, описывая отображение моделей

MOF и UML на стандарт XML [7]. Наиболее часто XMI применяется как формат обмена UML-моделями из одного инструмента UML-моделирования в другой. Он позволяет работать с моделями вне зависимости от того CASE-инструмента, которым пользуется разработчик.

Одним из ярких представителей MDA являются технология CASEBERRY и платформа Flexberry, разрабатываемая с её помощью. Собственный инструмент создания UML-диаграмм позволяет строить UML-модели, генерировать код, экспортировать и импортировать диаграммы. Разработка на Flexberry имеет свои уникальные особенности: модель сразу разрабатывается под конкретную платформу, нет нужды строить платформонезависимую модель; в модель закладывается каркас бизнес-логики, реализуется она путем доработки исполняемого кода приложения. Такие среды разработки программного обеспечения, как «Visual studio» и «Eclipse», также активно поддерживают MDA, в них присутствует необходимый инструментарий для построения UML моделей, генерации кода, экспорта и импорта моделей в стандарт XMI.

На текущий момент основной проблемой подхода MDA является синхронизация изменений в моделях. Когда в проекте, разрабатываемом под разные платформы, изменили модель, разработчики должны незамедлительно перестроить свой проект под новую модель. Таким образом, актуальной задачей является разработка платформонезависимого веб-редактора UML-моделей, который бы позволял строить, редактировать, сохранять, а также экспортировать и импортировать модели в стандарте для обмена метаданными XMI. Это бы позволило разработчикам эффективно разрабатывать проекты на разных платформах и быстро реагировать на изменение в моделях. Наличие веб-редактора позволит получить кроссплатформенное решение, не привязанное жестко к программной и аппаратной платформе. Также предполагается интеграция редактора с платформой Flexberry, что предоставит возможность опробовать возможность мгновенной генерации каркаса веб-приложения по созданным в редакторе моделям.

Исходя из всего вышеописанного, можно сформулировать цель работы: необходимо разработать веб-редактор, позволяющий создавать, редактировать, импортировать и экспортировать UML-модели, а также синхронизировать изменение в моделях на сервере Flexberry.

Обзор веб-редакторов показал, что большинство из них представляет собой простую возможность создания диаграмм и сохранения как картинки, без наличия импорта, экспорта в каком-то текстовом формате, задания собственных объектов или свойств. Так, веб-редактор *Diagramo* является неплохим и браузерным редактором диаграмм с открытым исходным кодом. Он позволяет рисовать простые элементы, задавать их свойства. Но у него маленькая функциональность: он умеет только рисовать диаграммы и экспортировать в картинку. Также он тяжело расширяемый и сложно интегрируемый фреймворк, который не подходит для задания логики приложения. *Rapid* – отличный редактор диаграмм, имеет множество возможностей. Например, отрисовка диаграмм, создание собственных объектов, задание определенных свойств, экспорт и импорт из JSON. Также у него есть удобное API, которое можно было бы использовать, но, к сожалению, этот редактор является платным решением. *jsUML2* – легкая библиотека HTML5/javascript с открытым исходным кодом для UML-диаграмм. Она позволяет разработчику легко вставлять UML-диаграммы в веб-приложения, просто вызывая ранее созданные методы в этом редакторе. В ней есть импорт и экспорт в XML. Но чтобы добавлять собственные элементы, нужно дописывать их код. К тому же нет возможности задать свой формат импорта и экспорта данных, валидации.

Поскольку редактор диаграмм позиционируется браузерным приложением, очевидным фактом является то, что он должен быть кросс-браузерным и не требовать от пользователя установки стороннего программного обеспечения, например Adobe Flash Player. Поэтому было решено для отображения диаграмм использовать технологию HTML5, которую поддерживает любой современный браузер. Основным языком разработки браузерных приложений является JavaScript. Наконец, чтобы значительно упростить процесс разработки редактора диаграмм, хотелось воспользоваться библиотекой, которая позволяла бы легко создавать элементы диаграмм, чтобы можно было ими манипулировать. С этой целью было проведено исследование существующих библиотек для рисования элементов диаграмм и получен следующий результат (таблица).

Сравнительная таблица библиотек для рисования диаграмм

Наименование	Бесплатная	Хорошо документирована	Модель MV*	HTML 5 SVG	Проста в использовании
Jgraph	нет	да	нет	да	да
GoJS	нет	да	да	да	да
d3js.org	да	нет	нет	нет	нет
Jointjs	да	да	да	да	да
jsplumb.org	да	да	нет	нет	нет

Из приведенной выше таблицы видно, что библиотека JointJS обладает всеми необходимыми свойствами, и было решено использовать ее для создания редактора диаграмм.

Основным языком разработки браузерных приложений является JavaScript. К сожалению, данный язык обладает рядом минусов. При разработке приложения хотелось бы использовать некоторые шаблоны проектирования, поэтому возникает необходимость использования сторонней JavaScript-библиотеки для создания MVC-приложения. Такой подход позволяет разделить приложения на независимые компоненты. Выбор пал на AngularJS, так как эта библиотека обладает всем необходимым для поддержки MVC, хорошо документирована и проста в использовании.

Обзор существующих JavaScript библиотек, позволяющих работать со стандартом XMI, выявил, что в свободном доступе есть единственная библиотека `escore.js`. Данная библиотека позволяет экспортировать и импортировать модели в формат обмена данными JSON, а также в стандарт для обмена метаданными XMI. `Escore.js` построена на основе пространства имен инструмента для графического моделирования «`eclipse escore tool`». Поэтому после экспорта `uml`-модели можно продолжать с ней работать в среде разработки `eclipse`. Для интеграции редактора с платформой `Flexberry` предлагается расширить стандарт XMI-атрибутами, необходимыми для генерации приложений.

Архитектура веб-редактора представлена на рис. 1. Клиентская часть состоит из визуального редактора диаграмм, а также компонента преобразователя модели в стандарт XMI и составления модели из XMI для редактирования. Модуль экспорта позволяет обмениваться метаданными XMI с сервером, а также выгружать файл с XMI разметкой пользователю на компьютер. Модуль импорта принимает метаданные XMI с сервера или выгружает их из пользовательского

файла, а затем визуализирует модель в редакторе диаграмм. Модуль расширений Flexberry добавляет в документ XMI дополнительные атрибуты из редактора, необходимые для генерации веб-приложений на платформе Flexberry.

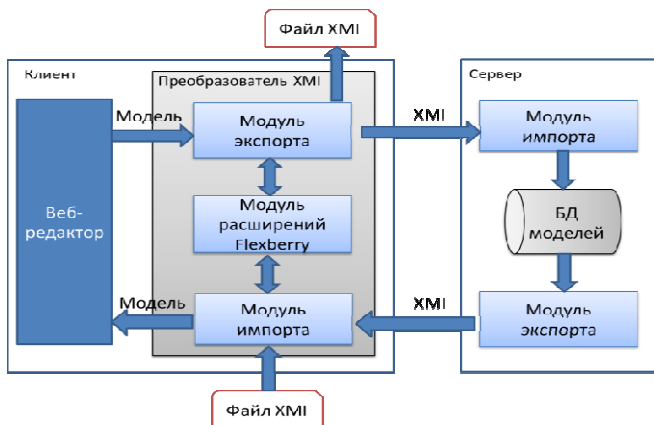


Рис 1. Архитектура веб-редактора

Серверная часть редактора позволяет сохранять и выгружать модели в базу данных.

Для работы с элементами диаграммы был реализован интерфейс веб-редактора (рис. 2).

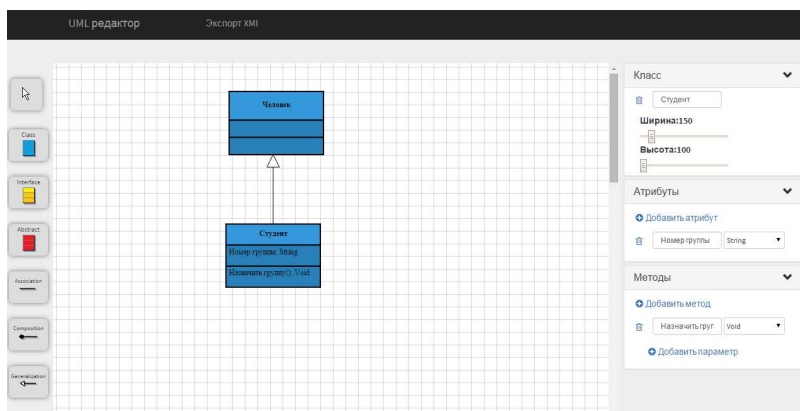


Рис. 2. Интерфейс веб-редактора

Интерфейс включает в себя три области: панель элементов, которая содержит перечень элементов для отображения на диаграмме; область диаграммы, предназначенную для размещения элементов диаграммы; панель работы с атрибутами элементов, позволяющую редактировать имя класса, его размер, добавлять/удалять/редактировать атрибуты класса, назначать им тип данных, а также добавлять/удалять/редактировать методы и их параметры.

Библиографический список

1. Куриленко И.Е., Борисов А.В. Современные архитектурные подходы к построению программного обеспечения // Информационные средства и технологии: сб. тр. XVIII Междунар. науч.-техн. конф. – Т. 2. – М.: Изд. дом МЭИ, 2010. – С. 176–184.

2. XML Metadata Interchange // OMG [Электронный ресурс]. – URL: <http://www.omg.org/spec/XMI/> (дата обращения: 29.11.2015).

3. Timothy J. Grose, Gary C. Doney, Stephen A. Brodsky. Mastering XMI: Java Programming with XMI, XML and UML // John Wiley & Sons. – 480 p.

4. Фаулер М. UML. Основы: пер. с англ. – 3-е изд. – СПб: Символ-Плюс, 2004. – 192 с.

5. JointJS API [Электронный ресурс]. – URL: <http://www.jointjs.com/api> (дата обращения: 29.11.2015).

6. MDA – новая концепция интеграции приложений // Открытые системы [Электронный ресурс]. – URL: <http://www.osp.ru/os/2003/09/183391/> (дата обращения: 29.11.2015).

ПОСТРОЕНИЕ МАТЕМАТИЧЕСКОЙ МОДЕЛИ ТЕХНОЛОГИЧЕСКОГО УЧАСТКА УЧЕБНО-ЭКСПЕРИМЕНТАЛЬНОЙ УСТАНОВКИ ЦЕЛЛЮЛОЗНО-БУМАЖНОГО ПРОИЗВОДСТВА

Аспирант А.А. Каверин, студент Ф.Н. Баянов
Пермский национальный исследовательский
политехнический университет

Рассмотрены основные этапы построения математической модели технологического участка учебно-экспериментальной установки целлюлозно-бумажного производства методом планирования эксперимента. Предложена функциональная схема программно-технического комплекса для автоматизации процесса сбора и обработки экспериментальных данных и выполнена её программная реализация в среде LabVIEW. Разработанный программно-технический комплекс успешно используется для построения математической модели исследуемого объекта в учебном процессе при изучении теории планирования эксперимента.

Ключевые слова: математическая модель; планирование эксперимента; технологический процесс; учебно-экспериментальная установка; экспериментальные данные; уравнение регрессии; программно-технический комплекс.

MATHEMATICAL MODEL DEVELOPMENT OF THE TECHNOLOGICAL SECTION OF THE TRAINING AND EXPERIMENTAL PLANT OF A PULP-AND-PAPER INDUSTRY

Graduate Student A.A. Kaverin, Student F.N. Bayanov
Perm National Research Polytechnic University

The main stages of the mathematical model development of the technological section of the training and experimental plant of pulp-and-paper industry were considered by the experimental design method. The function diagram of a software and hardware complex for automation process of experimental data collecting and its processing was offered. A implementation program of the function diagram was executed in the LabVIEW environment. The developed software and hardware complex is used successfully for mathematical model development of the studied object in department educational process studing the experimental design theory.

Keywords: mathematical model, experimental design, technological process, training and experimental plant, experimental data, regression equation, software and hardware complex.

Технологический процесс производства бумаги состоит из большого количества этапов, на каждом из которых имеется множество задач, связанных с автоматизацией отдельных локальных участков

для улучшения качества производимой бумаги. При этом в производственных условиях, как правило, нет возможности проведения необходимых экспериментальных исследований и наладки локальных контуров управления, так как в режиме работы испытаний неизбежно получение бракованной продукции и высока вероятность выхода из строя оборудования.

Для решения обозначенных проблем, с целью исследования и оптимизации системы управления технологическим процессом подготовки бумажной массы, на кафедре «Микропроцессорных средств автоматизации» совместно с группой предприятий «Пермская целлюлозно-бумажная компания» была создана специальная учебно-экспериментальная установка целлюлозно-бумажного производства [1]. Разработанная установка предназначена для имитации контуров регулирования технологического процесса подготовки бумажной массы и позволяет выполнять экспериментальные исследования, синтез, анализ и настройку локальных систем управления различными технологическими параметрами. Кроме этого установка успешно задействована в учебном процессе кафедры для подготовки специалистов по автоматизации технологических процессов [2].

Физическая модель технологического процесса подготовки бумажной массы на учебно-экспериментальной установке так же, как и на реальном производстве, содержит сложную цепь процессов и контуров управления. А для обеспечения непрерывности и бесперебойности протекания процесса требуется правильная настройка и наладка всех технических средств автоматизации.

Как известно, математическая модель объекта управления может быть получена путем составления математических уравнений, описывающих физику процесса, или эмпирическим путем с помощью экспериментальных исследований [3]. В данной работе предлагается решить задачу построения математической модели рассматриваемого участка, используя методы теории планирования эксперимента [3, 4]. При этом для практического использования методов планирования эксперимента важно учитывать и использовать правила минимизации числа опытов, одновременного варьирования всех выбранных переменных процесса, выбора стратегии для принятия обоснованных решений на каждой стадии экспериментальных исследований. Перед планированием активного эксперимента необходимо собрать дополнительную информацию об исследуемом объекте и выбрать зависимые переменные и входные воздействующие факторы процесса.

Одной из первых задач, которые необходимо решить при использовании теории планирования эксперимента, является выбор плана, на основе которого составляется матрица планирования. В матрицу заносятся основные условия проведения всех опытов, образующих выбранный план, при этом ее строки соответствуют различным опытам, а столбцы отдельным факторам.

В результате анализа различных методов планирования для рассматриваемого случая был выбран ортогональный центральный композиционный план (ОЦКП) второго порядка, так как установлено что в технологическом процессе имеется квадратичная зависимость расхода от частоты вращения сетевого насоса. При этом необходимо учитывать, что для описания поверхности отклика уравнением второго порядка нельзя ограничиваться варьированием факторов только на двух уровнях, так как это не позволяет получить необходимой информации для построения модели [3]. В связи с этим необходимо использовать варьирование факторов на трех или пяти уровнях, а для этого достроить план полнофакторного эксперимента (или его регулярную дробную реплику) до плана второго порядка добавлением к «ядру» определенного количества «звездных» и нулевых точек [4].

В качестве технологического участка для исследования предлагается рассмотреть контур подачи воды [1], который состоит из системы трубопровода с сетевым насосом, двух регулируемых клапанов с электроприводом и датчика расхода. Таким образом, ОЦКП имеет для трех факторов сравнительно небольшое количество опытов $N = 15$ [3]. Варьирование входных факторов представлено в таблице.

Варьирование входных воздействующих факторов процесса

№ фактора	Среднее	Δ	$\Delta\alpha$	$1-\alpha$	$\alpha^2-\alpha$	$-\alpha$
x_1 , Гц	30	10	12	2,7	7,5	-7,3
x_2 , %	50	30	36	8,1	22,5	-21,9
x_3 , %	50	30	36	8,1	22,5	-21,9

В таблице каждый фактор фиксируется на пяти уровнях: $-\alpha$; -1 ; 0 ; 1 ; $+\alpha$, где x_1 – величина задания частоты на частотный преобразователь, x_2 – степень открытия клапана № 1, x_3 – степень открытия клапана № 2, y – величина расхода воды в трубопроводе после сетевого насоса.

Для ОЦКП при числе факторов $n = 3$ получаем следующие параметры плана:

$$N_0 = 2^3 = 8, N = 8 + 1 \cdot 3 + 1 = 15,$$

$$a = \sqrt{\frac{8}{15}} \approx 0,73, \alpha = \sqrt{\frac{1}{2}(\sqrt{15 \cdot 8} - 8)} \approx 1,215. \quad (1)$$

По результатам опытов плана формируем модель в виде полинома:

$$y = b'_0 + b_1 x_1 + b_2 x_2 + b_3 x_3 + b_{12} x_1 x_2 + b_{13} x_1 x_3 + b_{23} x_2 x_3 +$$

$$+ b_{123} x_1 x_2 x_3 + b_4 x_1^2 + b_5 x_2^2 + b_6 x_3^2. \quad (2)$$

Определим коэффициенты полинома согласно выражениям [4]:

$$b_i = \frac{\sum_{u=1}^N x_{iu} Y_u}{\sum_{u=1}^N x_{iu}^2}, \quad b'_0 = b_0 - b_4 a + b_5 a - b_6 a. \quad (3)$$

В результате расчетов полином (1) примет вид:

$$y = 0,39 + 0,15x_1 + 0,11x_2 + 0,12x_3 + 0,04x_1x_2 + 0,05x_1x_3 +$$

$$+ 0,06x_2x_3 + 0,03x_1x_2x_3 + 0,0013x_1^2 + 0,0005x_2^2 + 0,003x_3^2. \quad (4)$$

Проверка однородности дисперсий по критерию Кохрена [3] показала, что расчетное значение критерия для уровня значимости $p = 0,05$ и чисел степеней свободы $f_1 = n - 1 = 4 - 1 = 3$, $f_2 = N = 15$ удовлетворяет условию $G_{\text{расч}} < G_{\text{табл}}$, следовательно, дисперсии однородны.

После оценки значимости коэффициентов уравнения (4) по критерию Стьюдента [3] установлено, что для всех коэффициентов расчетное значение критерия больше табличного, поэтому все коэффициенты значимы и полученное уравнение регрессии не изменяется.

Оценка адекватности полученной модели выполнена по критерию Фишера [3], в результате расчетов расчетное значение критерия Фишера для $p = 0,05$, $f_1 = 12$, $f_2 = 15$ удовлетворяет условию $F_{\text{расч}} < F_{\text{табл}}$, следовательно, уравнение регрессии (4) адекватно эксперименту.

Проведение экспериментов и построение математической модели объекта достаточно сложный и затратный по времени процесс. Поэтому предлагается разработать специальный программно-технический комплекс для автоматизации процессов снятия и обработки экспериментальных данных. Для разработки программной части предлагается использовать среду программирования фирмы Na-

tional Instruments LabVIEW [5] с дополнительным модулем Datalogging & Supervisory Control Module (DSC), предназначенным для графической разработки приложений мониторинга и управления.

Для реализации функций управления и сбора экспериментальных данных в составе автоматизированной системы учебно-экспериментальной установки используется модульный программируемый логический контроллер (ПЛК) Q-System Mitsubishi Electric. Основными компонентами ПЛК являются шасси, модуль центрального процессора, блок питания, модули ввода/вывода аналоговых и дискретных сигналов и ряд коммуникационных модулей для организации промышленной информационно-управляющей сети [6].

Функциональная схема программно-технического комплекса для проведения экспериментальных исследований представлена на рис. 1.

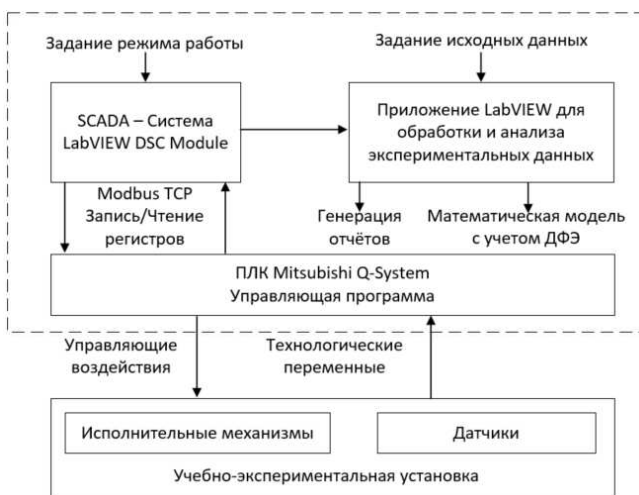


Рис. 1. Функциональная схема программно-технического комплекса

Для запуска процедуры обработки экспериментальных данных пользователь должен загрузить полученный на этапе снятия данных файл и запустить алгоритм построения уравнения регрессии и статистической проверки полученных результатов.

Интерфейс программного комплекса для экспериментальных исследований (рис. 2) достаточно прост и информативен, что позволяет пользователю за короткий промежуток времени разобраться в элементах управления.

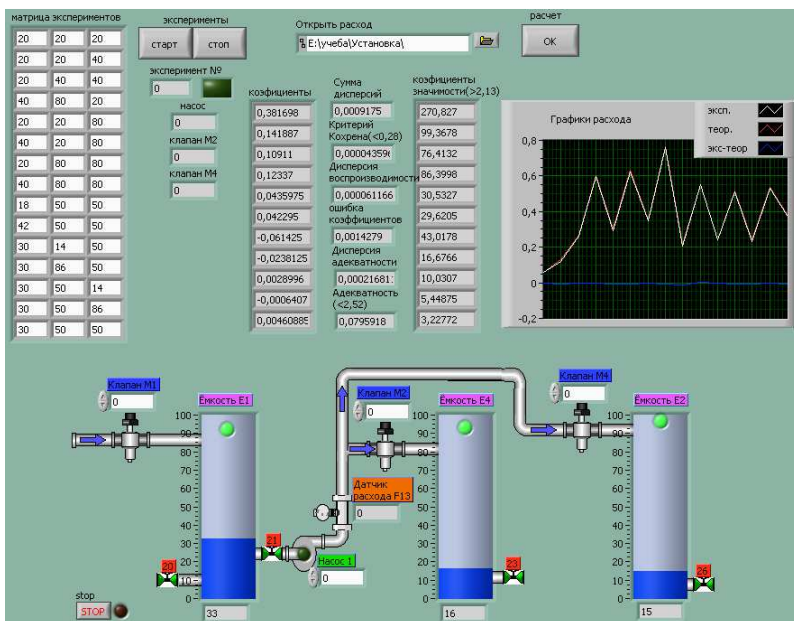


Рис. 2. Интерфейс программного комплекса

В левом верхнем углу располагается панель для управления и наблюдения за процессом снятия данных, где для наглядности отображаются матрица эксперимента, номер текущего эксперимента и значения входных факторов, которые передаются в текущий момент времени на контроллер. По окончании всех опытов программа предложит пользователю указать путь для сохранения файла с результатами эксперимента. Для наглядности результаты сводятся в график, на котором отображаются значения экспериментального расхода и теоретического, полученного по составленному уравнению регрессии.

Таким образом, разработанный программно-технический комплекс позволяет в автоматическом режиме построить математическую модель выбранного участка технологического процесса методом планирования эксперимента. Внедрение представленной в работе системы экспериментальных исследований позволило, с одной стороны, получить необходимую для синтеза системы управления математическую модель объекта, а с другой – полученные результаты

могут быть использованы и в учебных целях на лабораторных занятиях при изучении теории планирования эксперимента.

В дальнейшем планируются развитие и модернизация программно-технического комплекса с добавлением в систему других локальных участков установки. Кроме этого запланирована разработка нового специального модуля для выполнения процедуры синтеза и анализа локальных контуров управления.

Библиографический список

1. Разработка экспериментальной установки имитации контуров регулирования технологических процессов производства бумаги предприятия ООО «Пермский картон» / Д.А. Даденков, Л.В. Поносова, А.Б. Петроченков, И.Г. Друзьякин, А.Г. Лейсле // Энергетика. Инновационные направления в энергетике. CALS-технологии в энергетике: материалы 3-й Всерос. науч.-техн. интернет-конф.; 2–30 ноября 2009 г. – Пермь, 2009.

2. Даденков Д.А., Петроченков А.Б. Опыт создания лабораторно-тренажерного комплекса для подготовки специалистов в области автоматизированных систем управления технологическими процессами // Научно-технические ведомости СПбГПУ. – 2009. – № 5(87). – С. 251–255.

3. Сидняев Н.И. Теория планирования эксперимента и анализ статистических данных. – М.: Юрайт, 2013. – 400 с.

4. Тихомиров В.Б. Планирование и анализ эксперимента. – М.: Легкая индустрия, 1974. – 262 с.

5. Пейч Л., Точилин Д., Поллак Б. LabVIEW для новичков и специалистов. – М.: Горячая линия-Телеком, 2004. – 384 с.

6. Даденков С.А., Кон Е.Л., Южаков А.А. Алгоритм проектирования промышленной информационно-управляющей сети требуемой производительности (на примере технологии LonWorks) // Вестник Поволж. гос. технолог. ун-та. Сер. Радиотехнические и инфокоммуникационные системы. – 2014. – № 5(24). – С. 35–43.

МИКРОПРОЦЕССОРНАЯ СИСТЕМА УПРАВЛЕНИЯ ТЕМПЕРАТУРОЙ КАРТОФЕЛЕХРАНИЛИЩА

Магистрант Е.В. Кацко

Пермский национальный исследовательский
политехнический университет

В статье рассматривается модификация лабораторного стенда «Картофелехранилище» с использованием программируемого логического контроллера (ПЛК) LOGO. Проведен активный эксперимент (АЭ) для определения кривых нагрева объекта управления (ОУ), а также параметров передаточной функции этого объекта. Составлена программа для системы автоматического регулирования (САР) температурой данного стенда, описаны некоторые особенности программирования в среде LOGO! Soft Comfort V 6.1 (LSC).

Ключевые слова: лабораторный стенд калорифер, картофелехранилище, система автоматического регулирования, температура, программируемый логический контроллер, LOGO! Soft Comfort, кривая нагрева.

THE MICROPROCESSOR CONTROL SYSTEM OF POTATO STORAGE TEMPERATURE

Master Student E.V. Katsko

Perm National Research Polytechnic University

In the article the modification of "potato storage" laboratory stand is considered with using the LOGO programmable logic controller (PLC). Active experiment (AE) for heating curves definition of the control object (CO) as well as parameters of CO's transfer function is carried out. The program for automatic regulation system of stand temperature is designed, and some programming features in IDE of LOGO! Soft Comfort V 6.1 (LSC) are described.

Keywords: laboratory stand, heater, potato storage, automatic regulation system, temperature, programmable logic controller, LOGO! Soft Comfort, heating curve.

Лабораторной стенд «Картофелехранилище» наряду с другими стендами кафедры «Автоматика и телемеханика» Пермского национального исследовательского политехнического университета используется для проведения лабораторных работ по дисциплине «Локальные системы управления». Данный стенд можно рассматривать как модель реального объекта управления, с помощью которого студенты могут формировать навыки программирования ПЛК и наглядно наблюдать за процессом управления. На данном этапе решается основная задача, связанная с модернизацией аппаратной части стенда (в части ПЛК).

Конструктивно стенд представляет собой модель картофелехранилища (рис. 1), на одном из торцов которого выполнены два разреза для воздушных заслонок и один сверху – для соединения с воздушным каналом приточной вентиляции. Также в корпусе сделано еще одно отверстие для вала, который приводится в движение исполнительным механизмом (ИМ). Весь стенд размещен на деревянном основании, внутри которого находятся все необходимые электронные компоненты.

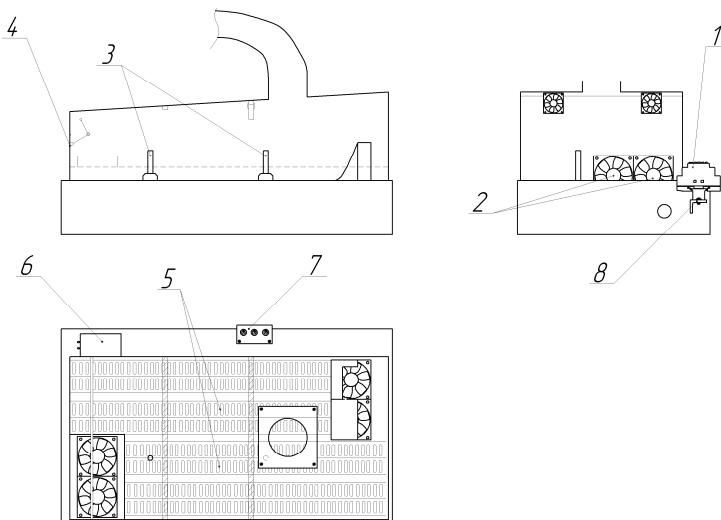
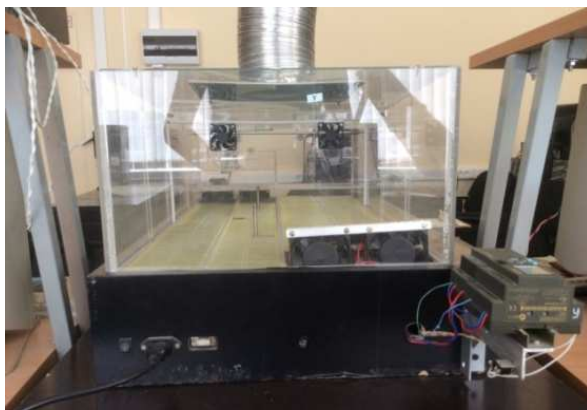


Рис. 1. Конструкция лабораторного стенда «Картофелехранилище»

Лабораторный стенд оснащен следующей элементной базой: ПЛК LOGO! 12/24 RC (позиция 1); двумя датчиками температуры ТПТ-15-2-100П – платиновые термометры сопротивления (позиция 3); шестью вентиляторами (позиция 2); двумя гибкими ленточными нагревательными элементами (ЭНГЛ, позиция 5); электроприводом с валом (позиция 6); воздушным клапаном (заслонка или шибер) (позиция 4); панелью для ручного управления стендом (позиция 7); кнопкой ручного управления электроприводом (позиция 8).

Автоматизированный стенд «Картофелехранилище» позволяет осуществлять функции регулирования температуры через управление вентиляторами и положением воздушной заслонки (рис. 2, а, б).

Процессы управления уместно рассматривать для двух случаев: с отсутствием возмущающего воздействия и с его наличием [4]. Во втором случае используется дополнительно стенд «Калорифер» (рис. 3).



а



б

Рис. 2. Конструкция стенда «Картофелехранилище» (*а*), вид сбоку (*б*)



Рис. 3. Конструкция стенда «Калорифер»

Управление стендом «Картофелехранилище» может выполняться в ручном режиме, для этого предусмотрена панель с набором тумблеров.

Стенд «Картофелехранилище» является объектом управления первого порядка и представляет апериодическое звено с параметром постоянной времени $T_{\text{карт}}$.

Структурная схема стенда представлена на рис. 4.

Задающее воздействие на ОУ – постоянный нагрев внутри картофелехранилища от ЭНГЛ.

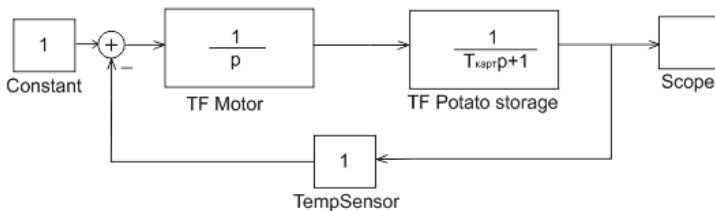


Рис. 4. Структурная схема стенда «Картофелехранилище»

Для определения параметров: коэффициента усиления регулирующего органа (K_{po}), коэффициента усиления ОУ ($K_{карт}$) и постоянной времени ОУ ($T_{карт}$) был проведен активный эксперимент (АЭ), результаты которого представлены в таблице.

Результаты активного эксперимента

t , мин	Вид АЭ		
	АЭ с учетом внешнего возмущающего воздействия		АЭ с задающим воздействием (без внешнего возмущающего воздействия)
	$T_{кал}$, °С	$T_{картоф}$, °С	$T_{картоф}$, °С
0	30	30	29,4
1	32	30,1	29,9
2	33	30,2	30,5
3	34	30,3	31,1
4	34,9	30,4	31,5
5	35,6	30,4	32,1
6	36,4	30,6	32,5
7	36,85	30,7	33,1
8	37,45	30,8	33,6
9	37,87	30,9	33,9
10	38,27	31	34,2
11	38,43	31	34,5
12	38,68	31	34,8
13	38,9	31,1	35,2
14	39,1	31,1	35,3
15	39,25	31,1	35,5
16	39,3	31,2	35,8
17	39,4	31,2	36

Ввиду отсутствия информации о мощности ЭНГЛ в настоящей работе расчет упомянутых параметров производился на основании существующего метода [1] и только для случая, когда АЭ происходит с учетом дополнительного внешнего воздействия. Значения этих параметров составили: $K_{po} = 1,03$; $K_{карт} = 1$; $T_{карт} = 1,9$ мин.

Дальнейшим этапом работы стало написание программы для ПЛК LOGO! 12/24 RC. Основная идея программы заключается в том, что ПЛК совершает опрос датчиков температуры и сравнивает текущее значение с уставками (верхний и нижний порог срабатывания), заданными пользователем. В случае выхода текущего значения температуры за диапазон уставок происходит включение/отключение релейных выходов ПЛК.

Моделирование коммутационной программы на контроллере показало следующие результаты:

- 1) инерционность процесса управления;
- 2) неинформативное отображение измеренной температуры (на ПК и ПЛК): значения отображаются в служебных единицах LOGO!.

Поэтому для разрешения этих неудобств программа была скорректирована (рис. 5) в части изменения настроек функционального блока «Аналоговый пороговый выключатель» (диапазон от -45 до 55 сменился на новый: от 0 до 4000 , а также значения порогов на ВКЛ и ОТКЛ изменены на значения 3330 и 3320 соответственно). Данные изменения позволили отслеживать температуру с точностью до сотых, а также избавиться от отображения значения температуры в служебных единицах. Но сужение диапазона (с 1 до $0,1^{\circ}\text{C}$) привело к тому, что стало происходить частое переключение релейных выходов ПЛК, поэтому для устранения этого было решено внести блоки задержек.

Параметр	Комментарий
Имя блока	
Датчик	0 - 10 В
Диапазон измерения	
Минимум	0
Максимум	4000
Параметры	
Усиление	4,00
Смещение	0
Порог	
Вкл.	3330
Откл.	3320
Число десятичных знаков в тексте сообщения	
	0 +12345
<input type="checkbox"/> Защита включена	

Рис. 5. Измененные настройки функционального блока «Аналоговый пороговый выключатель»

Чтобы убедиться в наличии автоколебаний в данной САР было решено фиксировать временные точки достижения нижних (T_{LOW}) и верхних (T_{HI}) порогов. В результате этого были построены графики

автоколебаний САР в случае внешнего возмущения (рис. 6) и постоянного задающего воздействия (рис. 7).

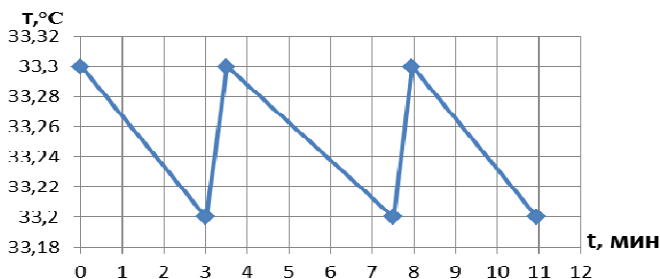


Рис. 6. Автоколебания САР в случае внешнего возмущения ($P_{нар} = 500$ Вт)

Стоит отметить, что вид графиков идеализирован, так как сами графики строились исключительно по времени достижения верхнего и нижнего порогов без промежуточных точек. Несмотря на это, на графиках подтверждается режим автоколебаний в САР с параметрами амплитуды $\left(\frac{T_{HI} - T_{LOW}}{2}\right)$ и частоты $(1/t_{cp})$, где t_{cp} – средний период на всем интервале работы САР).

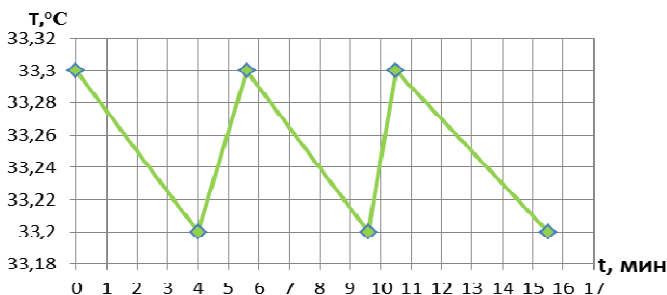


Рис. 7. Автоколебания САР в случае постоянного задающего воздействия

В результате работы проведен АЭ по определению кривых нагрева ОУ, по данным которых определили параметры передаточной функции ОУ и регулирующего органа. Разработана программа для ПЛК в среде LSC, позволяющая выполнять алгоритм по регулированию температуры внутри стэнда «Картофелехранилище» в заданном от пользователя диапазоне температур.

Дальнейшее развитие данного стенда связано с реализацией программного обеспечения верхнего уровня, например, разработанного на основе SCADA-пакета [2] или универсальной среды разработки (как, например, в работах [3, 5]).

Библиографический список

1. Кон Е.Л., Хижняков Ю.Н., Южаков А.А. Локальные системы управления параметрами объектов гражданского и промышленного строительства: учеб. пособие. – Пермь: Изд-во Перм. нац. исслед. политехн. ун-та, 2012. – 156 с.

2. Кавалеров М.В. К вопросу о термине «SCADA-система» // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2011. – № 5. – С. 205–209.

3. Кавалеров М.В., Матушкин Н.Н. Мехатронная система «шарик на балансирах с веб-камерой» для исследований на стыке компьютерного зрения и планирования задач реального времени // Управление в технических, эргатических, организационных и сетевых системах: материалы конф. / под ред. С.Н. Васильева, И.А. Каляева, Д.А. Новикова, Г.Г. Себрякова. – Пермь, 2012. – С. 744–747.

4. Лукас В.А. Теория управления техническими системами: учеб. пособие для вузов. – 4-е изд., испр. – Екатеринбург: Изд-во УГГУ, 2005. – 677 с.

5. Софин Н.А., Кавалеров М.В. Модернизация программного обеспечения лабораторного стенда «Модель вентиляционной системы» // Инновационные технологии: теория, инструменты, практика: матер. междунар. конф. – 2014. – Т. 2. – С. 459–465.

ВНЕДРЕНИЕ СОВРЕМЕННОГО ИНФОРМАЦИОННОГО РЕШЕНИЯ В ОБЛАСТИ ПОДГОТОВКИ ПРОИЗВОДСТВА НА КРУПНЫХ ПРОМЫШЛЕННЫХ ПРЕДПРИЯТИЯХ

**Студент А.А. Клещевникова,
канд. техн. наук, доцент А.Б. Петроченков**
Пермский национальный исследовательский
политехнический университет

Актуальность данной темы имеет большой потенциал в развитии промышленности России. Не всегда уровень автоматизации технологических решений достигает высоких результатов. Этот недостаток обусловлен непростым процессом согласования технологической и конструкторской документации. Для того чтобы оставаться конкурентоспособным предприятием, необходимо развиваться в одну ногу со временем и внедрять новейшие технологии в производство. Программа Teamcenter компании Siemens PLM Software является тем самым важным инструментом для поддержания высокого уровня процесса подготовки производства. В данной статье рассмотрены проблемы подготовки производства и показаны основные преимущества использования программы Teamcenter.

Ключевые слова: конструкторская документация; подготовка производства, Siemens PLM Software, Teamcenter, технологическая проработка.

THE INTRODUCTION OF MODERN INFORMATION AND DECISION IN THE FIELD OF PREPARATION OF PRODUCTION IN LARGE INDUSTRIAL ENTERPRISES

**Student A.A. Kleshchevnikova,
Ph.D. in Technical Sciences, Associate Professor A.B. Petrochenkov**
Perm National Research Polytechnic University

The relevance of this topic has great potential in the development of industry in Russia. Not always the level of automation of technological solutions achieve good results. This disadvantage is due to the difficult process of agreeing technological and design documentation. In order to remain competitive the company needs to develop in a single step with the times and implement the latest technology in production. Teamcenter software from Siemens PLM Software is the most important tool for maintaining a high level of process of production preparation. This article deals with the problems of preparation of production and shows the main advantages of using the Teamcenter software.

Keywords: design documentation, preparation of production, Siemens PLM Software, Teamcenter, serial enterprise, technological study.

Введение. В современном мире невозможно представить процессы подготовки производства и согласования чертежей (спецификаций), а также другой конструкторской документации без использования современных информационных решений. Очень важно применять именно современные и конкурентоспособные программы, которые помогут отслеживать все этапы жизненного цикла изделия.

Продукт компании Siemens PLM Software – программа Teamcenter является одной из программ высокого уровня, предназначенной для управления производственными процессами и инженерными данными на всем протяжении жизненного цикла изделия. Teamcenter является основной платформой для создания единой информационной системы управления подготовкой производством [1].

Данная программа используется уже на многих крупных промышленных предприятиях всего мира. Программа Teamcenter является уникальным продуктом, в функционале которой присутствуют огромный багаж приложений и удобный интерфейс для каждого пользователя. Это одна из самых распространенных программ среди промышленных предприятий всего мира. Teamcenter применим почти во всех сферах деятельности как на авиастроительных предприятиях, так и в сельхозпромышленности. С помощью данной программы все данные об изделии находятся под контролем, а сотрудники предприятий имеют быстрый и удобный доступ к конструкторской и технологической документации и спецификациям [2]. Актуальность данной темы имеет большой потенциал в развитии промышленности России. Применение программы Teamcenter поможет многим промышленным предприятиям стать более конкурентоспособными на рынке, а также реализовать свои идеи и решения по выпуску более качественной продукции [3].

Проблемы подготовки производства на крупных промышленных предприятиях. К сожалению, уровень автоматизации на крупных промышленных предприятиях России далек от уровня предприятий других стран. Данное отставание объясняется непростыми и долговременными процессами согласования документации, сборки и эксплуатации изготовленной продукции. Еще одним недостатком процесса подготовки производства является очень сложный процесс цехозаходов и, соответственно, этапов обработки и изготовления деталей. А также при технологической проработке деталей немаловажной

является информация о размерах деталей, средствах технологического оснащения, а также информация о способах получения исходной заготовки, о технологических базах, разработке маршрута обработки деталей [4]. Всю эту информацию сотрудники предприятия находят в конструкторской документации и чертежах. Однако инструментарий в области проектирования, сборки и эксплуатации давно устарел, и его недостаточно для отслеживания всего жизненного цикла изделия. Следовательно, не всегда сотрудник предприятия владеет актуальной информацией об изделии и легко может ошибиться в выборе необходимого размера или в выборе оснастки для изготовления деталей, отсюда у предприятия получается большой процент брака.

Существуют два способа решения данных проблем: внедрение нового программного обеспечения в подготовку производства или разработка более современного процесса создания технологий и согласования конструкторской документации.

Внедрение современного информационного решения в области подготовки производства на крупных промышленных предприятиях. Внедрение продукта Teamcenter позволит предприятию получить высокие PLM-преимущества и следовать поставленным целям без срывов сроков и без большого процента брака. Во-первых, значительно сокращаются сроки на подготовку производства. Во-вторых, 3D-моделирование дает предприятию возможность легко рассмотреть изделие во всех изометриях, увидеть расположение и местонахождение других объектов относительно главного изделия [5]. И, в-третьих, существенно повысится качество проектирования и качество изготовления продукции.

Если конкретно говорить о преимуществах программы Teamcenter, которые она дает крупному промышленному предприятию, то можно выделить следующие:

- сокращается объем хранимых данных за счет повторяющейся информации;
- сокращаются сроки на согласование необходимой документации;
- сокращаются сроки на проектирование, сборку и эксплуатацию продукции;
- сокращаются сроки на обнаружение брака в производстве продукции;
- наличие всегда актуальных данных о продукции;

- простой и удобный функционал для каждого пользователя;
- возможность отслеживать все этапы процессов;
- повышается уровень конкурентоспособности предприятия.

Основными приложения программы Teamcenter, которые участвуют в процессе подготовки производства, являются:

- планировщик процессов сборки;
- конструктор процессов;
- просмотр процессов;
- классификатор;
- организационная структура;
- мой Teamcenter;
- визуализация.

С их помощью можно свободно и без особых усилий построить процессы согласования производственной документации, проектирования и наблюдения за состоянием процессов [6], а также с помощью визуализации процесс сборки ускоряется в десятки раз.

Применение данной программы также позволит предприятию поднимать уровень профессионализма у молодых специалистов, поскольку программа Teamcenter нацелена в большей степени на использование более современных информационных технологий, следовательно, выпускникам вузов и техникумов будет интересно познавать инновационные технологии и полностью погружаться в интересную работу. Накопленный потенциал знаний и практика молодых специалистов помогут предприятию вырастить новый, сильный и конкурентоспособный персонал, который в будущем будет продолжать модернизировать процессы подготовки производства.

Таким образом, внедрение программы Teamcenter на крупном промышленном предприятии открывает для него значительные перспективы и большие внутренние резервы, позволяет значительно повысить качество производственных процессов и сократить издержки [7]. Конечно, для полноценной работы данной программы предприятию понадобится приложить большие усилия, но они, несомненно, окажутся вероятно полезными и в скором времени дадут предприятию выйти на новый, более высокий уровень в автоматизации технологических решений и поможет предприятию с выпуском более качественной и совершенной продукции.

Библиографический список

1. Когогин С.А., Мингалеев Г.Ф., Силенов М.А. Управление конструкторско-технологической подготовкой производства изделий на основе информационной системы «Teamcenter» // Поиск эффективных решений в процессе создания и реализации научных разработок в авиационной и ракетно-космической промышленности: материалы междунар. науч.-практ. конф. – 2014. – С. 634–636.

2. Тороп Д.Н., Терликов В.В. Teamcenter. Начало работы. – М.: ДМК-Пресс. – 2011. – С. 215–217.

3. Автоматизация внесения архива конструкторско-технологической документации в PLM систему Teamcenter / М.И. Чижов, А.В. Бредихин, Т.О. Хаустова, Н.А. Гирчев // Вестник Воронеж. гос. техн. ун-та. – 2012. – № 1. – С. 38–40.

4. Siemens Industry Software [Электронный ресурс] // Teamcenter Global Services: Mult-Site Collaboration. – 2015. – Vol. 59. – P. 1–3. – URL: http://m.plm.automation.siemens.com/en_us (дата обращения: 09.10.2015).

5. Teamcenter Engineering Product Data Management Student Guide October 2006 MT14330 – Version 2005 SR1MT.

6. Яблочников Е.И., Молочник В.И., Фомина Ю.Н. Реинжиниринг бизнес-процессов проектирования и производства: учеб. пособие. – СПб: Изд-во СПбГУИТМО, 2008. – 152 с.

7. Diedrich C., Muhlhouse M. Basic concept of the Digital factory // AT: Automatisierungstechnik. – 2011. – Vol. 59. – P. 18–25.

8. Model Based Enterprise [Электронный ресурс]. – URL: <http://model-based-enterprise.org/default.aspx> (дата обращения: 09.10.2015).

АЛГОРИТМ ПРЕОБРАЗОВАНИЯ ПЛОСКИХ ЗАГОТОВОК В ОБЪЕМНЫЕ ДЛЯ ТРЕХМЕРНОЙ ГРАВИТАЦИОННОЙ МОДЕЛИ

Аспирант Д.А. Лялин

Березниковский филиал Пермского национального
исследовательского политехнического университета

В статье предложен алгоритм преобразования плоских заготовок в объемные для трехмерной гравитационной модели. Описаны входные и выходные данные. Приведен верный алгоритм для получения триангуляции Делоне.

Ключевые слова: задача раскроя-упаковки, трехмерная гравитационная модель, трехмерные заготовки, триангуляция Делоне.

ALGORITHM OF CONVERSION FLAT WORKPIECE IN BULK FOR THREE-DIMENSIONAL GRAVITY MODEL

Graduate Student D.A. Lyalin

Berezniki branch of the Perm National Research Polytechnic University

In this article proposed the algorithm of conversion flat workpiece in bulk for three-dimensional gravity model. Described the input and output data. Shown harp algorithm for receiving Delaunay triangulation.

Keywords: cutting-packing problem, three-dimensional gravity model, three-dimensional workpiece, Delaunay triangulation.

Как известно, основные методы решения задачи фигурной нерегулярной укладки плоских геометрических объектов на листовых материалах используют двумерные заготовки [1]. Трехмерная гравитационная модель для укладки геометрических объектов использует объемные заготовки [2]. Физический движок модели позволяет создавать в физическом мире трехмерные объекты, состоящие из треугольников. Для преобразования двумерной заготовки в набор треугольников объемной заготовки используется следующий алгоритм:

- получение набора треугольников из набора точек двумерной заготовки с помощью триангуляции;
- вытягивание полученного набора треугольников по оси OZ ;
- генерация набора соединительных треугольников.

Триангуляция – это декомпозиция многоугольника P на множество треугольников, внутренние области которых попарно не пересекаются и объединение которых в совокупности составляет P . В строгом

смысле слова, вершины этих треугольников должны совпадать с вершинами исходного многоугольника. Задача построения триангуляции является одной из базовых в вычислительной геометрии на плоскости. Среди всех возможных классов триангуляций на практике чаще всего используется триангуляция Делоне. Триангуляцию называют триангуляцией Делоне, если внутри окружности, описанной вокруг любого построенного треугольника, не попадает ни одна из заданных точек триангуляции [3]. Ее трудоемкость составляет $O(n \log n)$ арифметических операций. Существуют алгоритмы, достигающие этой оценки в среднем и худшем случаях. Кроме того, известны алгоритмы, позволяющие в ряде случаев достичь в среднем $O(n)$. Подробно многие алгоритмы рассмотрены и классифицированы, например, в работе [4]. Там же приводятся оценки их эффективности и трудоемкости.

Гравитационная модель использует верный алгоритм построения триангуляции Делоне. В верном алгоритме триангуляции вначале из исходных точек выбирается та, которая находится как можно ближе к центру масс всех точек. Далее для остальных точек вычисляется полярный угол относительно выбранной центральной точки, и все точки сортируются по этому углу. Затем все точки соединяются рёбрами с центральной точкой и соседними в отсортированном списке. И в заключение производится полное перестроение триангуляции для выполнения условия Делоне.

На вход алгоритма триангуляции подается плоская заготовка, описанная в файле DXF (рис. 1, а). После получения триангуляции плоской заготовки (рис. 1, б) набор вершин и треугольников дублируется с измененной координатой Z на необходимое значение (рис. 1, в). Затем вершины двух заготовок объединяются с помощью соединительных треугольников (рис. 1, г).

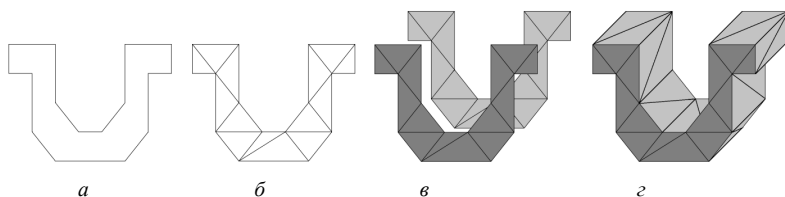


Рис. 1. Верный алгоритм триангуляции Делоне: исходная плоская заготовка (а); триангуляция заготовки (б); дублирование заготовки с изменением координаты OZ (в); объединение с помощью соединительных треугольников (г)

В результате работы алгоритма преобразования плоских заготовок в объемные получается набор вершин в формате:

$$\begin{aligned}
 &V1_x, V1_y, V1_z, \\
 &V2_x, V2_y, V2_z, \\
 &\dots \\
 &Vn_x, Vn_y, Vn_z,
 \end{aligned}$$

где Vn_x – координата X вершины n , Vn_y – координата Y вершины n , Vn_z – координата Z вершины n , набор треугольников в формате:

$$\begin{aligned}
 &N_{51}, N_{15}, N_{98}, \\
 &N_{33}, N_{68}, N_{56}, \\
 &\dots \\
 &N_{67}, N_{18}, N_{90},
 \end{aligned}$$

где N_n – порядковый номер вершины из набора вершин. В этом наборе указывается три номера вершин, по которым нужно построить треугольник. После получения набора вершин и набора треугольников объемной заготовки данные передаются в физический движок гравитационной модели. На основе полученных данных движок создает замкнутое объемное тело в физическом мире, присваивает ему массу по умолчанию $m = 1$, коэффициент трения $\mu = 0$ и коэффициента восстановления $k = 0$ [2]. Алгоритм преобразования применяется не только к созданию заготовок, но и к созданию листов. Лист также загружается из файла DXF в двумерной форме (рис. 2, *а*), затем происходят вытягивание его границ на указанную ширину, затем происходят триангуляция, дублирование и объединение соединительными треугольниками. В итоге в физическом мире создается тело, показанное на рис. 2, *б*.

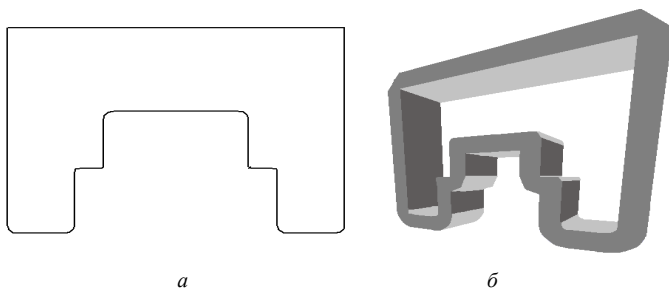


Рис. 2. Исходный лист (*а*), преобразованный объемный лист (*б*)

Алгоритм преобразования плоских заготовок в объемные используется в гравитационной модели при загрузке всех элементов физического мира. После преобразования есть возможность сохранить элементы в формате OBJ – это простой формат данных, который содержит только трехмерную геометрию, а именно позицию каждой вершины и параметры, которые создают полигоны [5].

Библиографический список

1. Мурзакаев Р.Т., Шилов В.С., Буркова А.В. Основные методы решения задач фигурной нерегулярной укладки плоских деталей // Инженерный вестник Дона. – 2013. – Т. 27. – № 4. – С. 122–130.

2. Лялин Д.А. Трехмерная гравитационная модель для решения задач фигурного раскроя-упаковки материалов // Решения: материалы четвертой Всерос. науч-практ. конф. – Пермь: Изд-во Перм. нац. иссл. политехн. ун-та, 2015. – С. 86–87.

3. Препарата Ф., Шеймос М. Вычислительная геометрия: Введение: пер. с англ. – М.: Мир, 1989. – 478 с.

4. Скворцов А.В. Обзор алгоритмов построения триангуляции Делоне // Вычислительные методы и программирование. – 2002. – Т. 3. – С. 14–39.

5. Wavefront OBJ File Format Summary – Fileformat.info. – URL: <http://www.fileformat.info/format/wavefrontobj/egff.htm> (дата обращения: 22.11.2015).

ДЕТЕКТИРОВАНИЕ ОБЪЕКТА УПРАВЛЕНИЯ С ПОМОЩЬЮ КОМПЬЮТЕРНОГО ЗРЕНИЯ В СИСТЕМЕ «ШАРИК НА БАЛАНСИРЕ»

Студент Н.Г. Макагонов

Пермский национальный исследовательский
политехнический университет

В статье рассмотрены аспекты автоматизации поиска балансира на изображении. Поиск производится с помощью обученных LBP-каскадов открытой библиотеки OpenCV. В результате достигнут необходимый уровень качества и скорости распознавания.

Ключевые слова: OpenCV, шарик на балансира, LBP-каскад.

DETECTION OF OBJECT MANAGEMENT WITH COMPUTER VISION IN «BALL AND BEAM» SYSTEM

Student N.G. Makagonov

Perm National Research Polytechnic University

In this article the aspects of automation search beam in the image are considered. The search is performed using the trained LBP-cascades, open source library OpenCV. As a result the required level of quality and speed of recognition are achieved.

Keywords: OpenCV, ball and beam, LBP-cascade.

В настоящий момент робототехника является одной из наиболее динамично развивающихся областей науки и техники. Уже сейчас на производстве роботы обеспечивают полный цикл сборки, выполняют функции личного ассистента, самостоятельно очищают помещения, общаются с людьми в торговых центрах. Зрение является одним из основных источников получения информации об окружающем мире, поэтому визуальное управление играет большую роль в процессе функционирования робота. Входные сведения об объекте и процессе управления в таких системах представляются в качестве видеoinформации.

Одной из классических задач компьютерного зрения является задача детектирования и выделения определенного объекта в видеопотоке либо на отдельном изображении. Существуют различные методы решения этой задачи. Некоторые из них обладают малой ресурсоемкостью, что особенно актуально в области мобильных

роботов, к таким методам можно отнести, например, обнаружение объекта по цвету. Однако эти методы не всегда дают точный результат, и, прежде чем его дать, либо требуют долгой предварительной настройки детектируемого объекта, или параметров регулятора. Для получения требуемого качества часто используются более сложные вычисления, направленные на преобразования и анализ получаемой видеoinформации.

Данная статья посвящена модификации уже существующей системы автоматического управления типовым объектом: «шарик на балансира» [1]. В изначальном варианте данная система нуждалась в предварительной калибровке положения балансира в кадре. Такое требование накладывало жесткие ограничения на взаимное положение камеры и балансира. Определение местоположения шарика осуществлялось за счет детектирования, светлого цвета шарика на темном фоне внутренней части П-образной балки балансира. Таким образом, использование, например, шарика другого цвета может привести к невозможности определения его положения без изменения внутренних параметров детектирующего алгоритма. Однако простота алгоритмов является огромным преимуществом, во-первых, обеспечивая достаточно хорошую скорость реакции системы на изменение положения шарика, во-вторых, снижая требования к аппаратной части управляющей системы.

Альтернативный вариант, который может позволить справиться с ресурсоемкими задачами компьютерного зрения при условии ограниченных аппаратных ресурсов – это удаленная обработка видеoinформации [2, 3]. Также этот способ может пригодиться для организации гибридной обработки, когда только часть видеoinформации обрабатывается удаленным компьютером. Поэтому применение систем распознавания объектов на изображениях является вполне оправданным и эффективным способом реализации систем управления, особенно тех, которые имеют в качестве сигнала обратной связи в основном только видеoinформацию. При этом, конечно, надо учитывать возрастающие задержки, обусловленные временем, затрачиваемым на выполнение задач компьютерного зрения, с помощью соответствующих оценок [4], что позволит выполнять эффективное планирование задач реального времени [5].

Данное исследование решает задачу автоматизации поиска балансира в кадре. Таким образом исчезает потребность в предвари-

тельной калибровке взаимного положения камеры и балансира, а также возникает возможность перемещения камеры, не вызывая при этом остановки процесса регулирования. При этом также ставится задача не допускать значительного увеличения времени обработки видеoinформации.

Таким образом, решаемая задача сводится к поиску некоторого заранее определенного объекта на изображении. Наиболее перспективным и известным методом решения данной задачи (в условиях ограниченных аппаратных ресурсов) являются алгоритмы на основе каскадов признаков, наиболее известными из которых являются следующие два алгоритма:

- алгоритм Виолы и Джонса [6];
- алгоритм LBP (Local Binary Patterns) [7].

Данные алгоритмы изначально создавались для распознавания лиц, однако впоследствии показали свою применимость практически к любым видам объектов, таким как человеческая фигура, автомобили, цифры. Поскольку система должна обеспечивать достаточный уровень быстродействия, то авторами был выбран алгоритм LBP, так как он является более быстрым в сравнении с алгоритмом Виолы–Джонса, хотя и требует более тонких настроек для обеспечения сопоставимого уровня распознавания. Данные алгоритмы выполняют детектирование на основе предварительно обученных классификаторов. Классификатор – это аппроксимирующая функция, определяющая принадлежность либо непринадлежность некоторого объекта к определенному классу. Обучение классификаторов производится на основе обучающей выборки, состоящей из изображений объектов двух типов: принадлежащих детектируемому классу и не принадлежащих данному классу. Полученные классификаторы объединяются особым образом и образуют так называемый каскад, который впоследствии используется программой-распознавателем.

В данном исследовании использовалась свободно распространяемая библиотека компьютерного зрения OpenCV. Процесс обучения реализован достаточно удобно, для обучения, как было изложено ранее, требуются 2 набора изображений. Кроме того, требуется файл в формате «.dat» с перечислением всех путей к используемым отрицательным изображениям, а также специальный файл в формате «.vec», генерируемый на основе положительных изображений, и файла в формате «.dat» с описанием путей к данным файлам, местоположения классифицируемого

объекта на них. В процессе формирования данного файла производится приведение всех положительных изображений к единому формату, другими словами, сжатие до единого заранее задаваемого размера. Библиотека содержит исполняемые файлы как для создания самого каскада (`opencv_traincascade`), так и для приведения положительных изображений к единому формату (`opencv_createsamples`).

Обучаемые каскады являются объектами класса `CascadeClassifier`. Детектирование производится посредством метода данного класса `detectMultiScale()` который принимает в качестве входных данных следующие параметры: `Image` – исследуемое изображение, `Vec` – переменная типа `vector` (для сохранения координат и размеров найденных объектов), `scaleFactor` – коэффициент масштабирования (указывающий, во сколько раз будет сжато изображение перед процедурой детектирования), `minNeighbors` – пороговое значение интенсивности для детектирования (определяет минимальную интенсивность обнаружения лица в заданной позиции для принятия положительного решения), `minSize` – минимальный размер детектируемого объекта (все объекты, имеющие размеры меньше данного, игнорируются), `maxSize` – максимальный размер детектируемого объекта, (все объекты, имеющие размеры больше данного, игнорируются).

В ходе проводимых исследований был обучен каскад, детектируемым объектом для которого является весь балансир целиком. Для его обучения было использовано 350 изображений данного балансира при различных условиях. Изменялись такие условия, как ракурс, степень освещенности, а также наличие шарика в кадре.

В итоге был получен результат, характеризующийся следующими недостатками: наличие ложноположительных срабатываний на предметы со схожими текстурными переходами, такие как потолочные гардины и т.п., а также наличие неприемлемой задержки, пропорциональной размеру изображения.

На рис. 1 виден конечный результат работы программы – балансир детектирован, который и выделен зеленым прямоугольником.

Для устранения выявленных недостатков было принято решение об использовании дополнительного каскада, целью которого является детектирование ограничителей хода шарика, находящихся на противоположных краях балансира. Данный каскад является вспомогательным и служит для отсеивания ложноположительных срабатываний. Поскольку данный каскад предполагается использовать в каче-

стве фильтра для объектов, детектированных с помощью основного каскада, то поиск данных объектов по всему изображению представляется абсолютно нерациональным.



Рис. 1. Результат распознавания балансира

Поэтому после детектирования с помощью основного каскада из изображения вырезаются относительно небольшие области вокруг краев объекта, на которых уже и производится детектирование ограничителей хода шарика. Введение данного фильтра практически не повлияло на время обработки. Стоит отметить, что для распознавания обоих ограничителей используется один и тот же каскад, эта особенность реализуется с помощью зеркального отображения изображения, полученного для детектирования одного из ограничителей.

На рис. 2 виден конечный результат работы усовершенствованной программы: детектированы балансир и оба ограничителя хода, балансир выделен зеленым прямоугольником, ограничители – оранжевыми прямоугольниками.

Проблема недостаточной скорости обработки также имеет решение. Принципом, положенным в основу данного решения, является допущение невозможности резкого перемещения балансира из одного местоположения в другое, т.е. если балансир на предыдущем изображении в видеопотоке имел некоторые координаты, то на следующем

кадре имеет смысл искать его вблизи этих координат. Таким образом, при первом обнаружении балансира в кадре происходит запоминание его координат, после чего на следующем кадре поиск проводится только в окрестностях этих координат. После обнаружения балансира в данной окрестности координаты обновляются. Если балансир не обнаружен в предполагаемой области, то производится поиск по всему изображению.



Рис. 2. Результат распознавания балансира усовершенствованным методом

В результате достигается требуемый уровень качества и скорости распознавания. В дальнейшем планируется создание альтернативных алгоритмов, например, использующих каскады, распознающие ограничители хода как первичный фильтр. Также алгоритм предполагается дополнить каскадом, распознающим крепление балансира к оси.

Библиографический список

1. Кавалеров М.В., Матушкин Н.Н. Мехатронная система «Шарик на балансира с веб-камерой» для исследований на стыке компьютерного зрения и планирования задач реального времени // Управление в технических, эргатических, организационных и сетевых системах: материалы конф. / под ред. С.Н. Васильева, И.А. Каляева, Д.А. Новикова, Г.Г. Себрякова; 9–11 октября 2012, г. Санкт-Петербург. – СПб., 2012. – С. 744–747.

2. Кавалеров М.В., Набиюллин А.Р., Патокин А.А. Возможность применения технологии Wi-Fi для организации удаленной обработки видеoinформации в системе компьютерного зрения мобильного робота // Управление в технических, эргатических, организационных и сетевых системах: материалы конф. / под ред. С.Н. Васильева, И.А. Каляева, Д.А. Новикова, Г.Г. Себрякова; 9–11 октября 2012, г. Санкт-Петербург. – СПб., 2012. – С. 748–751.

3. Кавалеров М.В., Набиюллин А.Р., Патокин А.А. Особенности реализации компьютерного зрения мобильного робота на основе удаленной обработки видеoinформации, передаваемой по каналу Wi-Fi // Прикладная фотоника. – 2012. – № 1–4. – С. 204–213.

4. Кавалеров М.В., Матушкин Н.Н. Получение оценок параметров выполнения запросов реального времени в условиях планирования с фиксированными приоритетами // Нейрокомпьютеры: разработка, применение. – 2014. – № 12. – С. 7–13.

5. Кавалеров М.В., Матушкин Н.Н. Повышение эффективности планирования с фиксированными приоритетами задач жесткого реального времени на основе применения R-алгоритма для формирования запросов этих задач // Нейрокомпьютеры: разработка, применение. – 2014. – № 12. – С. 14–19.

6. Viola P., Jones M. J. Robust real-time face detection // International journal of computer vision. – 2004. – Т. 57. – № 2. – С. 137–154.

7. Liao S., Law M. W. K., Chung A. Dominant local binary patterns for texture classification // Image Processing, IEEE Transactions on. – 2009. – Т. 18. – № 5. – С. 1107–1118.

МОДЕРНИЗАЦИЯ ФРЕЙМВОРКА НА ЯЗЫКЕ C# ДЛЯ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ВЕРХНЕГО УРОВНЯ

Магистрант Н.А. Софин

Пермский национальный исследовательский
политехнический университет

Разработана следующая версия фреймворка под названием ATFrameWork 2.0. Выполнен переход с менее производительной платформы Windows Forms на платформу Windows Presentation Foundation (WPF). Разработан ряд новых компонентов и модернизированы некоторые старые компоненты.

Ключевые слова: WPF, DirectX, программирование, C#, XAML, WCF, сокет, автоматизация разработки ПО.

UPGRADE OF FRAMEWORK BASED ON C# FOR DEVELOPING SOFTWARE OF UP LEVEL

Master Student N.A. Sofin

Perm National Research Polytechnic University

The next version framework is called ATFrameWork 2.0 is developed. The transition from a less productive platform Windows Forms to one Windows Presentation Foundation (WPF) is implemented. Set new components and some old one upgrade are developed.

Keywords: WPF, DirectX, programming, C#, XAML, WCF, sockets, automation of developing software.

Существуют два основных подхода к разработке программного обеспечения верхнего уровня систем управления:

- 1) применение SCADA-пакетов [1];
- 2) использование универсальных средств и сред программирования, в частности Microsoft Visual Studio или Qt framework (примеры их использования см. в работах [2, 3]).

Разрабатываемый фреймворк [4] (с названием ATFrameWork) реализуется на основе технологий Microsoft для повышения эффективности разработки программного обеспечения систем управления. Этот фреймворк ориентируется на нишу между SCADA-пакетами и универсальными средствами разработки, сочетая в себе гибкость универсальных средств с наличием специализированных библиотек компонентов и методов их объединения, присущих SCADA-пакетам.

Основные сведения о платформе WPF. Платформа Windows Presentation Foundation [5] по сравнению со своим предшественником, платформой Windows Forms [6], обладает рядом преимуществ:

1. Аппаратное ускорение. Всё рисование WPF выполняет через DirectX, что позволяет этой технологии пользоваться преимуществом современных видеокарт;

2. Независимость от разрешения экрана. Технология WPF настолько гибкая, что может автоматически выполнять масштабирование, приспособиваясь к глобальным настройкам монитора, в зависимости от системной настройки DPI;

3. Декларативный пользовательский интерфейс. Язык XAML позволяет строить окна без кода на языке C#, это позволяет программисту разрабатывать проект совместно с дизайнером, причём программист будет заниматься только логикой проекта, а дизайнер его внешним видом;

4. Стилизация различных частей программы, вплоть до самой мелкой детали элемента управления (ЭУ).

В связи с вышеуказанными преимуществами было решено перенести ATFrameWork и все проекты, ранее написанные с его использованием, на платформу WPF. Ниже перечислены проблемы, которыми обладала старая версия фреймворка, что, в свою очередь, послужило поводом для разработки новых компонентов и оптимизации старых:

1. Элемент управления для визуализации данных:

– слабая производительность при визуализации очень большого объёма данных. При количестве точек свыше 10 000 компонент с трудом отзывался на действия пользователя, и речь идёт лишь только об одном параметре, в то время как в любой системе их очень много;

– отсутствие пользовательской настройки внешнего вида. Все визуальные характеристики были прописаны в коде, и пользователь не мог настроить график под себя;

2. Библиотека последовательных интерфейсов стандарта RS 232 формировала неудобный в использовании список COM портов. Сложно было догадаться, к какому устройству относится, например, COM2 или COM3;

3. Поскольку большинство технологических процессов сопровождаются некоторым набором задач реального времени (ЗРВ), то необходим таймер высокой точности;

4. Увеличение числа хранимых параметров и нерациональная система сохранения и восстановления состояния приложения приводили к частым критическим ошибкам программы;

5. В связи с изучением сетевых технологий, таких как сокет, входящие в состав технологии межпроцессного взаимодействия (WCF), необходима библиотека, автоматизирующая передачу данных из одного процесса в другой;

6. Необходима утилита для постоянного мониторинга выполнения ЗРВ.

Обновлённый компонент для визуализации данных. Новая версия компонента лишена перечисленных ранее недостатков. Даже при больших объёмах данных, свыше 40 000, компонент показывает хорошую производительность. Однако если большое число данных будет на виду у пользователя, наблюдается её снижение. Проведя исследования данной проблемы, был сделан вывод, что все современные компоненты обладают такой проблемой, за исключением компонента в Builder C++ 6.0, но он уже является устаревшим и с современными средствами и принципами разработки не совместим. В связи с этим были разработаны алгоритмы обработки информации. Большое число данных на графике может появиться в случае долгой работы системы либо из-за большой частоты поступления. Стоит отметить, что в обоих случаях не каждая точка несёт в себе полезную информацию для пользователя, особенно на большой частоте приёма избыточности не избежать. Ниже представлен список алгоритмов обработки информации, используемых данным компонентом:

1. Усреднение по частоте. За указанный период времени накапливается объём данных, а на график наносится среднее значение. В результате, например, 100 точек превращаются в одну;

2. Построение по частоте. Через определённые промежутки времени из потока данных берётся значение и наносится на график;

3. Если же всё-таки нужно наносить на график абсолютно все значения, то используются следующие алгоритмы:

– обновление графика через определённое количество точек. Указанный объём данных накапливается и затем добавляется на график. Это снижает частоту обращения к графику и позволяет программе успеть выполнить остальные задачи;

– сравнение на равенство с предыдущей точкой. В случае если текущая точка равна предыдущей, то нет смысла добавлять её на график;

– сравнение разности двух точек и предельного значения. В случае если разность текущей и предыдущей точки не превышает предельного значения, то нет смысла добавлять её на график;

4. С течением времени, даже если применять указанные алгоритмы, количество данных на экране всё равно станет большим, поэтому была разработана система управления масштабированием, которая сама подстраивается под текущий объём данных.

Также стоит отметить, что график обладает полностью настраиваемым интерфейсом, панель настроек которого изображена на рис. 4. Внешний вид нового компонента представлен на рис. 1.



Рис. 1. Новый внешний вид ЭУ для построения графиков

Обновлённая библиотека последовательных интерфейсов. На рис. 2, *а* представлен список COM-портов, который формировала старая версия библиотеки, а на рис. 2, *б* показан новый способ формирования. Подобные сведения (даже в гораздо большем количестве) хранятся в реестре Windows.

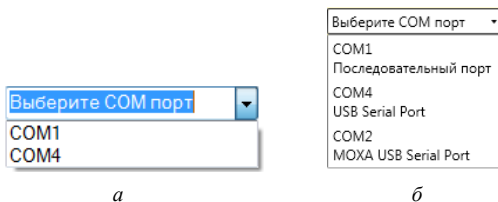


Рис. 2. Список COM-портов: старый способ (*а*) и новый способ (*б*)

Новый компонент «Точный таймер». Задачи реального времени решались путём создания экземпляра таймера на каждую задачу, что, в свою очередь, приводило к смещению графика их выполнения. Стоит отметить, что стандартный таймер платформы .NET Framework не обладает высокой точностью и не пригоден для решения ЗРВ. Также большое количество экземпляров таймера является нерациональным подходом к выполнению ЗРВ и нагружало систему. В связи с этим был разработан новый компонент, наследованный от Multimedia-таймера [7] (таймер, более близкий к Windows API) с возможностью выполнять несколько задач одновременно. Принцип работы данного компонента изображён на рис. 3 в виде структурной схемы.

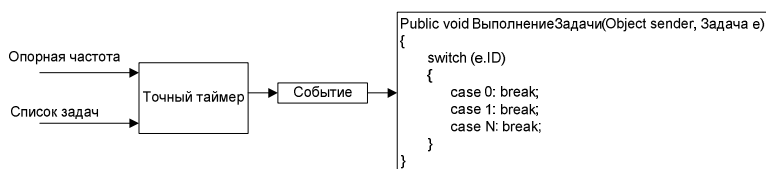


Рис. 3. Структурная схема точного таймера

Принцип действия основан на отсчёте необходимого числа циклов, причём длительность одного цикла задаётся опорной частотой. Разработанный таймер саму задачу не выполняет внутри своего класса, он воспринимает её в виде некой модели, описанной в виде класса на языке C# с перечисленными ниже свойствами, и при определённом условии генерирует событие выполнения:

- ID-идентификатор задачи;
- частота – это частота выполнения в миллисекундах, должна быть кратна опорной частоте таймера и быть большей либо равной ей;
- необходимое число циклов – частота, переведённая в количество циклов таймера;
- счётчик циклов – если данный счётчик станет равным необходимому числу циклов, значит, будет сгенерировано событие выполнения задачи;
- описание задачи, состояние задачи, время старта, время окончания;
- разность – это время выполнения задачи, т.е. время окончания минус время старта.

Система сохранения и восстановления состояния приложения. В объектно-ориентированных программах настройки програм-

мы можно поделить на две категории: общие и индивидуальные. Общие настройки применяются ко всем экземплярам класса, а индивидуальные – к конкретному экземпляру, причем не важно, чем именно является этот класс – элементом управления или невизуальным компонентом, т.е. обычным классом. В соответствии с данной системой программу можно условно разделить на объекты и панели настроек этих объектов. К примеру, описанный выше компонент для построения графиков содержит в себе много объектов (график одного параметра), которые при запуске или при подтверждении новой конфигурации восстанавливают свои параметры, созданные с помощью панели настроек, показанной на рис. 4. Некоторые объекты также могут заниматься созданием конфигурации, но только в случае индивидуальных параметров (например, СОМ-порта канала), присущих именно этому объекту.

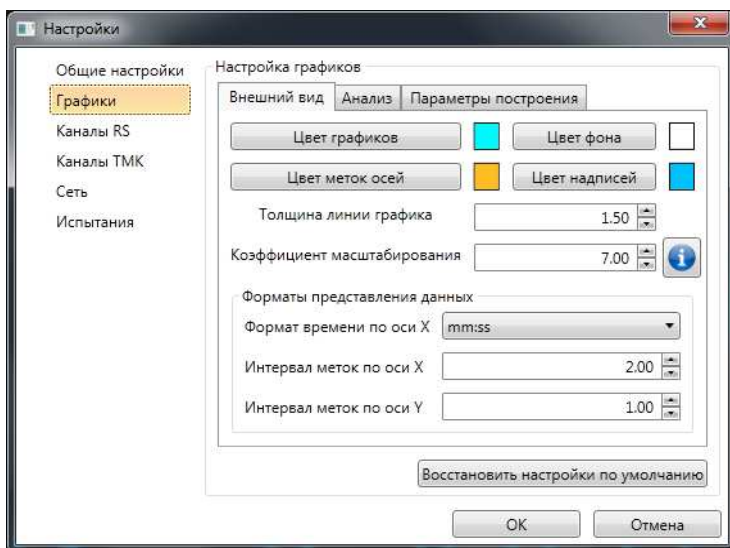


Рис. 4. Пример панели настроек

Для того чтобы объект мог пользоваться возможностями данной системы, он должен реализовывать специальную конструкцию языка C# под названием Interface IManageState. Такие конструкции содержат в себе только сигнатуры методов, а их реализацией занимается класс, который реализует данный интерфейс. В отличие от старой версии

этой системы, где все параметры приходилось хранить в текстовом файле собственной структуры, здесь используется более удобный подход к хранению конфигурационной информации, т.е. XML-файлы.

Утилита «Мониторинг выполнения ЗРВ». Данная программа является средством наблюдения за работой таймера ЗРВ. Программа работает с использованием межпроцессного взаимодействия. Таймер с помощью специального свойства знает, когда запускается эта утилита и начинает трансляцию потока данных по IP: 127.0.0.1 в порт 49008. Когда утилита запустится, она подключается по упомянутым параметрам и начинает приём данных. Здесь межпроцессное взаимодействие было сделано для того, чтобы утилита не мешала своим существованием работе основной программы, так как её цель – наблюдать, а не приносить дополнительные задержки в работу основной программы. Внешний вид программы представлен на рис. 5.

ID Задачи	Частота, мс	В циклах таймера	Время старта	Время окончания	Время выполнения, мс	Состояние	Описание
0	60000	600	2041:10:718	2041:10:720	2.0001	Активна	Построения графика по определённой частоте
1	0	0	00:00:00:000	00:00:00:000	0	Неактивна	Усреднения данных для графика по определённой частоте
2	1000	10	00:00:00:000	00:00:00:000	0	Неактивна	Задача записи принятых данных в файлы
3	1000	10	2041:58:718	2041:58:721	3.0002	Активна	Задача имитации приёма данных по каналу RS

Рис. 5. Утилита «Монитор задач реального времени»

С помощью данной утилиты можно наблюдать за ходом выполнения задач. К примеру, если время выполнения будет больше частоты, то значит в программе что-то не так и необходимо заняться оптимизацией .NET приложения, однако это уже повод для совершенно другой статьи.

Таким образом, благодаря очередному этапу модернизации повысилась производительность приложений, основанных на данном фреймворке, а также скорость и качество разработки.

Библиографический список

1. Кавалеров М.В. К вопросу о термине «SCADA-система» // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2011. – № 5. – С. 205–209.

2. Софин Н.А., Кавалеров М.В. Модернизация программного обеспечения лабораторного стенда «Модель вентиляционной системы» // Инновационные технологии: теория, инструменты, практика. – 2014. – Т. 2. – С. 459–465.

3. Кавалеров М.В., Матушкин Н.Н. Мехатронная система «Шарик на балансирах с веб-камерой» для исследований на стыке компьютерного зрения и планирования задач реального времени // Управление в технических, эргатических, организационных и сетевых системах: материалы конф. / под ред. С.Н. Васильева, И.А. Каляева, Д.А. Новикова, Г.Г. Себрякова; 9–11 октября 2012, г. Санкт-Петербург. – СПб., 2012. – С. 744–747.

4. Софин Н.А. Модернизация аппаратно-программного обеспечения системы управления перевёрнутым маятником на тележке // Автоматизированные системы управления и информационные технологии: материалы всерос. науч.-техн. конф.; 15 мая 2015, г. Пермь. – Пермь: Изд-во Перм. нац. исслед. политехн. ун-та, 2015. – С. 223–228.

5. Мак-Дональд Мэтью. WPF. Windows Presentation Foundation в .NET 4.5 с примерами на C# 5.0. – М.; СПб.; Киев, 2013. – 1015 с.

6. Бен Ватсон C# 4.0. На примерах. – СПб., 2011. – 590 с.

7. Multimedia таймер [Электронный ресурс]. – URL: [https://msdn.microsoft.com/en-us/library/windows/desktop/dd743609\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/dd743609(v=vs.85).aspx) (дата обращения: 16.04.2015).

МОДЕРНИЗАЦИЯ АППАРАТНО-ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ СИСТЕМЫ УПРАВЛЕНИЯ ПЕРЕВЁРНУТЫМ МАЯТНИКОМ НА ТЕЛЕЖКЕ С ПРИМЕНЕНИЕМ БЕСПРОВОДНЫХ ТЕХНОЛОГИЙ

Магистрант Н.А. Софин

Пермский национальный исследовательский
политехнический университет

Выполнен очередной этап модернизации аппаратно-программного обеспечения лабораторного стенда «Перевернутый маятник на тележке». Теперь реализована мобильная система управления на базе операционной системы Android смартфона с применением Bluetooth. Реализован ПД-регулятор для поддержания равновесия.

Ключевые слова: перевернутый маятник, Arduino, программирование, Java, GUI для Android, Bluetooth.

SOFTWARE AND HARWARE UPGRADE OF INVERTED PEN- DULUM SYSTEM CONTROL BY USING WIRELESS TECHNOLOGIES

Master Student N.A. Sofin

Perm National Research Polytechnic University

The next stage of hardware and software upgrade of the laboratory stand "Inverted pendulum" is implemented. Mobile control system based on smartphone by using Bluetooth is developed. PD controller is implemented for control of balance.

Keywords: inverted pendulum, Arduino, programming, Java, GUI for Android, Bluetooth.

Лабораторный стенд «Перевернутый маятник» представляет собой один из типовых объектов управления и позволяет проводить исследования на стыке теории управления и теории планирования вычислительных процессов. Так же, как и в случае объекта «Шарик на балансире» [1], система «Перевернутый маятник» позволяет исследовать проблему интеграции управления и вычислений, что особенно актуально в случае сложных алгоритмов управления, например, основанных на машинном обучении. В частности, на основе этой системы можно проводить анализ временных характеристик вычислительных задач и выполнять их эффективное планирование [2, 3].

Основываясь на работе, проделанной в статье [4], был выполнен очередной этап модернизации аппаратно-программного обеспечения лабораторного стенда «Перевернутый маятник». В этой статье были отмечены следующие направления развития:

- разработка и тестирование более сложных алгоритмов управления, таких как ПД-регулятор, нечёткое управление и алгоритмы самообучения, например, на основе Q-Learning;
- установка аккумуляторных батарей и модуля беспроводной связи. Это необходимо сделать, так как наличие проводов мешает машинке свободно двигаться.

Независимо от алгоритма управления старая версия машинки, изображенная на рис. 1, стабилизировала своё положение с большим трудом и легко выводилась из состояния равновесия.

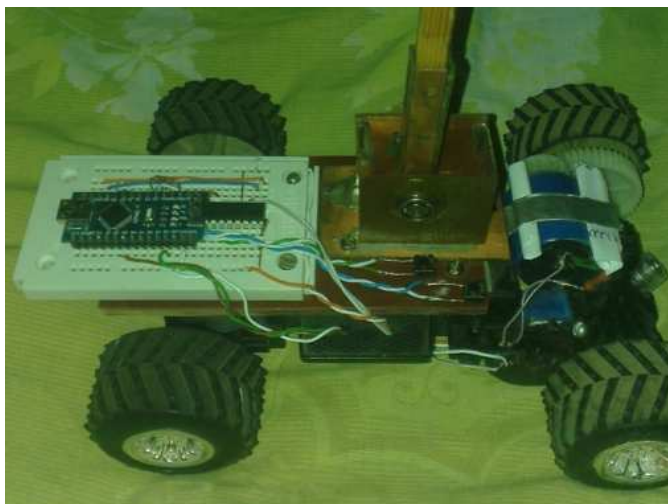


Рис. 1. Внешний вид старой версии машинки

Такое поведение объясняется рядом конструктивных причин:

1. Ведущая пара колёс обладает некачественным сцеплением с двигателем, в связи с чем иногда наблюдалось проскальзывание;
2. Колёса выполнены из твёрдой пластмассы. С таким видом материала был маленький коэффициент трения с поверхностью, также наблюдалось проскальзывание;
3. Мало места для размещения электроники и аккумуляторных батарей.

Вышеперечисленные причины послужили поводом для смены платформы маятника, новый внешний вид которой изображён на рис. 2.

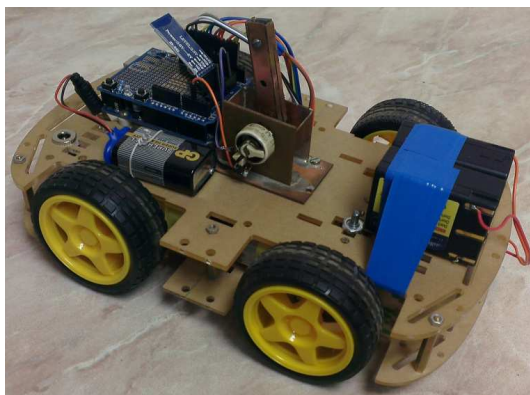


Рис. 2. Новый внешний вид лабораторного стенда «Перевернутый маятник»

Данная модель машинки в отличие от старой отличается более надёжной конструкцией. Вдобавок она обладает 4-приводной системой, т.е. на каждое колесо установлен свой двигатель с редуктором. Большое количество пространства на верхнем этаже машинки и внутри неё дают возможность поместить большое количество различных электронных модулей, а также два аккумулятора для питания силовой части и батарею для питания контроллера. Все 4 двигателя управляются централизованной подачей ШИМ-сигнала. В связи с увеличением количества двигателей электронная схема машинки претерпела следующие изменения, как показано на рис. 3.

На электронной схеме видно, что силовая часть обладает двумя источниками питания. Первый источник – это аккумуляторные батареи в сумме на 16,8 В. Он необходим для отладки стенда в лабораторных условиях, а также для работы при удалении от питающей сети. Второй источник питания представляет из себя малогабаритный трансформаторный блок питания на 15 В, который используется в качестве альтернативного источника. На схеме также видно, что напряжение с блока питания в схему подаётся не напрямую, а через регулируемый стабилизатор напряжения, который был необходим для того, чтобы снизить нагрузку на микросхемах L293D. В дополнение к этому можно сказать, что регулируемый стабилизатор напря-

жения даёт возможность для проведения целого ряда экспериментов по настройке регуляторов, так как при изменении напряжения питания математическая модель маятника становится другой, а значит, старые подобранные коэффициенты уже не будут подходить.

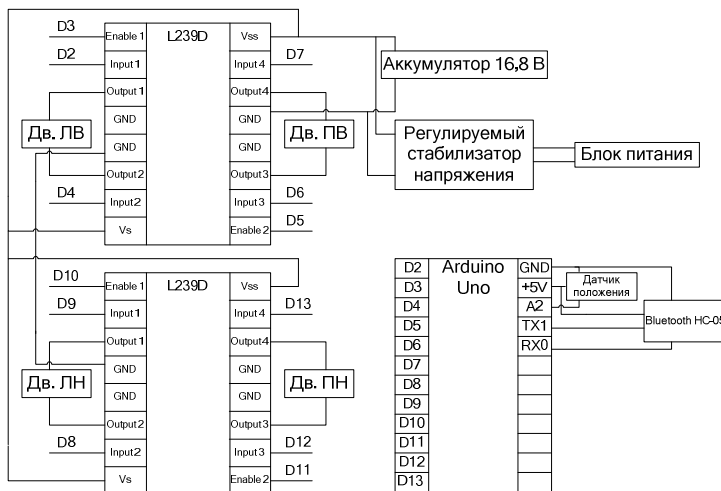
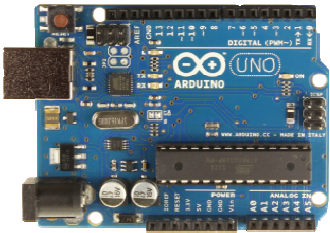


Рис. 3. Электрическая схема лабораторного стенда «Перевернутый маятник»

Модель контроллера (в предыдущей версии Arduino Nano) была заменена на Arduino Uno, подробная работа с которым описана в работе [5], по следующим причинам. Данная модель имеет отверстия для закрепления контроллера на корпус машинки винтами. Контроллер Arduino Uno поддерживает установку плат расширения (shield), на которой можно производить работу с паяльником. Контроллер имеет отдельный разъём для программирования и отдельный разъём для питания. Внешний вид нового контроллера представлен на рис. 4, а. Поскольку машинка теперь снабжена аккумуляторами, то нет необходимости использовать проводные источники питания (за исключением вышеговорённых ситуаций). Единственный провод, который осуществлял связь с системой управления, расположенной на компьютере, был заменён на беспроводную технологию связи Bluetooth, а сама система управления была полностью изменена и теперь представляет из себя приложение на базе операционной системы Android [6]. Bluetooth-модуль HC-05 был установлен на плату расширения контроллера. Внешний вид Bluetooth-модуля представлен на рис. 4, б.



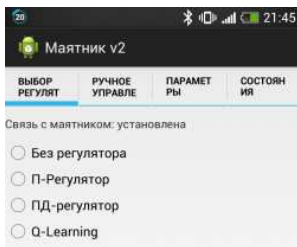
a



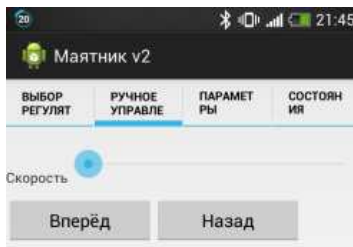
б

Рис. 4. Внешний вид Arduino Uno-модуля (*a*); внешний вид Bluetooth (*б*)

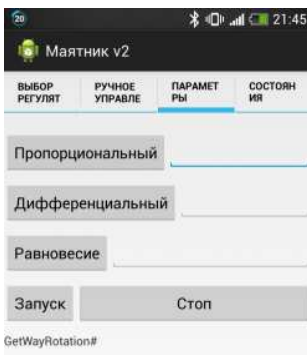
Приложение системы управления, внешний вид которой был разработан в среде проектирования Eclipse [7], представлен на рис. 5.



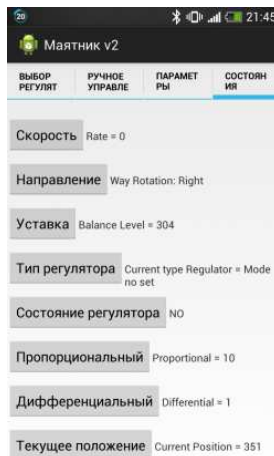
a



б



в



г

Рис. 5. Система управления: экран выбора регуляторов (*a*); экран ручного управления (*б*); экран настроечных коэффициентов (*в*); экран отладки программы контроллера (*г*)

Размеры экрана телефона очень далеки от размеров экрана компьютера, а значит, принципы проектирования пользовательского интерфейса мобильного приложения будут сильно отличаться. Все элементы управления пришлось сгруппировать по специальным категориям с помощью контейнерного элемента управления TabHost. Вкладки этого элемента управления могут разместить внутри себя большое число дочерних элементов управления, что, в свою очередь, позволяет проектировать сложные интерфейсы. В итоге получилось приложение, состоящее из четырёх экранов:

1. Первый экран, расположенный на рис. 5, *а*, отвечает за выбор алгоритма управления;

2. Второй экран, расположенный на рис. 5, *б*, необходим для ручного управления машинкой. На нём расположены кнопки смены направления движения и регулятор скорости;

3. На третьем, показанном на рис. 5, *в* экране, расположены настроечные коэффициенты и кнопки управления системой стабилизации маятника;

4. Последний экран, расположенный на рис. 5, *г*, необходим для отладки программы контроллера.

За каждой кнопкой закреплена соответствующая команда, которая через Bluetooth смартфона отправляется в Bluetooth машинки и далее обрабатывается программой контроллера.

На текущем этапе разработки маятник достигает положения равновесия за счёт работы ПД-регулятора. В отличие от П-регулятора, который вырабатывал управляющее воздействие в зависимости от значения величины ошибки (разница текущего угла и уставки), ПД-регулятор учитывает скорость её изменения. Программная реализация ПД-регулятора представлена ниже.

```
void PD_Regulator(){
    long TimeCurPos = millis();
    int CurPos = analogRead(Pin_Sensor);
    int Balance_Level =
atoi(Desired_Balance);
    if (CurPos > LeftLimit && CurPos <
RightLimit){
        if (CurPos > Balance_Level)
            FuncForPDController(TimeCurPos, CurPos
- Balance_Level, 1);
```

```

        else
            FuncForPDController(TimeCurPos, Balance_Level - CurPos, 0);
        }
        else
            TurnMotors(0, 0);
        }
        void FuncForPDController(long TimeCurPos, double CurError, int WayMotion){
            double P_x_Error = CurError *
atof(Kp);
            double DiffPart = (LastError-
CurError)*atof(Kd)/(TimeCurPos-TimeLastPos);
            double PMW = P_x_Error + DiffPart;
            TurnMotors(Constraint(PMW),
WayMotion);
            LastError = CurError;
            TimeLastPosition = TimeCurPos;
        }

```

Из программного кода видно, что регулятор работает только в определённом диапазоне текущего положения `LeftLimit` и `RightLimit`. Объясняется это тем, что за пределами этого диапазона маятник уже не сможет достичь положения равновесия, и чтобы избежать физических повреждений (так как скорость машинки в таком случае достигает максимума), машинка полностью прекращает своё движение.

Ниже перечислены итоги проделанной работы и достигнутые результаты:

1. Изменение конструкции лабораторного стенда привело к значительно лучшей стабилизации маятника даже с использованием простых классических алгоритмов управления;

2. Также благодаря новой конструкции появилась возможность установки аккумуляторных батарей, благодаря которым стало возможно избавиться от провода, питающего силовую часть, и от провода, питающего контроллер, а это, в свою очередь, повлекло замену компьютерной системы правления на мобильную.

Дальнейшие пути развития лабораторного стенда будут направлены на использование алгоритмов машинного обучения (Q-Learning) и нейронечёткого управления.

Библиографический список

1. Кавалеров М.В., Матушкин Н.Н. Мехатронная система «Шарик на балансирах с веб-камерой» для исследований на стыке компьютерного зрения и планирования задач реального времени // Управление в технических, эргатических, организационных и сетевых системах: материалы конф. / под ред. С.Н. Васильева, И.А. Каляева, Д.А. Новикова, Г.Г. Себрякова; 9–11 октября 2012, г. Санкт-Петербург. – СПб., 2012. – С. 744–747.

2. Кавалеров М.В., Матушкин Н.Н. Получение оценок параметров выполнения запросов реального времени в условиях планирования с фиксированными приоритетами // Нейрокомпьютеры: разработка, применение. – 2014. – № 12. – С. 7–13.

3. Кавалеров М.В., Матушкин Н.Н. Повышение эффективности планирования с фиксированными приоритетами задач жесткого реального времени на основе применения R-алгоритма для формирования запросов этих задач // Нейрокомпьютеры: разработка, применение. – 2014. – № 12. – С. 14–19.

4. Софин Н.А. Модернизация аппаратно-программного обеспечения системы управления перевёрнутым маятником на тележке // Автоматизированные системы управления и информационные технологии: материалы всерос. науч.-техн. конф.; 15 мая 2015, г. Пермь. – Пермь: Изд-во Перм. нац. исслед. политехн. ун-та, 2015. – С. 223–228.

5. Блум Д. Изучаем Arduino. Инструменты и методы технического волшебства. – СПб., 2015. – 336 с.

6. Голощاپов А. Google Android. Создание приложений для смартфонов и планшетных ПК. – СПб., 2014. – 923 с.

7. Ссылка на скачивание среды разработки Eclipse [Электронный ресурс]. – URL: <http://www.eclipse.org/downloads/> (дата обращения: 30.09.2015).

ОСОБЕННОСТИ ПРИМЕНЕНИЯ OPENSCADA ДЛЯ РЕАЛИЗАЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ СИСТЕМ УПРАВЛЕНИЯ

Магистрант М.А. Чудинов

Пермский национальный исследовательский
политехнический университет

Рассмотрены основные особенности SCADA-пакетов, приведено краткое описание системы OpenSCADA. Для представления особенностей работы с OpenSCADA проведена разработка алгоритмов управления на примере модели резервуара с водой и проведена проверка функционирования полученной системы.

Ключевые слова: SCADA-система, OpenSCADA, мнемосхема, пользовательский графический интерфейс.

FEATURES OF APPLYING OPEN SOURCE SCADA SYSTEM OPENSCADA FOR IMPLEMENTATION CONTROL SYSTEMS SOFTWARE

Master Student M.A. Chudinov

Perm National Research Polytechnic University

The article gives some information about main features of SCADA-packages and brief description of the open source SCADA-system called OpenSCADA. The article describes the control algorithms elaboration and testing of designed system. As an example, the model of water container is used.

Keywords: SCADA-system, OpenSCADA, mimic diagram, graphical user interface.

SCADA-пакеты (или SCADA-системы) являются одним из инструментальных средств разработки программного обеспечения при проектировании систем автоматизации и управления [1].

Вариант разработки программного обеспечения систем управления на основе SCADA-пакетов является альтернативой по отношению к разработке на основе универсальных средств и сред программирования. Например, в работе [2] показан пример разработки программного обеспечения системы управления на основе Microsoft Visual Studio и языка C#. Другой пример – это использование фреймворка Qt в качестве универсального инструментального средства для разработки программного обеспечения для системы взаимодействия с мобильным роботом [3].

К основным достоинствам SCADA-пакетов по сравнению с универсальными средствами разработки следует отнести: сокращение времени на разработку большой и распределенной системы за счет имеющихся готовых компонентов и методов их настройки, а также удобство и простоту последующих локальных модификаций работающей системы за счет встроенных средств редактирования и конфигурирования.

Одной из популярных SCADA-систем является OpenSCADA, которая представляет собой открытую SCADA-систему, построенную по принципам модульности, многоплатформенности и масштабируемости. Система OpenSCADA предназначена для сбора, архивирования, визуализации информации, выдачи управляющих воздействий, а также других родственных операций, характерных для полнофункциональной SCADA-системы [4].

Система OpenSCADA предназначена для выполнения как обычных функций SCADA-систем, так и для использования в смежных областях информационных технологий. Основная отличительная особенность данного SCADA-пакета состоит в том, что у него открытый исходный код, он бесплатный, имеет хороший функционал и работает под операционной системой Linux, что является довольно редким сочетанием свойств. Наличие открытого исходного кода крайне важно еще и потому, что это упрощает получение оценок времени выполнения задач реального времени [5] и позволяет применять это для решения проблем планирования в управляющих системах реального времени [6].

Учитывая ограниченный объем статьи, для представления особенностей работы с OpenSCADA рассмотрим разработку на примере модели резервуара с водой. При этом предполагаются следующие требования:

- программа должна содержать графический интерфейс пользователя;
- должна быть разработана модель резервуара с водой с возможностью изменения уровня;
- реализована возможность изменения значения притока и расхода во время работы с моделью;
- реализовано отображение оповещений об аварийной ситуации при достижении уровнем верхней или нижней границы модели резервуара.

В новом проекте создаем виджет (визуальный элемент) типа `wdg_Box` (рис. 1) для задания страницы, нажимая соответствующую кнопку на панели инструментов.

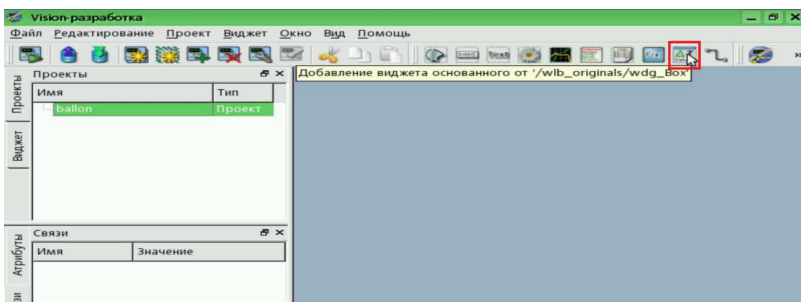


Рис. 1. Создание виджета (визуального элемента) типа `wdg_Box`

Размещение элементов интерфейса на странице. Графические примитивы могут быть нарисованы с помощью мыши или заданы с помощью списка графических примитивов («Список элементов»).

По аналогии с размещением модели резервуара с водой создаем остальные графические элементы, чтобы получить пользовательский интерфейс, подобный изображенному на рис. 2.

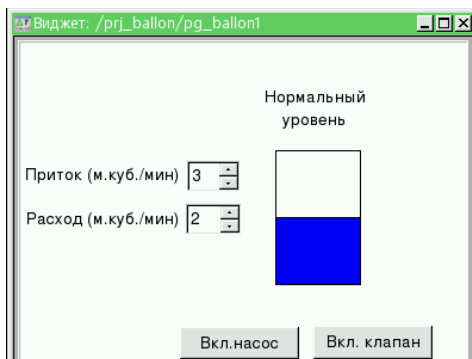


Рис. 2. Пример пользовательского интерфейса

Значения притока и расхода устанавливаются при помощи виджета элемента формы (`wdg_FormEl`). Кнопки включения насоса и клапана также являются одной из форм виджета `wdg_FormEl`. Во вкладке «Атрибуты» установим «тип элемента» – «Кнопка». А также

установим атрибут «Имя», который будет отображаться на кнопке. Для установки текстовой надписи служит виджет `wdg_Text`.

На следующем этапе займемся созданием источника сигнала и связыванием его с графическим интерфейсом.

Создание шаблона контроллера. Переходим в окно системного конфигуратора и следуем к ветке «Рабочая станция → Сбор данных». После этого нажимаем кнопку «Добавить элемент» на панели инструментов и создаем новую библиотеку шаблонов (рис. 3).

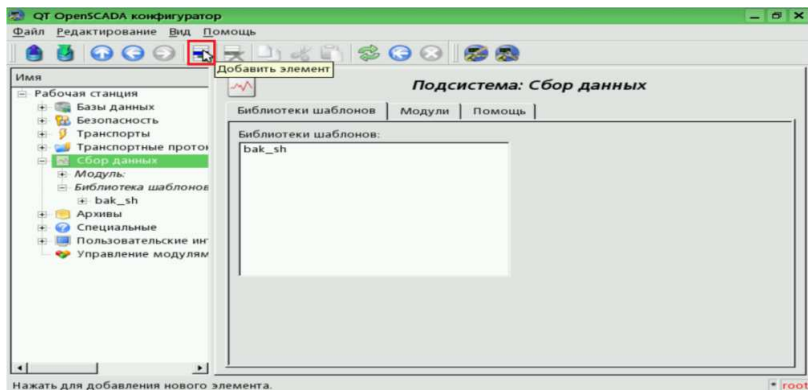


Рис. 3. Создание новой библиотеки шаблонов

В окне редактирования шаблона контроллера выбираем язык программирования (JavaLikeCalc → JavaScript) и размещаем в окне редактора разработанный код программы:

```
if (btn_nasos==true && btn_klapan==false &&
level_1<avar_level_verhn) {
    level_1=level_1+(nasos_prit);
}
if (btn_nasos==false && btn_klapan==true &&
level_1>avar_level_niz) {
    level_1=level_1-(klapan_rash);
}
if (btn_nasos==true && btn_klapan==true &&
level_1>avar_level_niz) {
    level_1=level_1-(klapan_rash-nasos_prit);}
```

Создание контроллера. Следуем в ветку «Модуль → Логический уровень», где создаем новый элемент, а именно контроллер, и связываем его с созданным ранее шаблоном (рис. 4).

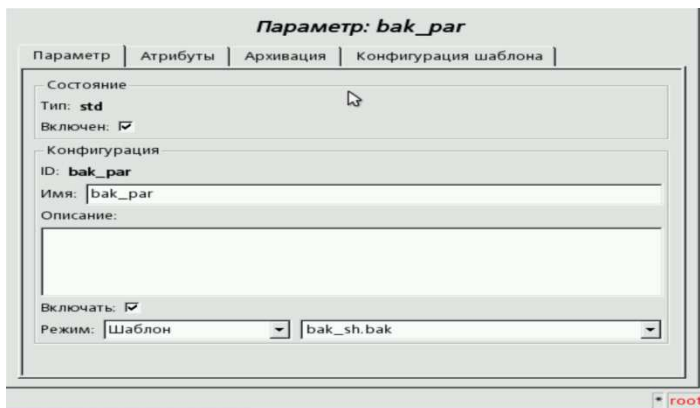


Рис. 4. Параметры логического контроллера

Связывание параметров сигнала с элементами графического интерфейса. Ранее были созданы источник сигнала и интерфейс пользователя, теперь надо связать их так, чтобы изменялся уровень в модели резервуара и можно было менять значения уровня с помощью элементов интерфейса. Вернемся к графическому редактору и откроем окно свойств основной страницы, перейдем к вкладке «Обработка». На данном этапе устанавливаются связи между созданными переменными и объектами пользовательского интерфейса на вкладке «Связи».

Индикация аварии. Эту программу необходимо поместить в окно программы на вкладке «Обработка» основной страницы.

Приведённая ниже программа отображает необходимые оповещения:

```
if (el_bak_level<=el_bak_avarv) {  
  if (state==1) {  
    state=0;  
    Text_bak_backColor="#aaff00";  
    Text_bak_text="Вак переполнен";  
  } else {  
    state=1;
```

```

Text_bak_backColor="#FF0000";
}}
else {
if
(el_bak_level>=el_bak_avar) {Text_bak_backColor="#
ff0000";
Text_bak_text="Бак пуст";}
else {Text_bak_backColor="#aaff00";
Text_bak_text="Нормальный уровень";}
}

```

Проверка функционирования полученного варианта системы. В качестве примера на рис. 5 представлено оповещение о переполнении бака, активируемое при достижении уровнем верхней границы модели резервуара.



Рис. 5. Аварийное оповещение модели «Бак переполнен»

При помощи средств визуализации и управления возможна разработка более сложной модели – мнемосхемы с элементами анимации. На рис. 6 представлена анимация бегущей воды, активированная включением насоса и открытием клапана.

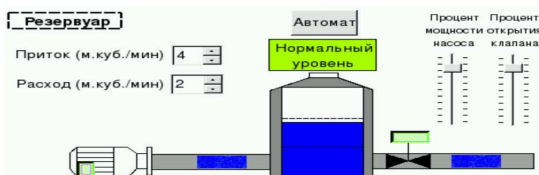


Рис. 6. Мнемосхема с анимацией бегущей воды

Заключение. В статье были рассмотрены некоторые особенности работы со SCADA-пакетами OpenSCADA, а также показана разработка алгоритмов управления на примере модели резервуара

с водой и проведена проверка функционирования полученной системы. Непосредственные вычисления обеспечиваются программным модулем «контроллер» и связыванием его с функцией модуля.

Учитывая редкое сочетание свойств (бесплатный открытый исходный код, работа в ОС Linux), в итоге можно заключить, что OpenSCADA является достаточно удобным SCADA-пакетом, с помощью которого можно реализовывать верхний уровень подобных систем автоматизации, т.е. реализовывать алгоритмы управления, разрабатывать мнемосхемы и элементы анимации.

Библиографический список

1. Кавалеров М.В. К вопросу о термине «SCADA-система» // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2011. – № 5. – С. 205–209.

2. Софин Н.А., Кавалеров М.В. Модернизация программного обеспечения лабораторного стенда «Модель вентиляционной системы» // Инновационные технологии: теория, инструменты, практика: Материалы VI Междунар. интернет-конф. молодых ученых, аспирантов, студентов. – Пермь: Изд-во Перм. нац. исслед. политехн. ун-та, 2014. – С. 459–465.

3. Кавалеров М.В., Набиюллин А.Р., Патокин А.А. Особенности реализации компьютерного зрения мобильного робота на основе удаленной обработки видеоинформации, передаваемой по каналу Wi-Fi // Прикладная фотоника. – 2012. – № 1–4. – С. 204–213.

4. Савоченко Р.А. О проекте OpenSCADA [Электронный ресурс]. – URL: <http://oscada.org/ru/glavnaja/o-proekte/> (дата обращения: 02.11.2015).

5. Кавалеров М.В., Матушкин Н.Н. Получение оценок параметров выполнения запросов реального времени в условиях планирования с фиксированными приоритетами // Нейрокомпьютеры: разработка, применение. – 2014. – № 12. – С. 7–13.

6. Кавалеров М.В., Матушкин Н.Н. Возможность существенного повышения эффективности планирования на основе непосредственного применения линейных интервальных ограничений реального времени // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2011. – № 5. – С. 4–13.

РЕАЛИЗАЦИЯ ВЗАИМОДЕЙСТВИЯ OPENSCADA С МИКРОКОНТРОЛЛЕРОМ ARDUINO

Магистрант М.А. Чудинов

Пермский национальный исследовательский
политехнический университет

В статье рассмотрены средства и способы взаимодействия OpenSCADA с микроконтроллером семейства Arduino, в частности, более подробно исследована возможность реализации взаимодействия на основе протокола DCON. Показаны принципы разработки программы для микроконтроллера Arduino, которая поддерживает систему команд для организации связи на основе DCON.

Ключевые слова: SCADA-система, OpenSCADA, мнемосхема, Arduino IDE, DCON, микроконтроллер, пользовательский графический интерфейс.

REALIZING OF OPEN SOURCE SCADA SYSTEM OPENSCADA AND MICROCONTROLLER ARDUINO INTERACTION

Master Student M.A. Chudinov

Perm National Research Polytechnic University

The article is about tools and methods of interaction between open source SCADA-system called OpenSCADA and microcontroller Arduino, particularly, more details are given about DCON-based interaction. The article describes the basics of program development for microcontroller Arduino with respect to command system of DCON-based communication.

Keywords: SCADA-system, OpenSCADA, mimic diagram, Arduino IDE, DCON, microcontroller, graphical user interface.

Современные SCADA-системы [1] не ограничивают выбора аппаратуры нижнего уровня (контроллеров), так как предоставляют большой набор драйверов или серверов ввода/вывода и имеют хорошо развитые средства создания собственных программных модулей или драйверов новых устройств нижнего уровня [2].

В качестве примера SCADA-пакета рассматривается OpenSCADA. Этот пакет имеет открытый исходный код, он бесплатный, имеет хороший функционал и работает под операционной системой Linux. Наличие открытого исходного кода крайне важно еще и потому, что это упрощает получение оценок времени выполнения задач реального времени [3] и позволяет применять это для решения проблем планирования в управляющих системах реального времени [4].

Основной задачей данной работы является обеспечение взаимодействия OpenSCADA с микроконтроллером Arduino, который является популярным техническим средством в составе систем управления (примеры таких систем представлены в работах [5, 6]). Рассмотрим возможные способы реализации такого взаимодействия. Одним из таких способов является предоставление данных системам верхнего уровня при помощи модуля транспортного протокола ModBus.

Синтаксис команд протокола позволяет адресовать 247 устройств на одной линии связи стандарта RS-485 (реже RS-422 или RS-232). В случае с режимом TCP адресация исключена из протокола, поскольку выполняется на уровне TCP/IP стека [7].

В данной статье предлагается альтернативный вариант реализации взаимодействия (на основе протокола DCON), который также поддерживается системой OpenSCADA.

Протокол DCON предполагает одно ведущее (запрашивающее) устройство в линии (master), которое может передавать команды одному или нескольким ведомым устройствам (slave), обращаясь к ним по уникальному в линии адресу. Инициатива проведения обмена всегда исходит от ведущего устройства. Ведомые устройства прослушивают линию связи. Мастер подаёт запрос (посылка, последовательность байт) в линию и переходит в состояние прослушивания линии связи. Ведомое устройство отвечает на запрос, пришедший в его адрес [8].

Команды протокола DCON делятся на 4 типа:

- команды модулей аналогового ввода;
- команды модулей аналогового вывода;
- команды дискретного ввода-вывода;
- команды счетчиков/таймеров [9].

В данном случае OpenSCADA всегда будет ведущим устройством в линии, т.е. будет опрашивать микроконтроллер Arduino, который, в свою очередь, будет отвечать на принятую команду в нужном формате.

На начальном этапе необходимо реализовать взаимодействие с микроконтроллером Arduino без участия OpenSCADA.

Реализация программного взаимодействия осуществляется с помощью микроконтроллера Arduino Nano в ОС Ubuntu. Написание программ, их компиляция, а также загрузка программы в микроконтроллер осуществляются в специальной среде разработки Arduino

IDE. Страница редактирования программного кода Arduino IDE представлена на рис. 1.

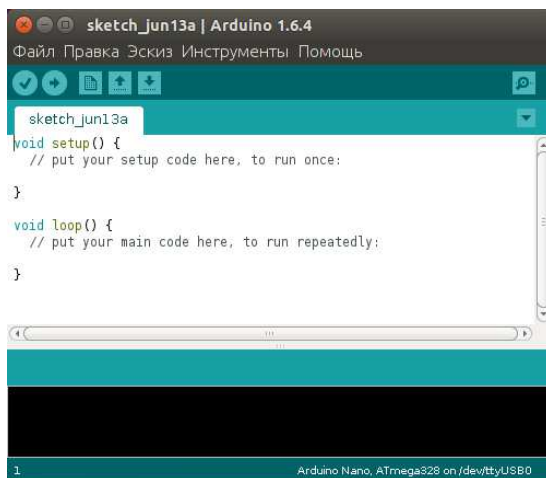


Рис. 1. Окно разработки программного кода Arduino IDE

Для примера разработаем программу для Arduino, которая предназначена для передачи строки через COM-порт. Ее задача состоит в проверке 3-го символа вводимой строки (начиная с нуля). Проверка именно этого символа связана с форматом команды (#AAN), которую мы будем формировать для OpenSCADA немного позже.

При соответствии введенного символа заданным в программе на экран монитора последовательного порта выводится соответствующая строка, например:

```
if (Comp("1")==0){Serial.println(">+1.100\r");}
```

Данное условие означает, что при вводе строки, в которой 3-м символом является «1», на экран монитора последовательного порта выводится строка (">+1.100"). В данной программе присутствуют еще два подобных условия. Если 3-й символ – это «3», выводится строка (">+1.300"), а если 3-й символ – «5», выводится строка (">+1.500"). Проверим работу программы.

На рис. 2 приведена проверка работы программы. Поочередно набрав в строку монитора последовательного порта #101, #103, #105 и после каждого ввода нажимая клавишу «Enter», наблюдаем результат выполнения программы.

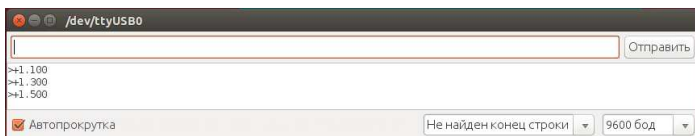


Рис. 2. Проверка выполнения программы, предназначенной для передачи строки через COM-порт

Как видно на рис. 2, программа выводит на экран соответствующие значения. Для нас это также стало показателем того, что связь с микроконтроллером Arduino установлена.

Конфигурирование языка запросов для микроконтроллера Arduino. Перед настройкой контроллера модуля DCON в OpenSCADA, необходимо настроить модуль транспорта Serial (последовательный интерфейс). Настраиваем исходящий транспорт. Сконфигурированный и запущенный исходящий транспорт открывает порт последовательного интерфейса для отправки запросов через него.

Во вкладке «Транспорты» переходим в «Последовательный интерфейс», а затем в «Выходной транспорт». На панели инструментов выбираем «Добавить элемент» и в строку «ID» вписываем название транспорта. На рис. 3 представлено окно конфигурации выходного транспорта.

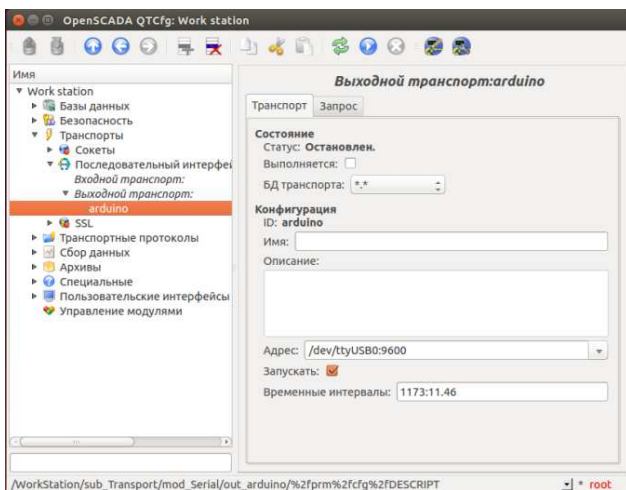


Рис. 3. Окно конфигурации выходного транспорта

В строке «Адрес», указываем адрес порта, который используется Arduino, и через двоеточие указываем скорость передачи данных, установленную при написании программы. В строке «Запрос» напишем «#101» и нажмем «Отправить». Как видно на рис. 4, в окне «Ответ» по запросу «#101» выводится соответствующее значение.

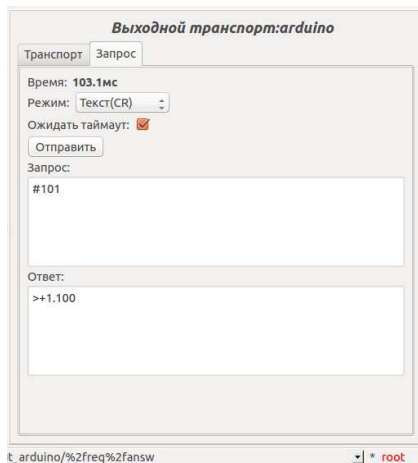


Рис. 4. Отправка запроса в Arduino через OpenSCADA

Теперь необходимо настроить контроллер модуля DCON. На рис. 5 представлено окно конфигурации контроллера модуля DCON.

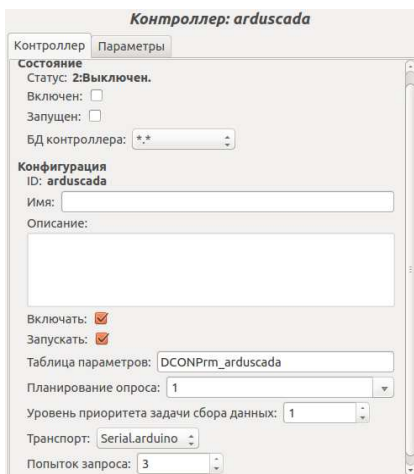


Рис. 5. Окно конфигурации контроллера модуля DCON

Из меню «Транспорт» выберем исходящий транспорт, созданный ранее.

Перейдем во вкладку «Параметры» и на панели инструментов выбираем «Добавить элемент» и в строку «ID» вписываем название параметра.

Окно параметра контроллера модуля DCON представлено на рис. 6.

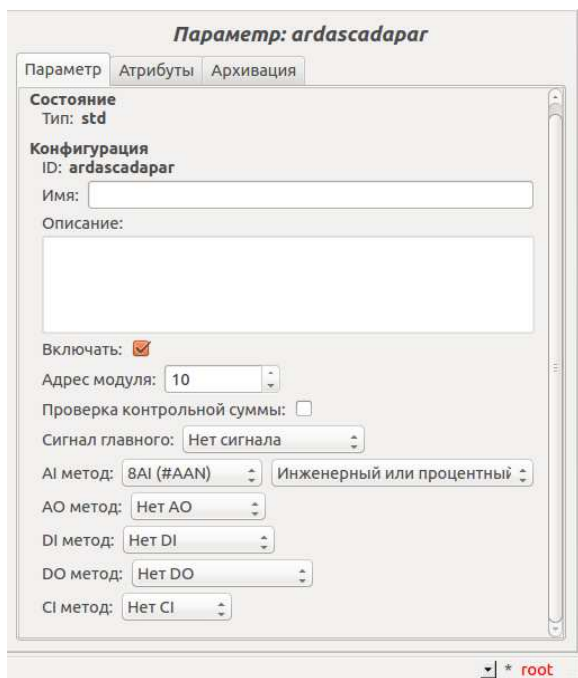


Рис. 6. Окно параметра контроллера модуля DCON

Укажем формат команды 8AI (#AAN), N – номер канала в модуле. Внутри Arduino программа принимает символы из COM-порта и определяет значение N. При этом адрес (AA) программа просто пропускает, так как он в данном простом примере не важен (в более сложных вариантах взаимодействия можно задействовать и поле адреса), и согласно заданному N возвращает (записывает в COM-порт) соответствующую строку.

На рис. 7 представлена вкладка «Атрибуты».

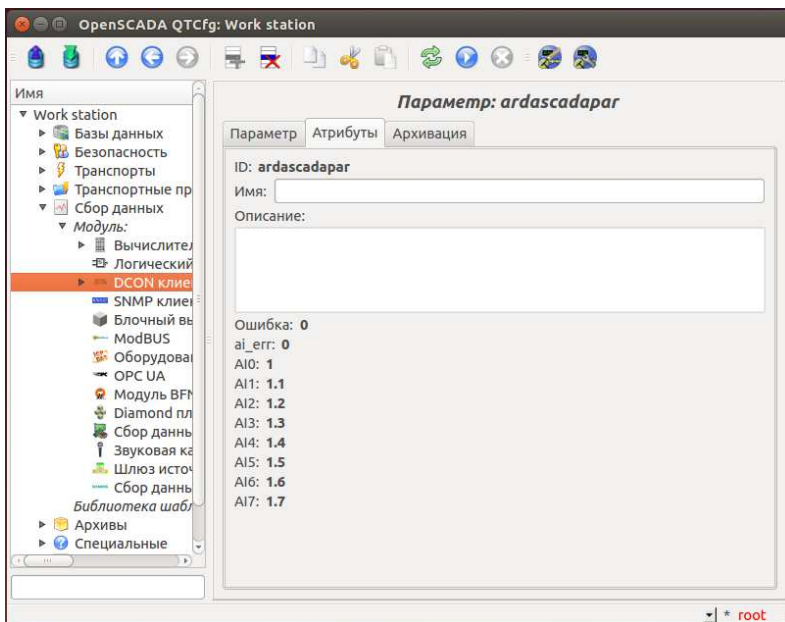


Рис. 7. Вкладка «Атрибуты» созданного параметра

В программе, загруженной в Arduino, были дополнительно добавлены условия для оставшихся N (чтобы был каждый от 0 до 7).

Заключение. В данной статье представлен пример конфигурирования языка запросов для микроконтроллера Arduino. Результаты, представленные на рис. 8, означают, что настройка проведена правильно. Выполняя данную задачу, мы удостоверились в том, что с помощью протокола DCON можно настроить связь между микроконтроллером Arduino и OpenSCADA. Также выявлено, что использование протокола DCON является простым способом организации взаимодействия между OpenSCADA и популярными микроконтроллерами семейства Arduino.

Библиографический список

1. Кавалеров М.В. К вопросу о термине «SCADA-система» // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2011. – № 5. – С. 205–209.

2. Организация взаимодействия с контроллерами [Электронный ресурс]. – URL: <http://asutp.ru/?p=600462> (дата обращения: 01.11.2015).

3. Кавалеров М.В., Матушкин Н.Н. Получение оценок параметров выполнения запросов реального времени в условиях планирования с фиксированными приоритетами // Нейрокомпьютеры: разработка, применение. – 2014. – № 12. – С. 7–13.

4. Кавалеров М.В., Матушкин Н.Н. Возможность существенного повышения эффективности планирования на основе непосредственного применения линейных интервальных ограничений реального времени // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2011. – № 5. – С. 4–13.

5. Софин Н.А., Кавалеров М.В. Модернизация программного обеспечения лабораторного стенда «Модель вентиляционной системы» // Инновационные технологии: теория, инструменты, практика: Материалы VI Междунар. интернет-конф. молодых ученых, аспирантов, студентов. – Пермь: Изд-во Перм. нац. исслед. политехн. ун-та, 2014. – С. 459–465.

6. Кавалеров М.В., Матушкин Н.Н. Мехатронная система «Шарик на балансирах с веб-камерой» для исследований на стыке компьютерного зрения и планирования задач реального времени // Управление в технических, эргатических, организационных и сетевых системах: материалы конф. / под ред. С.Н. Васильева, И.А. Каляева, Д.А. Новикова, Г.Г. Себрякова; 9–11 октября 2012, г. Санкт-Петербург. – СПб., 2012. – С. 744–747.

7. Савоченко Р.А. Модули <ModBus> подсистемы «Сбор данных» и подсистемы «Транспортные протоколы» [Электронный ресурс]. – URL: <ftp://oscada.org/OpenSCADA/0.7.1/doc/ru/Modules/DAQ.ModBus.pdf> (дата обращения: 02.11.2015).

8. Модуль подсистемы «Сбор данных» <DCON> [Электронный ресурс]. – URL: <http://wiki.oscada.org/Doc/DCON?v=baz> (дата обращения: 02.11.2015).

9. Энциклопедия АСУ ТП: Промышленные сети и интерфейсы: 2.10 Протокол DCON [Электронный ресурс]. – URL: http://www.bookasutp.ru/Chapter2_10.aspx (дата обращения: 01.11.2015).

РАЗРАБОТКА СИСТЕМЫ ВЕКТОРНОГО УПРАВЛЕНИЯ АСИНХРОННЫМ ДВИГАТЕЛЕМ С НАСТРОЙКОЙ РЕГУЛЯТОРОВ ПО ПАРАМЕТРАМ СХЕМЫ ЗАМЕЩЕНИЯ

Магистрант А.М. Шачков, инженер Е.М. Солодкий

Пермский национальный исследовательский
политехнический университет

Рассмотрено построение системы векторного управления с настройкой регуляторов тока и скорости по параметрам схемы замещения асинхронного двигателя. Представлены функциональная система векторного управления и структурная схема контура регулирования тока статора. Выполнена реализация системы управления в среде программирования MexBios Development Studio. Анализ результатов имитационного моделирования показал, что предложенная настройка регуляторов выполнена качественно и планируется дальнейшее исследование разработанной системы на лабораторной установке. Полученные результаты в дальнейшем будут использованы при разработке бездатчиковой системы управления асинхронным двигателем.

Ключевые слова: асинхронный двигатель, векторное управление, контур регулирования, синтез регулятора, схема замещения, имитационное моделирование.

DEVELOPMENT OF VECTOR CONTROL SYSTEM OF ASYNCHRONOUS MOTOR WITH REGULATORS ADJUSTMENT ON REPLACEMENT CIRCUIT PARAMETERS

Master Student A.M. Shachkov, Engineer E.M. Solodkiy

Perm National Research Polytechnic University

Development of vector control system with adjustment of current and speed regulators on replacement circuit parameters of asynchronous motor is considered. Functional system of vector control and the block diagram of control loop of stator current are presented. Realization of control system in programming environment of MexBios Development Studio is executed. The analysis of imitating modeling results showed offered regulators adjustment is performed qualitatively and research of developed system on laboratory stand is planned. The results will be used further when developing sensorless control system of asynchronous motor.

Keywords: asynchronous motor, vector control, control loop, regulator synthesis, replacement circuit, imitating modeling.

Одна из основных концепций векторного управления машиной переменного тока заключается в замене трехфазных синусоидальных переменных в двухфазные таким образом, чтобы управление асинхронным двигателем (АД) стало похоже на управление машиной

постоянного тока. Существует множество типов динамических моделей, которые используются для математического описания АД в системах векторного управления, и обычно в качестве базового представления АД выбирается Т-образная схема замещения [1]. Но так как для упрощенного имитационного моделирования переходных режимов АД достаточно одной индуктивности рассеяния, предлагается использовать Г-образную схему замещения, которая больше подходит для анализа динамических параметров АД и настройки регуляторов в системах векторного управления [2].

Для перехода к Г-образной схеме замещения необходимо переписать ее параметры, для чего введем постоянный коэффициент преобразования b и получим новые переменные:

$$\overline{\Psi}_R^s = b^2 L_r \overline{i}_R^s + b L_m \overline{i}_s^s, \quad (1)$$

$$\overline{i}_R^s = \frac{\overline{i}_r^s}{b}, \quad (2)$$

где $\overline{\Psi}_R^s$ – потокосцепление ротора, b – постоянная преобразования, L_r – индуктивность ротора Т-образной схемы замещения, \overline{i}_R^s – ток ротора, L_m – взаимная индуктивность Т-образной схемы замещения, \overline{i}_s^s – ток статора.

Анализируя полученную Г-образную схему замещения, можно сделать вывод, что магнитный поток ротора создается совместным действием токов статора и ротора. И тогда, учитывая, что коэффициенты при токах ротора и статора в уравнении (1) будут равны, постоянную преобразования b можно определить по выражению:

$$b^2 L_r = b L_m \Rightarrow b = \frac{L_m}{L_r}. \quad (3)$$

а взаимную индуктивность Г-образной схемы по уравнению:

$$\overline{\Psi}_R^s = \frac{L_m^2}{L_r} (\overline{i}_s^s + \overline{i}_R^s) = L_M (\overline{i}_s^s + \overline{i}_R^s) \Rightarrow L_M = \frac{L_m^2}{L_r}. \quad (4)$$

Подставив в известное уравнение потокосцепления статора [3] полученные выражения для коэффициента преобразования (3), тока ротора (2) и индуктивности намагничивания (4), можно получить выражение для индуктивности рассеяния L_σ :

$$\begin{aligned}\bar{\Psi}_s^s &= L_s \bar{i}_s^s + L_m \bar{i}_r^s = L_s \bar{i}_s^s + b L_m \bar{i}_R^s = L_s \bar{i}_s^s + \frac{L_m^2}{L_r} \bar{i}_R^s = L_s \bar{i}_s^s + L_M \bar{i}_R^s = \\ &= (L_\sigma + L_M) \bar{i}_s^s + L_M \bar{i}_R^s \Rightarrow L_\sigma = L_s - L_M,\end{aligned}\quad (5)$$

где $\bar{\Psi}_s^s$ – потокосцепление статора, L_s – индуктивность статора Т-образной схемы замещения, \bar{i}_r^s – ток ротора Т-образной схемы замещения.

Для получения выражения для определения сопротивления ротора записанные ранее уравнения (1) и (2) подставляем в известное уравнение равновесия ротора [3] и в результате преобразований получаем выражение:

$$R_R = b^2 R_r, \quad (6)$$

где R_r – сопротивление ротора Т-образной схемы замещения, R_R – сопротивление ротора Г-образной схемы замещения

При рассмотрении Г-образной схемы замещения АД используются уравнения математического описания АД в стационарных осях координатах α – β . Однако для реализации системы векторного управления АД удобнее использовать уравнения во вращающейся системе координат d – q [3], которая имеет ту же скорость вращения, что и поток ротора, а ось d ориентирована по направлению потока ротора.

Обозначим угол между системами координат d – q и α – β как θ_1 , а скорость вращения системы координат d – q как ω_1 , тогда динамические уравнения модели асинхронного двигателя в координатах d – q могут быть представлены в виде:

$$L_\sigma \frac{d\bar{i}_s^s}{dt} = \bar{u}_s - (R_s + R_R + j\omega_1 L_\sigma) \bar{i}_s^s - (j\omega_r - \frac{R_R}{L_M}) \bar{\Psi}_R, \quad (7)$$

$$\frac{d\bar{\Psi}_R}{dt} = R_R \bar{i}_s^s - (\frac{R_R}{L_M} - j(\omega_r - \omega_1)) \bar{\Psi}_R, \quad (8)$$

где \bar{u}_s – напряжение статора, R_s – сопротивление статора.

Далее рассмотрим синтез контуров регулирования тока и скорости системы векторного управления асинхронным двигателем [4]. Функциональная схема системы ориентированного векторного управления, выбранная для анализа и имитационного моделирования, представлена на рис. 1.

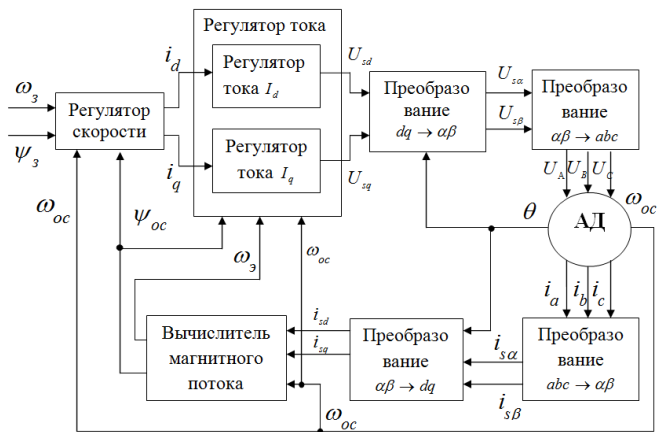


Рис. 1. Функциональная схема системы векторного управления АД

Настройка регуляторов тока осуществляется во вращающейся системе координат по осям d - q . Общая структура регулятора тока представлена на рис. 2.

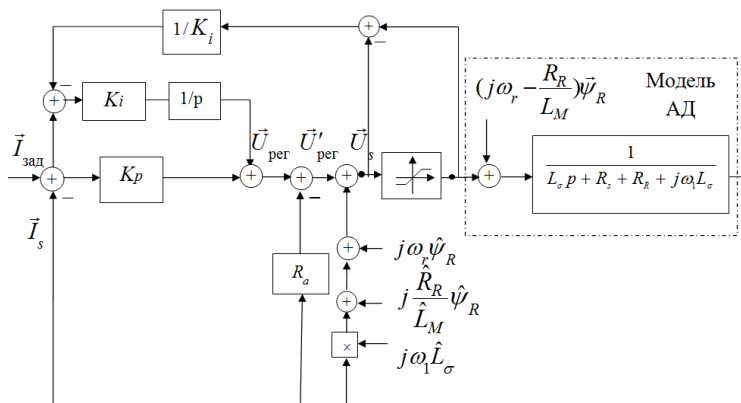


Рис. 2. Структурная схема контура регулирования тока

Передаточная функция замкнутого контура регулирования тока на представленной структурной схеме (см. рис. 2) определяется выражением:

$$G_c(p) = \frac{F_c(p)G'(p)}{1 + F_c(p)G'(p)}, \quad (9)$$

где $F_c(p)$ – передаточная функция ПИ-регулятора.

Зададимся желаемой передаточной функцией замкнутого контура:

$$G_c(p) = \frac{a_c}{p + a_c}, \quad (10)$$

где a_c – коэффициент, учитывающий пропускную способность замкнутого контура регулирования тока.

В результате элементарных преобразований получаем выражение для расчета параметров ПИ-регулятора тока:

$$K_p = \alpha_c L_\sigma \text{ и } K_i = \alpha_c (R_s + R_R + R_a), \quad (11)$$

где R_a – добавочное сопротивление, $R_a = \alpha_c L_\sigma - R_s - R_R$.

Далее для синтеза контура регулирования скорости запишем динамическое уравнение механической части асинхронного электропривода в виде

$$\frac{d\omega}{dt} = -\frac{k_{тр} \cdot \omega}{J} + \frac{M - M_c}{J}, \quad (12)$$

где ω – механическая угловая скорость ротора, M – момент двигателя, M_c – момент сопротивления, $k_{тр}$ – коэффициент трения, J – момент инерции.

Структурная схема регулятора скорости может быть получена по аналогии с контуром регулирования тока. Передаточная функция механической части согласно выражению (12) определяется в виде

$$\frac{\omega_r(p)}{M(p)} = \frac{1}{J + k_{тр}}. \quad (13)$$

Зададимся желаемой передаточной функцией замкнутого контура регулирования скорости в виде

$$\frac{\Omega_r(p)}{\Omega_{r.ref}(p)} = \frac{a_\omega}{p + a_\omega}, \quad (14)$$

где a_ω – коэффициент, учитывающий пропускную способность замкнутого контура регулирования скорости.

Поскольку электромагнитные процессы в асинхронном электроприводе протекают гораздо быстрее, чем механические, то можно считать, что выходной сигнал регулятора скорости воздействует на двигатель почти мгновенно. При этом внутренний контур регулиро-

вания тока при синтезе можно рассматривать безинерционным звеном с единичным коэффициентом усиления. Для этого необходимо при настройке регулятора скорости учесть, что его быстродействие гораздо меньше, чем регулятора тока, поэтому коэффициент a_ω , учитывающий пропускную способность контура скорости, рекомендуется выбирать в пределах от 10 до 100.

С учетом введенных допущений после элементарных преобразований получаем выражения для расчета параметров ПИ-регулятора скорости:

$$K_p = \alpha_\omega J \text{ и } K_i = \alpha_\omega (B + B_a), \quad (15)$$

где B_a – коэффициент активного демпфирования, $B_a = \alpha_\omega J - B$.

На рис. 3 представлена реализация ПИ-регулятора скорости в среде MexBios Development Studio [5].

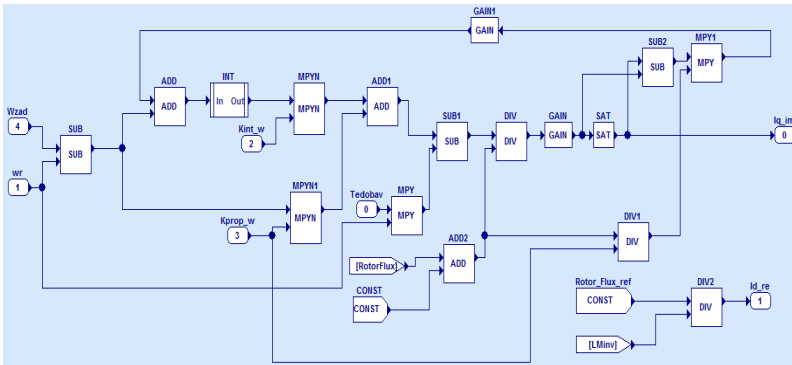


Рис. 3. Реализация регулятора скорости в MexBIOS

Перейдем непосредственно к имитационному моделированию разработанной системы векторной управления. При моделировании использованы параметры двигателя, представленные ниже.

Параметры имитационной модели асинхронного двигателя

$U_{нл}$, В	$f_{н}$, Гц	R_s , Ом	R_r , Ом	L_m , Гн	$L_{\sigma s}$, Гн	$L_{\sigma r}$, Гн	J , кг·м ²	Z_p	$F_{тр}$, Н·м·с
400	50	1,405	1,395	0,1722	0,00583	0,00583	0,0131	2	0,00298

Графики переходных процессов, полученные по результатам имитационного моделирования, приведены на рис. 4.

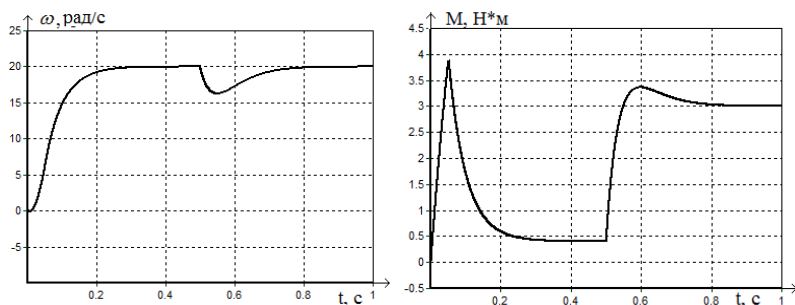


Рис. 4. Графики переходных процессов по скорости и моменту

При моделировании системы векторного управления АД принято, что в начальный момент времени задание скорости составляет 20 рад/с и двигатель запускается без нагрузки. После выхода на заданную скорость (в момент времени $t = 0,5$ С) выполняется наброс нагрузки. Анализируя полученные графики переходных процессов скорости и момента, можно сделать вывод, что настройка параметров ПИ-регуляторов тока и скорости по параметрам схемы замещения выполнена качественно и привод должным образом отрабатывает задающее и возмущающее воздействие.

На следующем этапе работ планируется реализация разработанных алгоритмов управления на микропроцессорной платформе и исследование разработанной системы векторного управления на лабораторной установке.

В дальнейшем реализованные в среде MexBIOS имитационные модели могут быть успешно использованы при разработке бездатчиковых систем векторного управления [6] с добавлением модулей косвенного вычисления скорости и угла магнитного потока ротора.

Библиографический список

1. Анучин А.С. Системы управления электроприводов: учебник для вузов. – М.: Изд. дом МЭИ, 2015. – 338 с.
2. Даденков Д.А., Солодкий Е.М., Шачков А.М. Моделирование системы векторного управления асинхронным двигателем в пакете Matlab/Simulink // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2014. – № 3(11). – С. 117–128.

3. Качачёв Ю.Н. Векторное регулирование (заметки практика) [Электронный ресурс]. – URL: http://www.privod-news.ru/docs/Vector_Kalachev.pdf (дата обращения: 20.10.2015).

4. Костыгов А.М., Солодкий Е. М., Даденков Д.А. Синтез регуляторов тока и скорости в системе векторного управления вентильным электроприводом // *Фундаментальные исследования*. – 2014. – № 7 (11). – С. 1490–1495.

5. MexBIOSDevelopmentStudio – визуальная среда разработки и моделирования [Электронный ресурс]. – URL: <http://mechatronica-pro.com/ru/catalog/software/> (дата обращения: 20.04.2015).

6. Бездатчиковое векторное управление вентильным двигателем с коррекцией ошибки вычисления электрического угла / А.М. Костыгов, Д.А. Даденков, Е.М. Солодкий, А.М. Шачков // *Информационно-измерительные и управляющие системы*. – 2015. – № 9(13). – С. 27–33.

Секция 3

**ИННОВАЦИОННЫЕ НАПРАВЛЕНИЯ В ЭНЕРГЕТИКЕ.
ЭНЕРГОРЕСУРСОСБЕРЕЖЕНИЕ**

ИСПОЛЬЗОВАНИЕ НЕЙРО-НЕЧЕТКИХ МОДЕЛЕЙ ДЛЯ КОМПЛЕКСНОЙ ОЦЕНКИ СОСТОЯНИЯ ЭЛЕКТРОТЕХНИЧЕСКОГО ОБОРУДОВАНИЯ

Студентка В.В. Бояршинова,
канд. техн. наук, доцент Д.К. Елтышев
Пермский национальный исследовательский
политехнический университет

В статье рассмотрены вопросы применения современных интеллектуальных технологий для комплексной оценки состояния электротехнического оборудования. Проведен анализ методов оценки состояния, основанных на использовании нейросетевых моделей, нечеткой логики и гибридных нейро-нечетких систем класса ANFIS. В качестве примера использования нейро-нечеткого подхода разработана упрощенная модель оценки состояния силового маслонаполненного трансформаторного оборудования.

Ключевые слова: электротехническое оборудование, техническое состояние, диагностика, мониторинг, интеллектуальные технологии, нейронные сети, нечеткая логика.

THE USE OF NEURO-FUZZY MODELS FOR INTEGRATED ASSESSMENT OF THE ELECTRICAL EQUIPMENT CONDITION

Student V.V. Boyarshinova
Ph.D. in Technical Sciences, Associate Professor D.K. Eltyshv
Perm National Research Polytechnic University

The article deals with the application of modern intelligent technologies for integrated assessment of electrical equipment condition. The methods for assessing the condition based on the use of neural network models, fuzzy logic and hybrid neuro-fuzzy systems such as ANFIS, have been analyzed. As an example of the use of neuro-fuzzy approach there has been developed a simplified model for assessing oil-filled power transformer equipment condition.

Keywords: electrical equipment, technical condition, monitoring, diagnosis, intelligent technologies, neural networks, fuzzy logic.

Введение. Вопросы оценки технического состояния электротехнического оборудования (ЭО) относятся к одной из наиболее важных задач надежного функционирования электроэнергетических объектов. Это обусловлено, во-первых, высоким уровнем износа ЭО по продолжительности срока службы, во-вторых, отсутствием простых и однозначных критериев определения дефектов и срока службы для различных видов ЭО и, в-третьих, необходимостью обеспечения тре-

бований безопасности. Поэтому особую роль и значимость приобретают модели обслуживания ЭО по фактическому техническому состоянию, знание которого позволяет предотвратить возможные аварийные ситуации, продлить срок межремонтной эксплуатации и сократить время простоев оборудования, улучшив его показатели надежности и снизив затраты [3].

Постановка задачи исследования. Для решения задач, связанных с накоплением истории и формированием базы мониторинга эксплуатируемых объектов, анализом надежности, а также выявлением причин и факторов, приводящих к отказам или авариям, необходимо иметь представление о фактическом техническом состоянии (ТС) ЭО. Данная информация должна являться основой для своевременного определения потребности оборудования в ремонте, а также прогнозирования возможного момента, когда значения его ключевых показателей (состояния элементов) достигнут критической отметки [3].

Таким образом, задачу комплексной оценки состояния ЭО можно представить в виде функции следующего вида:

$$S = f(X, M, N, P), \quad (1)$$

где $X = \{x_1, \dots, x_n\}$ – перечень контролируемых средствами мониторинга и диагностики физико-химических параметров, которые характеризуют изменение ТС ЭО под воздействием различных факторов; $M = \{m_1, \dots, m_k\}$ – различные методы, используемые для контроля параметров X ; $N = \{n_1, \dots, n_n\}$ – нормативные значения для каждого исследуемого параметра X ; $P = \{p_1, \dots, p_l\}$ – правила принятия решений, характеризующие связь значений параметров X с состоянием ЭО.

Неструктурированный и плохо формализуемый характер задачи (1) [2], связанный с неполнотой и неоднозначностью исходных данных, сложностью определения связей между значениями параметров ЭО и признаками дефектных состояний, необходимостью использования экспертных знаний для повышения объективности результата и многофакторности процесса принятия решений, ограничивает возможность применения традиционных методов, предполагающих обработку больших объемов статических данных [5].

Одним из эффективных способов решения данной проблемы может являться использование интеллектуальных методов и технологий, в частности, аппарата нечеткой логики и нейросетевого моделирования.

Анализ интеллектуальных методов оценки ТС ЭО. Нейронные сети представляют собой метод моделирования сложных зависимостей в задачах прогноза, классификации, управления и диагностики [1] с возможностью обучения и адаптации к различным изменениям.

К основным преимуществам моделей оценки состояния объектов, построенных на основе нейронных сетей, можно отнести [8]: возможность моделирования задач с большим числом переменных; минимальное вмешательство пользователя в процесс обучения; минимальные требования к структуре нейронной сети; быстрота и параллельность вычислений; возможность работы с разнородной информацией и др.

Среди недостатков нейросетевого подхода можно отметить: отсутствие формализованных алгоритмов настройки сети; отсутствие гарантий успешного решения задачи и высокую сложность внутренней структуры сети.

Под нечеткой логикой понимается многозначная логика, позволяющая дать промежуточные значения для общепринятых оценок [1, 11], особенно при недостаточном знании объекта, но при наличии опыта работы с ним, а также в случаях, когда по условию задачи необходимо использовать знания эксперта. Данный аппарат чаще всего применяется при построении моделей, которые сложно идентифицировать.

Очевидными преимуществами использования нечеткой логики в задачах оценки состояния ЭО являются: простота построения моделей, в том числе с использованием опыта человека; высокое качество результатов; сокращение объема вычислений и высокое быстродействие.

К недостаткам моделей, разработанных на основе нечеткой логики, можно отнести: отсутствие возможности анализа полученных моделей с использованием традиционных методов; повышение сложности вычислений и необходимость расширения базы правил при увеличении числа входных переменных.

Для повышения эффективности решения задач оценки состояния ЭО целесообразно объединить преимущества двух рассмотренных выше методов и использовать так называемую гибридную сеть, которая представляет собой многослойную нейронную сеть со специальной структурой без обратных связей [4]. Ключевая идея данного подхода состоит в использовании существующей выборки для оптими-

зации параметров нечеткой модели (с помощью известных процедур обучения нейросетей) с целью повышения точности соответствия результата.

Реализовать указанный выше способ можно с применением адаптивной системы нейронечеткого вывода ANFIS, которая, с одной стороны, представляет собой нейронную сеть с одним выходом и несколькими входами, а с другой стороны, является системой нечеткого вывода типа Сугено, в которой каждое из правил нечетких продукций имеет свой постоянный вес.

Исследование возможности применения нейронечетких моделей для оценки состояния ЭО. В качестве примера использования ANFIS для оценки ТС ЭО построим упрощенную модель оценки состояния силовых маслонаполненных трансформаторов по двум параметрам: x_1 – температура верхних слоев масла трансформатора; и x_2 – температура нагрева обмоток трансформатора. На практике перечень контролируемых параметров должен быть расширен с учетом особенностей работы оборудования и опыта специалистов. При этом предполагается, что при каждом обследовании трансформатора в режиме мониторинга проверяется и записывается температура всех ключевых параметров трансформатора для обеспечения его бесперебойной работы.

В нормативно-технической документации [9] оговаривается предельное значение температуры верхних слоев масла. При номинальной нагрузке их температура не должна превышать 95 °С при естественном масляном охлаждении или с обдувом вентиляторами. С использованием различных средств охлаждения и обдува температура будет варьироваться в пределах от 55–95 °С. Также нормами оговаривается и значение температуры нагрева обмоток трансформатора, которая изменяется в пределах от 65 до 105 °С (превышение температуры обмоток, равное 70 °С, соответствует их номинальной нагрузке при температуре окружающего воздуха 35 °С). Допустимая температура нагрева определяется тем, что износ изоляции обмоток должен обеспечивать срок службы трансформаторов 20–25 лет.

Если температура масла или обмоток превышает допустимую, нужно выяснить причины и принять меры к устранению неисправности. В первую очередь следует проверить исправность системы охлаждения. Если в системе охлаждения неисправностей не обнаружено, то повышение температуры масла или обмоток в большинстве

случаев свидетельствует о возникновении внутренних повреждений в трансформаторе.

Рассмотрим задачу классификации состояния трансформатора и в качестве выходной переменной модели y примем параметр, характеризующий различные уровни состояния: «Хорошее», «В пределах допуска», «Аварийное».

Для разработки модели оценки ТС трансформатора использован тестовый набор данных, сгенерированный на основе анализа нормативно-технической документации и содержащий в себе измерения значений температур масла и обмоток трансформатора, а также соответствующие им уровни его технического состояния.

Система нечеткого вывода (типа Сугено) для рассматриваемой задачи имеет вид, представленный на рис. 1, и сгенерирована с использованием метода решетки [11]. Каждая из входных переменных модели описывается тремя термами: «Низкая», «Средняя», «Высокая» с гауссовыми функциями принадлежности.

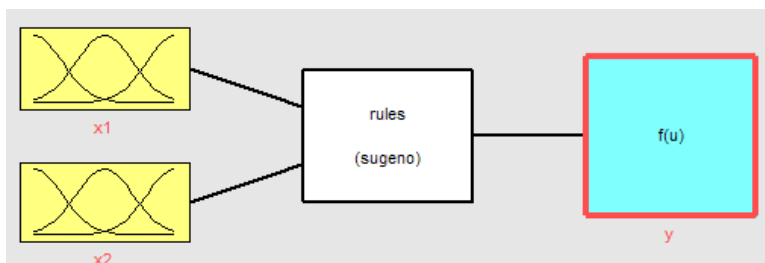


Рис. 1. Модель сгенерированной системы нечеткого вывода для оценки состояния трансформатора

Для того чтобы модель работала нужным образом, необходимо задаться рядом правил, рассмотрев те ситуации (диапазоны температур), при которых нечеткая система будет выдавать один из трех возможных состояний трансформатора (в норме, в пределах допуска, аварийное). Поэтому база знаний системы включает 9 основных правил:

ЕСЛИ (x_1 = «Низкая») И (x_2 = «Низкая») ТО (y = «Хорошее»);

ЕСЛИ (x_1 = «Низкая») И (x_2 = «Средняя») ТО (y = «В пределах допуска»);

ЕСЛИ (x_1 = «Низкая») and (x_2 = «Высокая») ТО (y = «Аварийное»);

ЕСЛИ (x_1 = «Средняя») И (x_2 = «Низкая») ТО (y = «В пределах допуска»);

ЕСЛИ ($x_1 = \text{«Средняя»}$) И ($x_2 = \text{«Средняя»}$) ТО ($y = \text{«В пределах допуска»}$);

ЕСЛИ ($x_1 = \text{«Средняя»}$) И ($x_2 = \text{«Высокая»}$) ТО ($y = \text{«Аварийное»}$);

ЕСЛИ ($x_1 = \text{«Высокая»}$) И ($x_2 = \text{«Средняя»}$) ТО ($y = \text{«Аварийное»}$);

ЕСЛИ ($x_1 = \text{«Высокая»}$) И ($x_2 = \text{«Низкая»}$) ТО ($y = \text{«Аварийное»}$);

ЕСЛИ ($x_1 = \text{«Высокая»}$) И ($x_2 = \text{«Высокая»}$) ТО ($y = \text{«Аварийное»}$).

Для простоты обозначения состояний системы было принято решение выразить их с помощью постоянных величин (классов), т.е. значение «Аварийное» = 3, «В пределах допуска» = 2 и «Хорошее» = 1.

На рис. 2 представлена структура полученной адаптивной модели ANFIS для оценки состояния трансформатора, содержащая 6 слоев.

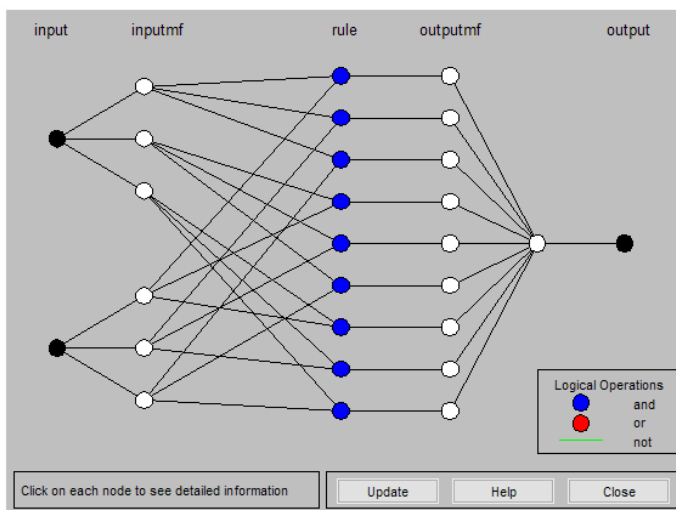


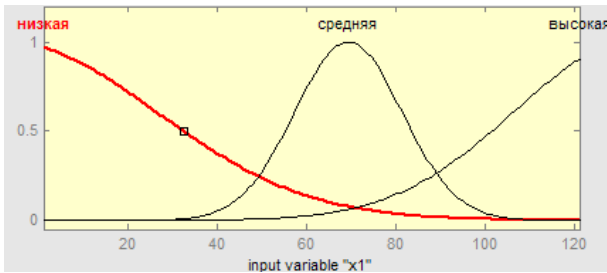
Рис. 2. Структура системы нейронечеткого вывода

Первый слой представляет собой набор входных элементов (x_1 и x_2), которые являются управляемыми переменными.

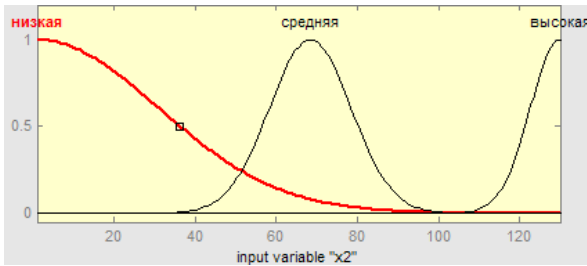
Второй слой определяет послылки нечетких правил. Каждый узел этого слоя соответствует одному нечеткому правилу и соединен с теми узлами первого слоя, которые формируют предпосылки данного правила. Третий слой присваивает степени важности выполнения правил.

Узлы данного слоя рассчитывают относительный вес выполнения правила. Четвертый слой определяет вклад каждого нечеткого правила в выход сети, а пятый слой формирует управляющий сигнал на выходной элемент (шестой слой).

После обучения модели на заданной тестовой выборке получены итоговые функции принадлежности (рис. 3 *а, б*) для каждой из входных переменных.



а



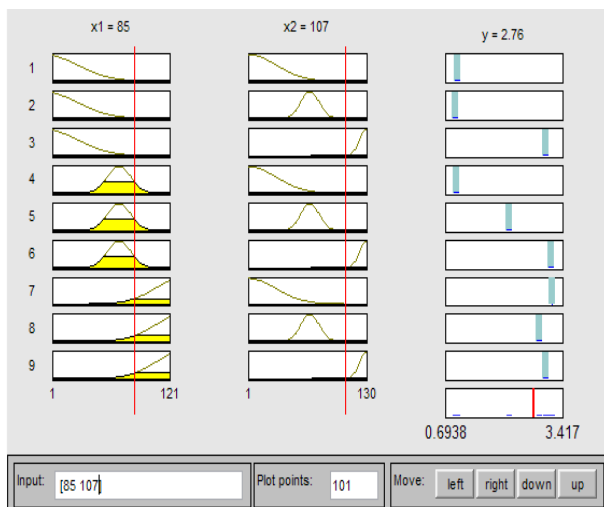
б

Рис. 3. Функции принадлежности входов модели:
а – x_1 – температура масла трансформатора;
б – x_2 – температура обмоток трансформатора

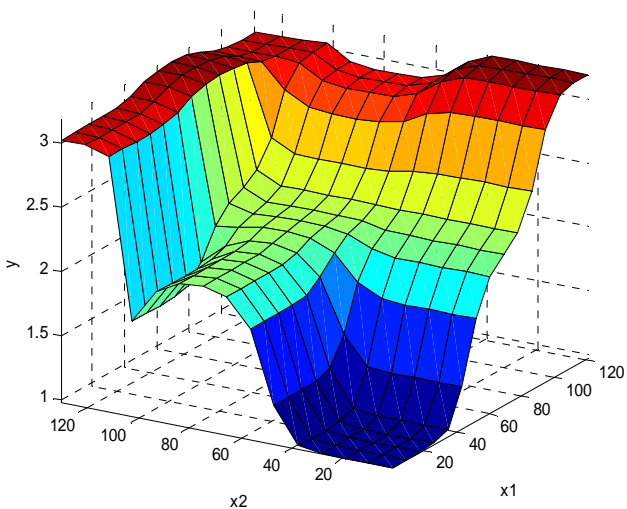
Проверка правильности работы нейронечеткой системы определялась с помощью задания произвольных значений температур (рис. 4, *а*) при округлении значений выходного параметра y по общепринятым правилам (для отнесения к классам состояния) в соответствии с полученной диагностической функцией (рис. 4, *б*).

На рис. 5 приведено распределение оценок состояния трансформатора полученных из тестового набора данных и с использованием модели. При этом ошибка классификации равна 14 %, что является

приемлемым результатом при достаточно небольшом наборе используемых для обучения исходных данных.



a



б

Рис. 4. Оценка ТС трансформатора при проверке правильности работы модели:
a – процесс нахождения результата; *б* – поверхность отклика

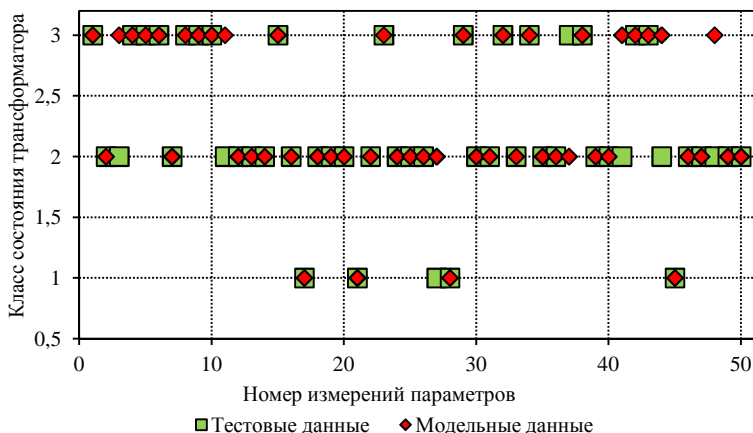


Рис. 5. Представление результатов тестирования ANFIS модели

Заключение. В статье дана постановка задачи комплексной оценки ТС ЭО. На основе сравнительного анализа современных интеллектуальных методов сделан вывод о возможности использования для оценки ТС ЭО нейронечетких моделей. На этапе апробации выбранного подхода разработана гибридная сеть ANFIS для оценки ТС силового трансформатора по двум ключевым параметрам (температура верхних слоев масла и температура обмоток изоляции). Тестирование полученной нейронечеткой структуры показало, что она с точностью до 86 % правильно определяет фактическое состояние ЭО. Для того чтобы ошибка классификации состояния была минимальной, необходимо увеличить объем обучающих данных для существующей нейронечеткой сети.

Библиографический список

1. Борисов В.В., Круглов В.В., Федулов А.С. Нечеткие модели и сети. – М: Горячая линия-Телеком, 2007. – 283 с.
2. Бочкарев С.В., Петроченков А.Б., Ромодин А.В. Интегрированная логистическая поддержка эксплуатации электротехнических изделий: учеб. пособие. – Пермь: Изд-во Перм. гос. техн. ун-та, 2009. – 398 с.
3. Бояршинова В.В., Елтышев Д.К. Комплексная интеллектуальная оценка состояния электротехнического оборудования // Автоматизированные системы управления и информационные технологии:

материалы всерос. науч.-практ. конф.; г. Пермь, 15 мая 2015 г. – Пермь: Изд-во Перм. нац. исслед. политехн. ун-та, 2015. – С. 243–248.

4. Гаглоева И.Э. Алгоритм функционирования интеллектуальной системы поддержки принятия решений для оценки технического состояния и эффективности процесса обновления и модернизации производственных фондов электроэнергетических объектов // Научное ведение: интернет-журнал. – 2013. – № 3 (16). – С. 1–9.

5. Глущенко П.В. Диагностирование электротехнических объектов на основе моделей: монография. – СПб.: Изд-во СПГУВК, 1996. – 100 с.

6. Леоненков А.В. Нечеткое моделирование в среде MatLab и fuzzyTECH. – СПб.: БХВ-Петербург, 2005. – 719 с.

7. Назарычев А.Н. Основные принципы и критерии управления техническим состоянием электрооборудования // Вестник ИГЭУ. – 2006. – № 2. – С. 67–71.

8. РД 34.45-51.300-97. Объем и нормы испытаний электрооборудования. – М.: Атомиздат, 2001. – 154 с.

9. Ярушкина Н.Г. Основы теории нечетких и гибридных систем: учеб. пособие. – М.: Финансы и статистика, 2004. – 313 с.

10. Штовба С.Д. Проектирование нечетких систем средствами MatLab. – М.: Горячая линия-Телеком, 2007. – 288 с.

РАСЧЕТ РАБОЧИХ ХАРАКТЕРИСТИК СИНХРОННОГО ДВИГАТЕЛЯ В ДВУХМАШИННОМ АГРЕГАТЕ НАГРУЗОЧНОГО УСТРОЙСТВА СИНХРОННЫХ МАШИН

Студент Д.И. Косташ, канд. техн. наук, доцент В.А. Трефилов
Пермский национальный исследовательский
политехнический университет

В статье представлен расчет характеристик синхронного двигателя в нагрузочном устройстве синхронных машин. Приведены рабочие характеристики синхронного двигателя, рассчитанные по уравнениям в координатной системе $d-q$.

Ключевые слова: нагрузочное устройство, синхронные машины, расчет рабочих характеристик.

THE SYNCHRONOUS MOTOR CHARACTERISTICS CALCULATION INTO THE SYNCHRONOUS MACHINES LOAD DEVICE OF TWO MACHINES GENERATORS

Student D.I. Kostash,
Ph.D. in Technical Sciences, Associate Professor V.A. Trefilov
Perm National Research Polytechnic University

The article presents the calculation of the characteristics of the synchronous motor in a load device of synchronous machines. Contains the operational characteristics of the synchronous motor calculated by the equations in the coordinate system $d-q$.

Keywords: the load device, a synchronous machine, calculation of performance characteristics.

Схема двухмашинного агрегата нагрузочного устройства представлена на рис.1.

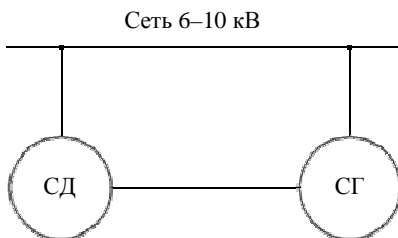


Рис. 1. Схема нагрузочного устройства синхронных машин

Нагрузочное устройство представляет собой две синхронные машины одинаковой мощности, валы роторов которых жестко соединены между собой. Одна из машин СД работает в режиме двигателя, потребляет энергию из сети и преобразует ее в механическую энергию. Другая машина СГ работает в режиме генератора и отдает электрическую энергию обратно в сеть.

В нагрузочном устройстве угловые положения роторов электрических машин смещены относительно друг друга. Ротор генератора поворачивают относительно ротора двигателя по ходу вращения на угол γ , а затем валы машин жестко соединяют между собой. Это приводит к относительному смещению угловых характеристик синхронных машин, которые при угле рассогласования $\gamma = \pi/2$ представлены на рис. 2. На рисунке видно, что установившийся режим работы машин характеризуется углом нагрузки θ_n , при этом одна машина работает двигателем, потребляя электрическую энергию из сети (т. 1), а вторая – генератором, отдавая её в сеть (т. 2).

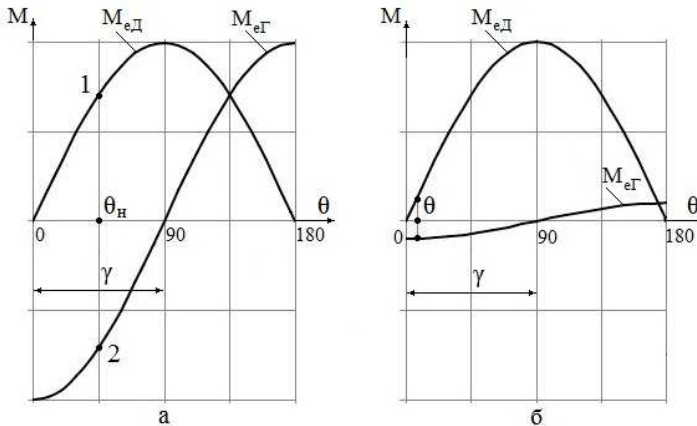


Рис. 2. Угловые характеристики электрических машин

Переходные процессы в двухмашинном агрегате нагрузочного устройства можно описать в ортогональной системе координат $d-q$, жёстко связанной с ротором синхронной машины [1].

Статические характеристики двухмашинного агрегата могут быть рассчитаны по уравнениям установившегося режима, которые получаются из уравнений переходных процессов, если принять в них

производные потокоцеплений и частоты вращения равными нулю, а частоту вращения роторов синхронных машин принять равной синхронной скорости [2].

Статический режим работы машин в нагрузочном устройстве описывается следующими уравнениями:

– для синхронного двигателя:

$$\begin{aligned} U_d + \omega \cdot \Psi_q - r_c \cdot I_d = 0; \quad U_q - \omega \cdot \Psi_d - r_c \cdot I_q = 0, \\ U_f - r_f \cdot I_f = 0. \end{aligned} \quad (1)$$

– для синхронного генератора:

$$\begin{aligned} U_{d\Gamma} + \omega \cdot \Psi_{q\Gamma} - r_{c\Gamma} \cdot I_{d\Gamma} = 0, \\ U_{q\Gamma} - \omega \cdot \Psi_{d\Gamma} - r_{c\Gamma} \cdot I_{q\Gamma} = 0, \\ U_{f\Gamma} - r_{f\Gamma} \cdot I_{f\Gamma} = 0. \end{aligned} \quad (2)$$

Напряжение статорных обмоток двигателя по осям d и q :

$$U_d = -U_c \cdot \sin \theta; \quad U_q = U_c \cdot \cos \theta. \quad (3)$$

Напряжение статорных обмоток генератора по осям d и q :

$$U_{d\Gamma} = -U_c \cdot \sin(\theta - \gamma); \quad U_{q\Gamma} = U_c \cdot \cos(\theta - \gamma). \quad (4)$$

Токи возбуждения машин считаются известными и определяются из соотношений:

$$I_f = \frac{U_f}{R_f}; \quad I_{f\Gamma} = \frac{U_{f\Gamma}}{R_{f\Gamma}}. \quad (5)$$

Потокоцепления машин по осям d и q можно записать в виде:

– для синхронного двигателя:

$$\Psi_d = x_{ad}(I_d + I_f); \quad \Psi_q = x_{aq} \cdot I_q, \quad (6)$$

– для синхронного генератора:

$$\Psi_{d\Gamma} = x_{ad\Gamma}(I_{d\Gamma} + I_{f\Gamma}); \quad \Psi_{q\Gamma} = x_{aq\Gamma} \cdot I_{q\Gamma}. \quad (7)$$

Электромагнитные моменты двигателя и генератора:

$$\begin{aligned} M_{ед} = I_d \cdot I_q \cdot (x_{ad} - x_{aq}) + I_q \cdot I_f \cdot x_{ad}, \\ M_{ег} = I_{d\Gamma} \cdot I_{q\Gamma} \cdot (x_{ad\Gamma} - x_{aq\Gamma}) + I_{q\Gamma} \cdot I_{f\Gamma} \cdot x_{ad\Gamma}. \end{aligned} \quad (8)$$

Активные мощности двигателя и генератора:

$$\begin{aligned} P_{1д} = U_d \cdot I_d + U_q \cdot I_q, \\ P_{1г} = U_{d\Gamma} \cdot I_{d\Gamma} + U_{q\Gamma} \cdot I_{q\Gamma}. \end{aligned} \quad (9)$$

Реактивные мощности двигателя и генератора:

$$\begin{aligned} Q_{1д} &= U_q \cdot I_d - U_d \cdot I_q, \\ Q_{1г} &= U_{qг} \cdot I_{dг} - U_{dг} \cdot I_{qг}. \end{aligned} \quad (10)$$

Коэффициенты мощности двигателя и генератора:

$$\begin{aligned} \cos \phi_{1д} &= P_{1д} / \sqrt{P_{1д}^2 + Q_{1д}^2}, \\ \cos \phi_{1г} &= P_{1г} / \sqrt{P_{1г}^2 + Q_{1г}^2}. \end{aligned} \quad (11)$$

Модули токов статорных обмоток синхронных машин:

$$I_{1д} = \sqrt{I_d^2 + I_q^2}, \quad I_{1г} = \sqrt{I_{dг}^2 + I_{qг}^2}. \quad (12)$$

Расчет статического режима проводится итерационным методом по углу нагрузки θ , при котором должно выполняться условие равенства моментов:

$$M_{ед} + M_{ег} - M_c \leq \varepsilon, \quad (14)$$

где M_c – момент сопротивления на валу агрегата, определяемый механическими потерями машин; ε – допустимая погрешность, выбираемая по соображениям точности расчета.

Следует отметить особенность создания нагрузочного режима синхронных машин в данном нагрузочном устройстве. Она состоит в том, что для этого необходимо деформировать угловую характеристику одной из синхронных машин устройства, а именно для создания режима нагрузки синхронного двигателя необходимо изменять угловую характеристику генератора, и, наоборот, чтобы нагрузить синхронный генератор необходимо деформировать угловую характеристику двигателя.

Рассмотрим, как создается режим нагрузки синхронного двигателя. Двигатель возбуждается номинальным током, его угловая характеристика имеет вид, представленный на рис. 2. Для изменения нагрузочного режима двигателя деформируем угловую характеристику синхронного генератора, изменяя его ток возбуждения. При малом токе возбуждения критический момент генератора станет меньше, его угловая характеристика по оси моментов сожмётся, и установившийся режим наступит при малом угле нагрузки θ (рис. 2, б).

Если увеличить ток возбуждения генератора, то критический момент машины увеличится, угловая характеристика генератора по оси моментов растянется, и установившийся режим наступит при большем

угле нагрузки θ . Установившийся режим с углом нагрузки θ_n наступит при номинальном токе возбуждения генератора. Угловая характеристика генератора при этом токе возбуждения примет вид, представленный на рис. 2. На рис. 3 приведены рабочие характеристики синхронного двигателя нагрузочного устройства, содержащего две синхронные машины серии СТД-12500 ($U_n = 10000$ В, $I_n = 819,8$ А, $P_n = 12500$ кВт, $n_n = 3000$ об/мин). Характеристики получены путем изменения тока возбуждения синхронного генератора в диапазоне его изменения от нуля до номинального значения.

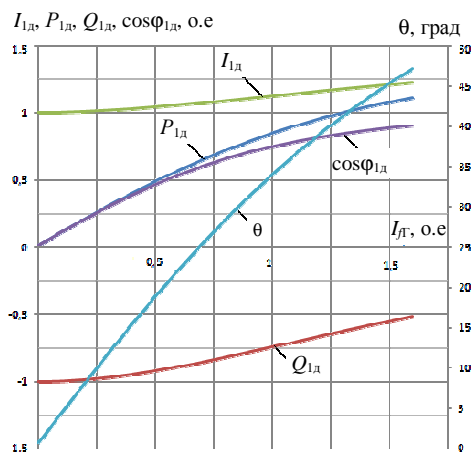


Рис. 3. Рабочие характеристики СД нагрузочного устройства

Анализ рабочих характеристик СД показывает следующее: активная мощность $P_{1д}$, потребляемая двигателем из сети, и ток $I_{1д}$ обмотки статора растут по величине; реактивная мощность $Q_{1д}$ двигателя является ёмкостной и уменьшается; коэффициент мощности $\cos\varphi_{1д}$ двигателя и угол нагрузки θ возрастают.

Библиографический список

1. Сипайлов Г.А., Лоос А.В. Математическое моделирование электрических машин. – М.: Высшая школа, 1980. – 176 с.
2. Трефилов В.А. Установившийся режим работы электромашиной системы с двухмашинным агрегатом // Электрические машины и электромашиные системы: сб. науч. тр. / Перм. гос. техн. ун-т. – Пермь, 2003. – С. 61–66.

АНАЛИЗ СОСТОЯНИЯ ОБЪЕКТА ПО РЕЗУЛЬТАТАМ МОНИТОРИНГА ЕГО ЭНЕРГЕТИЧЕСКИХ ПАРАМЕТРОВ

Студент В.Д. Кулик, канд. техн. наук, доцент Д.К. Елтышев
Пермский национальный исследовательский
политехнический университет

Рассмотрены вопросы выбора методов оценки состояния разнородных объектов на основе данных, полученных в процессе мониторинга их энергетических параметров. Разработана концептуальная схема оценки состояния объекта при управлении его энергоэффективностью. Приведен анализ подходов к определению норм энергопотребления объектов. Исследованы существующие алгоритмы кластеризации данных как один из способов выявления нормативных значений, отмечены их достоинства и недостатки.

Ключевые слова: энергопотребление, энергоэффективность, нормирование, кластеризация.

ANALYSIS OF THE OBJECT CONDITION BASED ON THE RESULTS OF ENERGY PARAMETERS MONITORING

Student V.D. Kulik,
Ph.D. in Technical Sciences, Associate Professor D.K. Eltyshv
Perm National Research Polytechnic University

This article considers the choice issues of methods for evaluation the heterogeneous objects condition on the basis of the data obtained while monitoring energy parameters. A conceptual evaluation scheme of the object condition while managing its energy efficiency has been developed. Analysis of approaches to the setting of energy consumption rates of objects has been offered. The available clustering algorithms as a way of identifying the rate values have been studied, and their advantages and disadvantages have been shown.

Keywords: energy consumption, energy efficiency, rate setting, clustering.

Введение. Решение вопросов эффективного функционирования объектов генерации, распределения и потребления энергоресурсов в настоящее время является одним из ключевых направлений практически в любой сфере антропогенной деятельности. Предполагается, что использование меньшего количества энергии для обеспечения того же уровня энергетического обеспечения зданий или технологических процессов на производстве может существенно изменить уровень переменных издержек, тем самым снизив себестоимость продукта и услуг, эксплуатационные затраты и др.

Чтобы оценить уровень энергоэффективности объекта, необходимо иметь представление о динамике изменения его энергетических параметров. Под энергетическими параметрами будем понимать параметры, характеризующие величину потребления различных энергетических ресурсов. К ним можно отнести абсолютный показатель энергопотребления, а также удельные или комплексные показатели энергопотребления (с учетом различных факторов, таких как затраты временных и трудовых ресурсов, мощность, нагрузка, производительность и др.).

Собрать необходимые для анализа данные позволяет организация системы мониторинга за объектом как базового инструмента для дальнейшей оценки и прогнозирования [6]. Однако сведения о потреблении энергоресурсов, полученные в результате мониторинга, сами по себе не позволяют сделать выводы об эффективности их использования, а также о качестве функционирования объекта и его отдельных систем, подсистем и элементов. Поэтому в целях принятия объективных решений, связанных с анализом причин отклонения в работе объекта и их устранением, требуется предварительная обработка данных с использованием соответствующего математического и алгоритмического обеспечения. Таким образом, целью работы являются исследование и выбор методов оценки состояния разнородных объектов (зданий, процессов, оборудования и т.д.) по результатам мониторинга их энергетических параметров для задач поддержки принятия решений при управлении энергоэффективностью.

Разработанная концептуальная схема (рис. 1) оценки состояния объекта при управления его энергоэффективностью предполагает необходимость создания базы данных мониторинга и наполнения его энергетических параметров (на примере контроля энергопотребления (ЭП)). В зависимости от постановки задачи и характера данных для их информационно-аналитической обработки могут быть использованы различные методы. В ходе анализа исследуются тренды изменения параметров, формируются и корректируются нормы энергопотребления, выполняется прогноз энергопотребления на выбранный интервал времени. При анализе состояния объекта, расчете норм энергопотребления и прочем используются данные о состоянии его элементов (оборудования), его заявленных характеристиках, режимах работы и внешней среде, что позволяет получить более объективную

оценку состояния относительно исследуемого периода и более точную информацию о его возможных изменениях в будущем. По результатам анализа и оценки состояния объекта формируются рекомендации, направленные на обеспечение энергетической эффективности, а также необходимая отчетная документация.

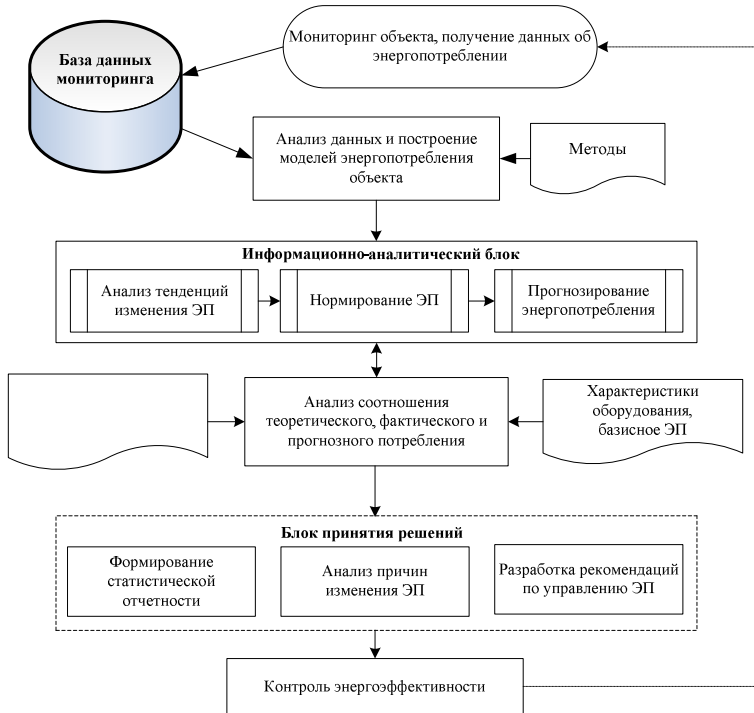


Рис. 1. Концептуальная схема процесса оценки состояния объекта на основе мониторинга энергетических параметров

Методы определения нормы энергопотребления при оценке состояния объекта. Одним из ключевых компонентов предлагаемой системы оценки является определение норм потребления энергоресурсов. Однако при определении нормы потребления электрической энергии следует учитывать, что различные объекты отличаются друг от друга типом электрической нагрузки и по-разному реагируют на изменяющиеся условия. Чтобы получить адекватные данные для анализа, необходимо организовать мониторинг не только электропотребления,

но и зачастую ряда внешних неэлектрических показателей. Кроме того, желательно организовать учет потребления различными типами потребителей. Поэтому для эффективного анализа важен не только выбор метода, но и определение факторов, оказывающих наибольшее влияние на величину энергопотребления.

По результатам анализа энергопотребления объекта с привязкой к внешним факторам возможно выработать рекомендации по его оптимизации в зависимости от различных условий работы.

Для определения нормы энергопотребления могут быть применены различные методы, в частности:

1) Опытный метод, основанный на использовании данных, полученных в результате испытаний при проектировании и разработке новых технологических установок и промышленных объектов, а также при модернизации уже имеющихся [5]. Недостатками данного метода являются жесткое соответствие технологического процесса режиму, предусмотренному технологическими регламентами и инструкциями, а также требования к технической исправности оборудования.

2) Отчетно-статистический метод, основанный на анализе данных статистической отчетности о фактическом расходе топливно-энергетических ресурсов за предыдущий период и их интерполяции на расчетный период [7], с использованием аналитических моделей и моделей базового периода [8].

3) Расчетно-аналитический метод, основанный на выполнении поэлементных расчетов по данным различной технической документации с учетом экспериментально установленных нормативных характеристик энергопотребляющих агрегатов и объектно-ориентированной математической модели [3].

4) Комбинированный метод, учитывающий связь энергопотребления со структурой и режимом работы производства и предполагающий использование комбинированной математической модели (сочетание объектно-ориентированной и аналитической моделей, связанных между собой через понятие энергетического профиля) [4].

5) Методы, основанные на кластерном анализе данных, решающие задачи разбиения множества объектов на группы. Внутри каждой группы должны оказаться «похожие» объекты, а объекты разных групп должны как можно больше отличаться [1].

Кластерный анализ данных об изменении энергетических параметров. Методы и алгоритмы, использующие ретроспективные данные об изменении ЭП для выявления общих тенденций работы объекта (табл. 1, 2), являются перспективным направлением исследований [2].

Таблица 1

Классификация методов и алгоритмов кластеризации [9]

№ п/п	Признак классификации	Название метода
1	Иерархические методы	– методы AGNES (агломеративные); – методы DIANA (дивизимные)
2	Неиерархические методы	– k-средних (k-means); – PAM (k-medoids); – CLOPE (кластеризация с градиентом)
3	По способу анализа данных	– четкие; – нечеткие
4	По количеству применений алгоритмов кластеризации	– с одноэтапной кластеризацией; – с многоэтапной кластеризацией
5	По возможности расширения объема обрабатываемых данных	– масштабируемые; – немасштабируемые
6	По времени выполнения кластеризации	– потоковые (on-line); – не потоковые (off-line)

Таблица 2

Сравнение методов кластеризации

№ п/п	Метод	Достоинства	Недостатки
1	CURE	Хорошее качество кластеризации, в том числе при наличии выбросов, с возможностью выделения кластеров сложной формы и различных размеров	Необходимо задавать пороговые значения и количество кластеров
2	BIRCH	Возможность кластеризации (двух-ступенчатой) больших объемов данных при небольших объемах используемой памяти	Хорошо выделяются только выпуклые или сферические кластеры. Необходимо задавать пороговые значения. Работа только с числовыми данными
3	MST	Возможность выделения кластеров произвольной формы с выбором из нескольких оптимальных решений	Чувствительность к выбросам

№ п/п	Метод	Достоинства	Недостатки
4	k-средних	Простота и понятность в использовании, высокая скорость работы	Высокая чувствительность к выбросам. Снижение быстродействия с ростом объема данных. Необходимость задания количества кластеров
5	PAM	Простота и понятность в использовании. Быстрота работы. Меньшая чувствительность к выбросам в сравнении с k-means	Необходимость задания количества кластеров. Низкая скорость на больших выборках данных
6	CLOPE	Высокие масштабируемость и скорость работы, качество кластеризации, а также легкость расчетов и интерпретации результата	–
7	Карты Кохонена	Использование нейронной сети с обучением без учителя и возможностью самоорганизации. Простота реализации	Работа только с числовыми данными, минимизация размеров сети. Необходимость задания количества кластеров
8	Алгоритм HCM	Легкость реализации, вычислительная простота	Отсутствие гарантии в нахождении оптимального решения. Необходимость задания количества кластеров
9	Fuzzy c-means	Возможность работы с граничащими кластерами благодаря нечеткой логике	Вычислительная сложность. Необходимость задания количества кластеров

В общем виде задача кластеризации состоит в разбиении набора данных об изменении энергетического параметра $X = \{x_1, \dots, x_n\}$ на подмножества (кластеры) $C = \{c_1, \dots, c_m\}$ из элементов, близких по значению заданной функции расстояния (метрике) D между ними, определяющего степень их сходства. При этом элементы из разных кластеров существенно отличаются. Число кластеров и вид функции D зависят от выбранного алгоритма кластеризации.

Из-за неоднозначности данных и довольно близких по величине числовых значений в работе предлагается использовать метод нечеткой кластеризации c-means. В будущем для повышения качества про-

водимого анализа стоит совместить кластерный анализ с отчетно-статистическим или комбинированным методом.

В алгоритме *c-means* кластеры описываются центрами c_j – векторами того же пространства, которому принадлежит исходное множество значений исследуемых энергетических параметров. В ходе решения задачи нечеткой кластеризации решается задача минимизации целевой функции:

$$\sum_{j=1}^m \sum_{i=1}^n U_{ij}^m \cdot \|x_i - c_j\|^2, \text{ при } \sum_{j=1}^m U_{ij} = 1, i = 1 : n,$$

где U – матрица принадлежности кластерам; x_i – значение параметра; m – коэффициент неопределённости (обычно равен 2); $C = \{c_j\}$ – центры кластеров, рассчитываемые по формуле:

$$c_j = \frac{\sum_{i=1}^n U_{ij}^m \cdot x_i}{\sum_{i=1}^n U_{ij}^m}.$$

Исследование алгоритма кластерного анализа. В результате мониторинга были получены данные об электропотреблении и температуре в учебном помещении за период с 11 февраля по 31 марта 2015 г., просуммированные по дням (рис. 2).



Рис. 2. График изменения электропотребления учебной лаборатории

На рис. 3 приведены результаты кластеризации полученных данных при четырех априори заданных кластерах. Пунктирные линии соответствуют значениям рассчитанных центров кластеров.

Анализ результатов позволяет сделать некоторые выводы. В частности, видно, что потребление электроэнергии в будние дни в основном приходится на значения до 2000 Вт. При этом есть дни с пониженным потреблением (1-й кластер, черный цвет) и средним потреблением (2-й кластер, синий цвет). Наиболее вероятными причинами повышенных (3-й кластер, розовый цвет) и «аномально» высоких (4-й кластер, красный цвет) значений электропотребления может являться повышенная производительность или оставленное включенным в ночное время оборудование.

Потребление в выходные дни в целом не превышает величины 500 Вт, кроме одного случая. Данный факт может быть с большой вероятностью связан с тем, что указанная суббота являлась рабочей.

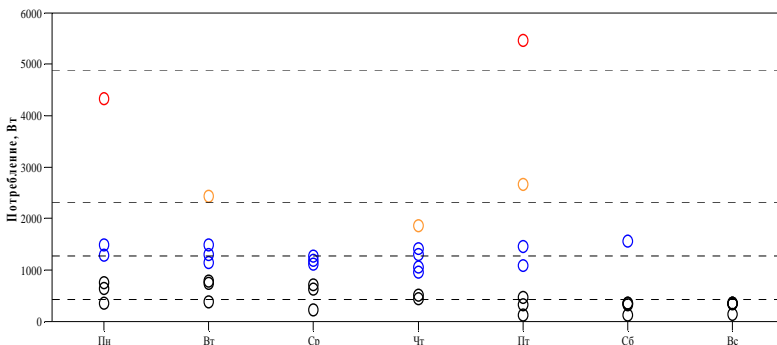


Рис. 3. Визуальное представление результатов кластеризации данных об электропотреблении

Выводы. В рамках работы предложена концептуальная схема процесса оценки состояния объекта на основе анализа значений его энергетических параметров, полученных в ходе мониторинга. В качестве способа реализации подсистемы нормирования параметров предложено использовать алгоритмы кластерного анализа. Практическое исследование алгоритмов кластеризации с использованием специально разрабатываемого программного обеспечения на примере учебной аудитории позволило выявить временные отрезки со схожими показателями электропотребления, возможные варианты формирования норм потребления по отдельным дням недели, а также вероятные причины отклонения в работе оборудования.

Библиографический список

1. Прикладная статистика: Классификация и снижение размерности / С.А. Айвазян, В.М. Бухштабер, И.С. Енюков, Л.Д. Мешалкин. – М.: Финансы и статистика, 1989. – 607 с.
2. Гнатюк В.И. Пределный алгоритм нормирования электропотребления объектов техноценоза. – М.: Энергобезопасность и энергосбережение, 2012. – С. 28–33.
3. Гофман И.В. Нормирование потребления энергии и энергетические балансы промышленных предприятий. – М.: Энергия, 1966.
4. Гринёв А.В. Комбинированный метод расчёта норм потребления топливно-энергетических ресурсов // Энергосбережение и водоподготовка. – 2011. – № 6.
5. Константинов Б.А. О применении математических методов при нормировании потребления электроэнергии в промышленности // Электричество. – 1964. – № 1. – С. 66.
6. Ратников А.О. Пути решения проблемы энергосбережения в России. – Казань: Изд-во «Познание» Ин-та экономики, управления и права, 2010. – 304 с.
7. Тайц А.А. Методика нормирования удельных расходов электроэнергии. – М.: Госэнергоиздат, 1946.
8. Чубукова И.А. Data Mining: учеб. пособие / Интернет-университет информационных технологий. – М.: БИНОМ: Лаборатория знаний, 2006. – 382 с. (Сер. Основы информационных технологий).
9. Ястребов П.П. Использование и нормирование электрической энергии в процессах, переработки и хранения. – М.: Колос, 1973. – 311 с.

РАЗРАБОТКА СТРУКТУРЫ СИСТЕМЫ ЭНЕРГОМОНИТОРИНГА АКТИВНО-АДАПТИВНОЙ СЕТИ НА ОСНОВЕ WAMS-ТЕХНОЛОГИИ

**Магистрант Г.П. Микрюков,
канд. техн. наук, доцент А.В. Кычкин**
Пермский национальный исследовательский
политехнический университет

В статье рассмотрены принципы разработки распределенной системы энергомониторинга активно-адаптивной сети по технологии WAMS. Система предназначена для сбора синхронизированных по времени данных, организации их долгосрочного хранения, визуализации и последующего автоматического анализа. Рассматриваются технология WAMS, и объясняется высокая точность синхронизации данных, достигаемая при ее применении. В работе формулируются основные задачи, выполняемые системой, а также описываются технические средства, используемые при ее реализации.

Работа выполнена в рамках гранта Президента Российской Федерации по государственной поддержке молодых российских ученых – кандидатов наук, МК–5279.2014.8 «Синтез эффективных технологий удаленного мониторинга и управления состоянием интеллектуальной электроэнергетической системы с активно-адаптивной сетью».

Ключевые слова: AAC, синхронизированные измерения, WAMS, энергетический мониторинг, GPS, распределенная система мониторинга, MicroGrid.

DESIGN OF SMARTGRID ENERGY MONITORING SYSTEM STRUCTURE BASED ON WAMS-TECHNOLOGY

**Master Student G.P. Mikriukov,
Ph.D. in Technical Sciences, Associate Professor A.V. Kychkin**
Perm National Research Polytechnic University

Main principles of SmartGrid energy monitoring system structure based on WAMS are considered in this article. The system is designed for long-term storing and automated analysis of collected time-synchronized data. Explained how does high accuracy of data synchronization is achieved due to use of WAMS technology. The paper formulates main tasks carried out by energy monitoring system and describes hardware used for its implementation.

Research has been completed under the grant of the President of Russian Federation for the government support of young scientists, МК–5279.2014.8 «Design of the efficient technologies for remote monitoring and state management of the smart power system with Smartgrid».

Keywords: SmartGrid, synchronized measurements, WAMS, energy monitoring, GPS, distributed monitoring system, Microgrid.

Одной из основных тенденций развития мировой энергетики является ввод в эксплуатацию распределенных систем энергомониторинга (РСЭМ). Внедрение РСЭМ в интеллектуальных активно-адаптивных сетях (ААС) – SmartGrid обусловлено необходимостью повышения эффективности перераспределения потоков электрической энергии от производителя к потребителю, выявления нерационального использования энергии, прогноза показателей потребления, внедрения тарифной политики, формирования оптимальных режимов работы сети и др. Данные эффекты достигаются за счет автоматизации технологических процессов получения и передачи данных о текущих состояниях энергетических объектов ААС, а также процессов расчета параметров энергоэффективности. Эти процессы сонаправлены с интеллектуализацией процессов учета, управления и повышением безопасности персонала, обслуживающего объекты сети.

Однако практика сбора энергетических данных с территориально распределенных сетевых объектов показывает низкую совместимость технологий, форматов, интерфейсов и протоколов обмена данными, программно-аппаратных комплексов учета, вызванную одновременной заменой измерительных средств, различием финансовых условий и политики модернизации сетей в регионах, климатических факторов и др. Это усложняет формирование единой картины энергетического состояния ААС на больших территориях в режимах реального времени.

В связи с этим проектирование и внедрение РСЭМ реального времени для ААС являются важным направлением исследований. Оно включает в себя анализ процессов сбора, долгосрочного хранения получаемой информации о ключевых объектах сети, а также ее аналитической обработки, с учетом временной синхронизации измерений на узлах по технологии WAMS. Внедрение энергетического мониторинга ААС с WAMS предоставляет синхронизацию измерений в произвольном числе точек наблюдения за счет назначения в средствах измерений временных меток при помощи модулей глобальной навигационной системы GPS [2, 3].

Распределенный мониторинг ААС на основе WAMS-технологий. Построение системы распределенного энергомониторинга на основе WAMS-технологии можно разделить на несколько этапов:

- проведение исследования текущего состояния активно-адаптивной сети (ААС) и оборудования, находящегося в ней;
- выявление точек и параметров мониторинга ААС;
- выбор технического и программного комплекса для реализации РСЭМ;
- разработка оптимальной архитектуры РСЭМ для поставленной задачи.

Основные задачи, выполняемые автоматизированной РСЭМ:

- сбор данных о потреблении энергии объектами мониторинга и их синхронизация по времени;
- преобразование данных в форматы для долгосрочного хранения в БД, а также передача данных в систему управления;
- визуализация данных в удобном для восприятия виде, таком как графики и тренды;
- генерация сообщений об аварийных событиях;
- аналитическая работа с хранимой информацией: прогнозирование потребления, составление математических моделей потребителей и пр.

Мониторинг и анализ данных основываются на полученной информации с объектов ААС как в режимах «жесткого», так и «мягкого» реального времени. Технология WAMS обеспечивает систему мониторинга синхронизированными во времени измерениями параметров мониторинга. Это позволяет строить более точные модели ААС и детальнее изучать переходные процессы [2, 3]. Структура РСЭМ на основе WAMS показана на рис. 1.

Для измерений используются блоки фазовых измерений (PMU, Phasor Measurement Unit), они подключаются непосредственно к ЭЭС и собирают необходимые параметры мониторинга. Синхронизация по времени с точностью до 2 мс обеспечивается за счет встроенных в PMU GPS-часов. Фазовые концентраторы данных (PDC, Phasor Data Concentrator) собирают данные с подключенных к ним PMU (и других PDC) и формируют единый поток синхронизированных по времени данных, которые используются системами мониторинга и управления ЭЭС [1].

Синхронизированные данные хранятся долгосрочно как в PCD, так и в верхних уровнях системы. Потоки данных изображены на рис. 2.

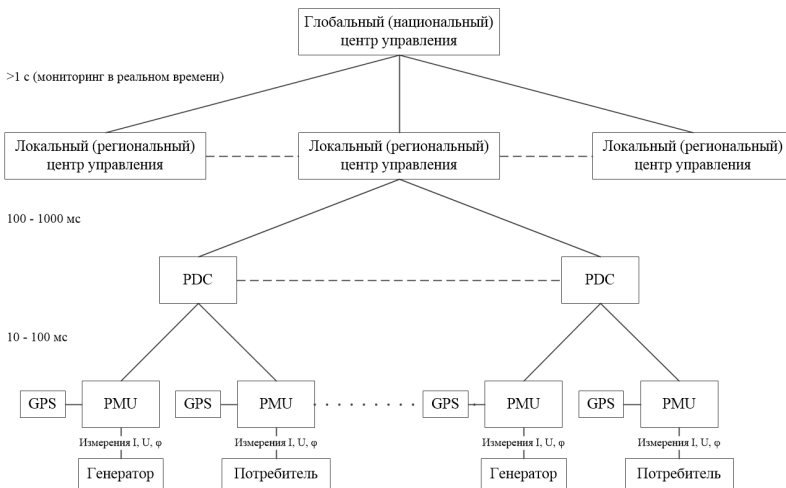


Рис. 1. PCЭМ на основе WAMS

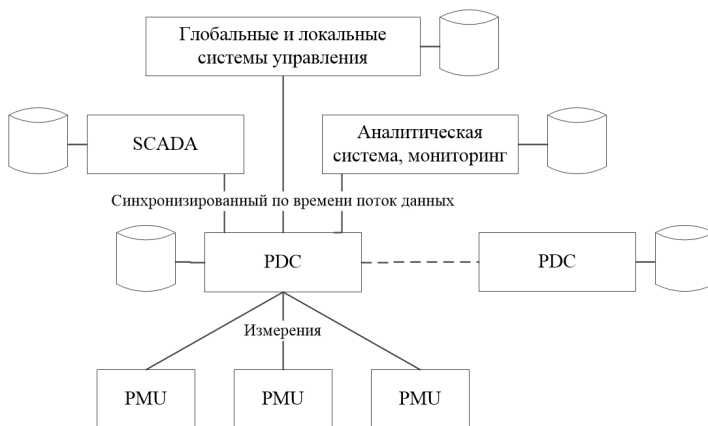


Рис. 2. Данные в PCЭМ на основе WAMS

Построение системы мониторинга для HIL MicroGrid. Разработка распределенной системы энергомониторинга проводится на основе полунатурной модели ААС, в ее основе лежат следующие принципы:

- принцип аппаратного полунатурного моделирования (Hardware-in-the-loop, HIL);

– технология ПЛИС (программируемая логическая интегральная схема) для реализации математических моделей ввода и вывода сигналов в реальном времени;

– вычисления на базе контроллера PXI с ОС реального времени LabView Real-Time, которая широко используется при разработке ААС.

Полунатурное моделирование ААС имитирует работу потребителей и генераторов энергии сети электроснабжения. Это обеспечивается за счет аппаратного ввода-вывода переменных тока и напряжения, которые масштабируются для условий лаборатории с границами ± 10 В. Структурная схема данной системы представлена на рис. 3.

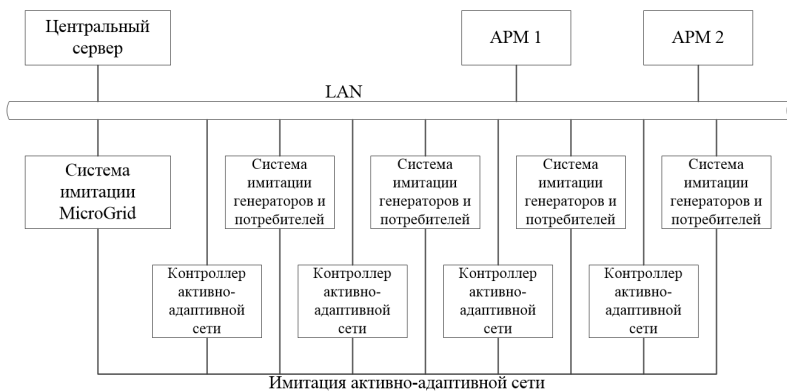


Рис. 3. Структурная схема полунатурной модели ААС

Как видно из схемы, все элементы модели подключаются к локальной сети. Системы имитации генераторов и потребителей аппаратно реализуют математические модели различных видов генераторов и потребителей электроэнергии.

Контроллеры активно-адаптивной сети отвечают за удаленный сбор данных и управление работой генераторов и потребителей. Система имитации MicroGrid отвечает за аппаратную имитацию ААС.

Структура РСЭМ активно-адаптивной сети для полунатурной модели ААС представлена на рис. 4.

На данной схеме приняты следующие обозначения: пунктирные линии – линии передачи информации; сплошные – линии передачи электроэнергии; G_i – генераторы электроэнергии; C_i – потре-

бители электроэнергии; RTC_i – удаленные устройства управления; P_i – защитная автоматика, коммутирующая автоматика; ЕЕИС – внешняя энергосеть (линии передачи энергии и информации); PMU – блоки фазовых измерений; PDC – фазовый концентратор данных; СЭМ – система энергомониторинга (информационная аналитическая система).

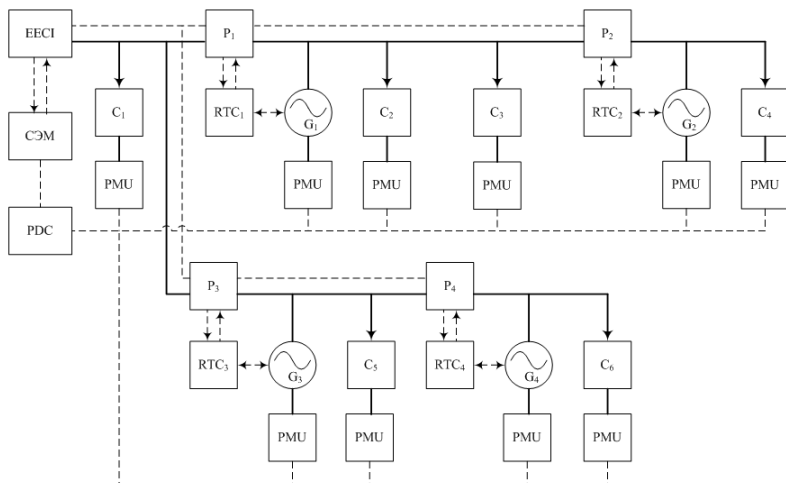


Рис. 4. Структура РСЭМ для полунатурной модели ААС

Данные об энергетическом состоянии объектов ААС поступают с PMU в СЭМ через PDC, где происходит формирование единого синхронизированного по времени информационного потока. СЭМ осуществляет обработку и анализ полученных данных и передает полученную статистику и информацию в диспетчерские центры и центры управления, где по ним могут вноситься корректировки и изменения в режимы генерации и потребления, это возможно как в ручном, так и в автоматическом режиме.

Заключение. Распределенные системы энергомониторинга, разработанные на основе технологии WAMS, обладают важным преимуществом: они гарантируют синхронизацию измерений по времени с точностью порядка 10 мс. Это позволяет более детально изучать переходные режимы в энергосетях и составлять более точные модели реальных энергосистем. Такие РСЭМ рекомендуются к использованию

в активно-адаптивных сетях, это обуславливается более простой операцией внедрения таких систем и большим экономическим эффектом от внедрения.

Библиографический список

1. C37.118-2005 – IEEE Standard for Synchrophasors for Power Systems.

2. Верификация динамической модели ЕЭС/ОЭС по данным систем мониторинга переходных режимов / А.С. Герасимов, А.Х. Есипович, С.Е. Кузьмин, Ю.А. Куликов, А.Н. Смирнов, А.И. Суковицин // Современные системы возбуждения вращающихся электрических машин и устойчивость электроэнергетических систем: материалы междунар. науч.-практ. конф. / ПЭИПК. – СПб., 2008. – С. 60–66.

3. Оценка достоверности динамических моделей сложных электроэнергетических систем по данным СМРР / А.В. Жуков, Ю.А. Куликов, А.С. Герасимов, А.Х. Есипович, А.Н. Смирнов // Оперативное управление в электроэнергетике. – 2011. – № 6.

АНАЛИТИЧЕСКИЙ ОБЗОР ПОДХОДОВ К ОЦЕНКЕ ПРИОРИТЕТОВ ПРИ ЭКСПЛУАТАЦИИ ЭЛЕКТРОЭНЕРГЕТИЧЕСКИХ ОБЪЕКТОВ

Студент С.А. Мошкин,
канд. техн. наук, доцент Д.К. Елтышев
Пермский национальный исследовательский
политехнический университет

Рассматриваются существующие методы, используемые для оценки приоритетов при выборе электроэнергетических объектов и проектов, связанных с их техническим перевооружением.

Ключевые слова: ранжирование, выбор альтернатив, оценка приоритетов, критерии, электроэнергетические объекты, методы оптимизации.

ANALYTICAL REVIEW OF APPROACHES TO ASSESSING THE PRIORITIES IN THE OPERATION OF ELECTRIC POWER FACILITIES

Student S.A. Moshkin,
Ph.D. in Technical Sciences, Associate Professor D.K. Eltyshv
Perm National Research Polytechnic University

This article discusses the existing methods which are used for assessing the priorities when choosing electric power facilities and projects related to their technical modernization.

Keywords: ranking, selection of alternatives, evaluation of priorities, criteria, electric power facilities, optimization methods.

Введение. В современном мире электроэнергетика является основой развития базовых отраслей промышленности, определяющих прогресс общественного производства [1]. Чтобы обеспечить промышленность и население столь важными ресурсами, как электрическая и тепловая энергия, необходимо создать и поддерживать функционирование специальных электроэнергетических объектов, состоящих из электрических станций, подстанций, линий электропередачи, трубопроводов и т.д., которые необходимы для создания, переработки и передачи данных видов энергии.

При эксплуатации электроэнергетических объектов (ЭЭО) может возникнуть задача, связанная с выполнением работ по их модернизации, реконструкции, техническому обслуживанию, ремонту. Как

правило, средства и ресурсы на реализацию подобных мероприятий ограничены. При планировании и реализации проектов, связанных с указанными выше задачами, необходимо определять, какие объекты являются наиболее приоритетными. При этом нужно учитывать не только базовые экономические критерии, но и специфические критерии, связанные с эксплуатацией ЭЭО, такие как перечень и категория потребителей, ущерб от сбоя в работе оборудования, удаленность объекта от центра, где располагается обслуживающий персонал, и т.д. Также важно принимать во внимание критерии, связанные с состоянием рассматриваемых объектов (например, срок службы оборудования, вероятность отказа и т.д.). Кроме того, всегда имеется определенный риск вкладывать средства в тот или иной объект, связанный с различными факторами [2]. Одной из трудностей принятия решений при оценке приоритета реализации проектов является то, что рассматриваемая задача является сложноформализуемой и происходит в условиях неопределенности ввиду того, что результаты принимаемых решений заранее не известны.

Неопределенность и многокритериальность накладывают определенные ограничения на выработку решений, связанных с оценкой приоритета проектов для отдельных ЭЭО [3]. Поэтому одним из важных моментов при решении подобного рода задач является выбор эффективного метода, позволяющего объективно оценить приоритет различных объектов с позиций проведения работ по их перевооружению с учетом риска и других критериев в условиях неопределенности исходной информации.

Постановка задачи оценки приоритета. В общем случае задачу оценки приоритета ЭЭО можно представить в виде набора следующих элементов:

$$\langle F, A, K, Y, M, W \rangle,$$

где A – множество возможных альтернатив (ЭЭО, по отношению к которым возможна реализация проекта), $A = \{a_1, a_2, \dots, a_n\}$; F – цель (ранжирование альтернатив по величине приоритета); K – множество критериев оценки проектов, $K = \{K_1, K_2, \dots, K_m\}$; W – относительные веса критериев, $W = \{w_1, w_2, \dots, w_m\}$; M – решающие правила выбора альтернатив (методы многокритериальной оптимизации), $M = \{m_1, m_2, \dots, m_r\}$.

Анализ методов решения задачи многокритериальной оценки приоритетов. Рассмотрим некоторые методы, связанные с решением задач многокритериальной оптимизации.

1. Метод анализа иерархий Т. Саати [4].

Применение метода анализа иерархий к решению многокритериальных задач основано на их скаляризации при помощи линейной свертки критериев. А именно согласно данному методу «наилучшим» решением многокритериальной задачи объявляется то, которое составляет наибольшее возможное значение линейной (аддитивной) свертки критериев $\sum_{i=1}^m w_i K_i(a)$, т.е. такое решение $a^* \in A$, для которого выполняется равенство

$$a^* = \max_{a \in A} \sum_{i=1}^m w_i \cdot K_i(a_j), j = 1:n, \quad (1)$$

где $K_i(a_j)$ – оценка j -й альтернативы по i -му критерию.

Эксперту для сравнения по «весу» или «важности» предлагается набор критериев K_1, K_2, \dots, K_m , которые выступают в качестве сравниваемых объектов. Формируется матрица парных сравнений, а искомым весовой вектор $w = (w_1, w_2, \dots, w_m)^T$ вычисляется как собственный вектор этой матрицы, отвечающий максимальному собственному значению. Получив в распоряжение конкретные положительные значения w_1, w_2, \dots, w_m коэффициентов свертки $\sum_{i=1}^m w_i \cdot K_i(a_j)$, можно приступить к ее максимизации на множестве возможных решений A и тем самым нахождению «наилучшего» (в соответствии с методом анализа иерархий) решения a^* .

2. Метод, основанный на использовании критерия Гурвица.

Применительно к рассматриваемой проблеме решения задач многокритериальной оптимизации (при минимизации частных критериев) используют следующие оценки:

– самая худшая / наибольшая из всех оценок заданных частных критериев;

– самая лучшая / наименьшая из всех оценок заданных частных критериев.

При этом результирующий показатель для выбора по методу Гурвица формируется как взвешенный показатель на основе синтеза указанных выше двух крайних типов оценок.

При таком подходе критериальная функция для критерия выбора имеет вид:

$$\alpha \cdot \max(w_i \cdot K_i(a_j)) + (1 - \alpha) \cdot \min(w_i \cdot K_i(a_j)), j = 1:n, \quad (2)$$

где максимум и минимум ищутся по всем частным критериям, а $\alpha \in [0;1]$ – параметр, выбирающийся для адаптации линий уровня критерия к предпочтениям ЛПР. Соответственно, точка максимума этой функции принимается как искомое оптимальное решение [5].

3. Метод, основанный на использовании критерия Вальда.

В формате такого подхода применительно к задачам многокритериальной оптимизации выбираются управляющие параметры (ими могут являться и коэффициенты значимости критериев), которые задает ЛПР с учетом имеющейся системы предпочтений в пространстве значений частных критериев и специфики многокритериальной задачи оптимизации.

При таком подходе на основе заданных частных критериев в рамках исходной многокритериальной задачи формируется критериальная функция выбора следующим образом:

$$\min(\alpha_i(w_i \cdot K_i(a_j))). \quad (3)$$

Здесь максимум ищется по всем доступным альтернативным решениям, а $\alpha_i > 0$ – параметры, которые задает ЛПР. Соответственно, точка максимума этой синтезированной критериальной функции выбора принимается как искомое оптимальное решение. Если найденное значение достигается при двух или более альтернативных решениях, то требуется проверка выбираемого решения на оптимальность по Парето.

В формате указанного подхода, но применительно к задачам оптимизации решений в условиях неопределенности на параметры α_i дополнительно накладывається условие $\sum \alpha_i = 1$ (аналогия вероятности случайных событий, влияющих на конечный экономический результат). Более того, соответствующие параметры ЛПР может задавать и в более удобном для себя виде, например, в виде некоторых пропорций, характеризующих его отношение к оценкам частных критериев [6].

4. Метод, основанный на использовании критерия Сэвиджа.

В соответствии с этим методом, если требуется в любых условиях избежать большего риска, то оптимальным будет та альтернатива A , для которой риск, максимальный при различных вариантах условий, окажется минимальным.

Использование данного критерия обеспечивает наименьшее значение максимальной величины риска [7]:

$$\min_j \max_i(w_i K_i(a_j)). \quad (4)$$

5. Метод, основанный на теории нечеткой логики Л. Заде.

Нечеткая логика используется в тех случаях, когда необходимо принять решение в условиях неопределенности информации и строится на теории нечетких множеств, где функция принадлежности элемента множеству может быть любой на отрезке $[0; 1]$. Если функция принимает крайнее значение, равное 0, то тогда она не принадлежит множеству, а если равно 1, то принадлежит соответственно. Задача многокритериального анализа состоит в упорядочивании элементов множества A по критериям K , имеющим весовые оценки W .

Пусть $b_{i,j} = w_i \cdot K_i(a_j)$ – число в диапазоне $[0, 1]$, которое характеризует уровень оценки альтернативы $a_j \in A$ по критерию K_i с учетом веса w_i : чем больше число, тем выше оценка альтернативы по критерию, $i=1:m, j=1:n$. Тогда набор оценок b_i можно представить в виде нечеткого множества \bar{b}_i на универсальном множестве альтернатив A :

$$\bar{b}_i = \left\{ \frac{\mu_{b_i}(a_1)}{a_1}, \frac{\mu_{b_i}(a_2)}{a_2} \dots \frac{\mu_{b_i}(a_n)}{a_n} \right\}, \quad (5)$$

где $\mu_{b_i}(a_j)$ – степень принадлежности элемента a_j нечеткому множеству b_i .

При использовании этого метода одним из способов решения является формирование матриц парных сравнений вариантов по каждому критерию. Общее количество таких матриц совпадает с количеством критериев и равняется m . Другим вариантом является использование функций принадлежности и алгоритмов нечеткого логического вывода [8].

Наилучшей альтернативой будет та, которая одновременно является лучшей по всем критериям. Нечеткое решение находится как пересечение частных критериев:

$$\begin{aligned} & \bar{b}_1 \cap \bar{b}_2 \cap \dots \cap \bar{b}_m = \\ & = \left\{ \frac{\min_{i=1,m} \mu_{b_i}(a_1)}{a_1}, \frac{\min_{i=1,m} \mu_{b_i}(a_2)}{a_2}, \dots, \frac{\min_{i=1,m} \mu_{b_i}(a_n)}{a_n} \right\}. \end{aligned} \quad (6)$$

Согласно полученному нечеткому множеству наилучшей альтернативой следует считать то, для которого степень принадлежности является наибольшей.

Заключение. В ходе работы рассмотрены методы, применяемые для решения задач многокритериальной оптимизации при выборе приоритетов реализации проектов в сфере электроэнергетики. Основным недостатком большинства рассмотренных методов является отсутствие возможности работы с различными видами неопределенности при описании задач принятия решений. На практике могут возникнуть ситуации, когда информация о задаче практически отсутствует, задана качественно, либо является неполной. Для некоторых критериев и их значений могут быть определены не их точные описания, а лишь множества, к которым эти описания принадлежат и т.д. В случае с ЭЭО есть большая вероятность неоднозначного определения как системы критериев, по которым будет производится ранжирование, так и степеней предпочтения одних критериев над другими. С учетом данных особенностей задачи оценки приоритета наиболее удобным подходом к оценке приоритета является использование теории нечеткой логики Л. Заде. Данный подход позволяет учитывать не только количественные, но и качественные показатели, не имеющие точной числовой оценки, и достаточно легко описывать сложные, не поддающиеся математическому анализу, задачи оценки приоритета.

Библиографический список

1. Экологические проблемы энергетического обеспечения человечества [Электронный ресурс]. – URL: <http://nuclphys.sinp.msu.ru/ecology/ecol/ecol05.htm> (дата обращения: 27.11.2015).
2. Назарычев А.Н. Методы и модели оптимизации ремонта электрооборудования объектов энергетики с учетом технического состояния. – Иваново: Изд-во Иван. гос. энерг. ун-та, 2002. – 168 с.
3. Проблема принятия решений [Электронный ресурс]. – URL: <http://www.studfiles.ru/preview/2419995> (дата обращения: 27.11.2015).
4. Ногин В.Д. Принятие решений в многокритериальной среде: количественный подход. – М.: Физматлит, 2002. – 165 с.
5. Бородецкий Г.Л. Методы оптимизации многокритериальных решений в логистике. – М.: Эксмо, 2009. – 230 с.
6. Жуковин В.Е. Многокритериальные модели принятия решений с неопределенностью. – Тбилиси: Мецниереба, 1983. – 195 с.
7. Кини Р.Л., Райфа Х. Принятие решений при многих критериях: предпочтения и замещения. – М.: Радио и связь, 1981. – 244 с.
8. Борисов В.В., Круглов В.В., Федулов А.С. Нечеткие модели и сети. – М.: Горячая линия-Телеком, 2012. – 284 с.

ПОСТРОЕНИЕ МНОГОШИННОЙ СЕТЕВОЙ АРХИТЕКТУРЫ СИСТЕМЫ СБОРА ЭНЕРГЕТИЧЕСКИХ ДАННЫХ

Студент В.В. Носков,
канд. техн. наук, доцент А.В. Кычкин
Пермский национальный исследовательский
политехнический университет
Технический специалист А.А. Яблоков
Энвидатек ГмбХ, г. Гамбург

В статье рассмотрены принципы построения многоуровневой сетевой архитектуры системы сбора и передачи энергетических данных на основе устройства удаленной телеметрии и управления VIDA350 компании Envidatec GmbH, поддерживающей различные протоколы и стандарты передачи измерительной информации по промышленным шинам связи. Энергетические данные включают в себя данные по потреблению электрической, тепловой энергии, воды, пара, газа и других ресурсов с приборов учета, счетчиков или датчиков расхода с импульсным или цифровым выходом. Кроме этого совместно с энергетическими данными производится сбор параметров температуры, давления, состояния объекта наблюдения.

Ключевые слова: многоуровневая сетевая архитектура, промышленная шина, энергетические данные, устройство удаленной телеметрии и управления, VIDA350.

DEVELOPMENT THE MULTYBUS NETWORK ARCHITECTURE OF ENERGY DATA COLLECTION SYSTEM

Student V.V. Noskov,
Ph.D. in Technical Sciences, Associate Professor A.V. Kychkin
Perm National Research Polytechnic University
Software Engineer A.A. Yablokov
Envidatec GmbH, Hamburg

The article describes the principles of multi-level network architecture of the collection and transmission of energy data based on the device remote telemetry and control VIDA350 company Envidatec GmbH, supporting various protocols and standards for signal transmission for industrial communication buses. Energy data includes data on consumption of electricity, heat, water, steam, gas and other resources meters, meters or flow sensors with pulse or digital output. In addition, together with the energy data is collected parameters of temperature, pressure, state of the object.

Keywords: multylevel network architecture, fieldbus, energy data, remote telemetry and controlunit, VIDA350.

Введение. Сегодня многие промышленные предприятия активно проводят мероприятия по энергосбережению и повышению эффективности использования ресурсов. Важным инструментом энергоресурсосбережения являются системы учета, архитектура которых, несмотря на широкое распространение и глубокую проработку технических решений, за последние несколько лет практически не изменилась. В её состав входят: нижний уровень, представленный первичными измерительными приборами, средний, включающий устройства сбора и передачи данных, и верхний, отвечающий за хранение, обработку и анализ результатов измерений.

Наблюдаемое развитие средств измерений, оснащенных цифровыми выходами, поддерживающими различные протоколы передачи данных, часто приводит к изменению традиционных архитектур, исключая или добавляя новые вспомогательные уровни. Новые возможности многошинной организации системы сбора энергоданных предоставляет устройство удаленной телеметрии и управления VIDA350 компании Envidatec GmbH, выполняющее сбор, хранение и передачу данных на верхние уровни системы. Многошинная организация сетевой архитектуры системы сбора энергетических данных на основе устройства VIDA350, являющегося программируемым логическим контроллером (ПЛК), во многом достигается его аппаратными особенностями и возможностью программной реализации сетевых протоколов (стандартных и пользовательских).

Принцип построения многошинной архитектуры. Сетевая архитектура представляет набор топологий и технологий функционирования сети. Выбор сетевой архитектуры во многом влияет на качества данных, а также на загруженность аппаратуры.

Для построения эффективной сетевой архитектуры сбора и передачи данных на основе оборудования VIDA350 можно использовать различные типовые топологии. Каждая топологии имеет свои особенности построения, но общим является наличие ПЛК, который отвечает за сбор и передачу данных.

На рис. 1 представлена архитектура, на которой два различных устройства VIDA350 снимают показания с датчиков по шине 1-wire и S0, передают их по каналу GSM/GPRS на общую шину L2, которая связана с Интернетом.

Шина 1-Wire предоставляет различные виды проводки, которые соединяются в топологии «дерево», «звезда» или «цепь». Шина пере-

даци импульсов S0 определяется стандартом DIN 43864, применяется для передачи сигналов со счетчиков расхода электричества, воды, тепла, газа и других ресурсов.

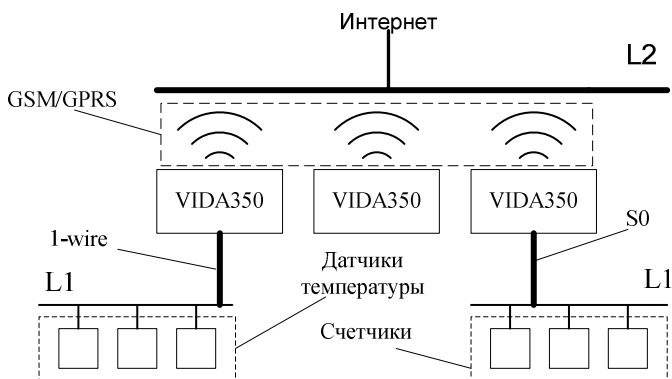


Рис. 1. Двухуровневая архитектура сбора данных с шиной 1-wire

VIDA350 имеет возможность подключения разных промышленных шин. Данная возможность позволяет использовать устройство в качестве роутера. На рис. 2 показана архитектура, где к VIDA350 подключено несколько датчиков по различным шинам. В табл. 1 показаны стандартные каналы связи и интерфейсы VIDA350.

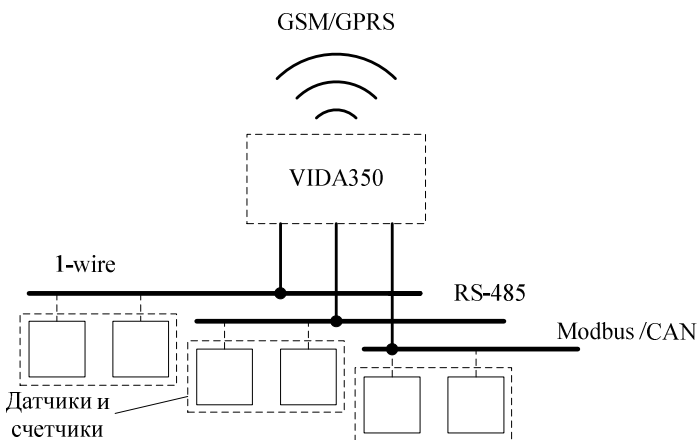
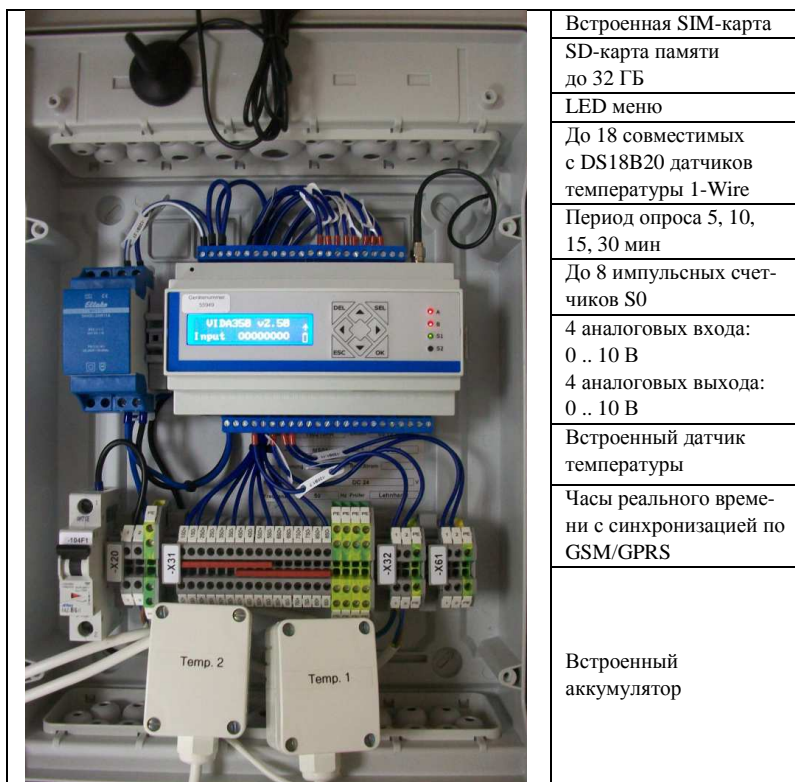


Рис. 2. Двухуровневая архитектура сбора данных с нескольких шин

Стандартные каналы связи и интерфейсы VIDA350

№ п/п	Интерфейс	Протокол
1	S0	–
2	1-wire	–
3	RS-232/RS-485	Modbus RTU
4	CANbus	CANopen
5	Lon	LonWorks
6	USB	USB 2.0
7	Ethernet	TCP/IP
8	GSM / GPRS	TCP/IP

Общий вид устройства в сборке, используемой для подключения различных шин, и основные технические характеристики представлены на рис. 3.



Встроенная SIM-карта
SD-карта памяти до 32 ГБ
LED меню
До 18 совместимых с DS18B20 датчиков температуры 1-Wire
Период опроса 5, 10, 15, 30 мин
До 8 импульсных счетчиков S0
4 аналоговых входа: 0 .. 10 В
4 аналоговых выхода: 0 .. 10 В
Встроенный датчик температуры
Часы реального времени с синхронизацией по GSM/GPRS
Встроенный аккумулятор

Рис. 3. Технические характеристики VIDA350

Структурная организация системы сбора и передачи данных с использованием VIDA350 и типовых сетевых технологий. На рис. 4 приведена общая схема архитектуры системы сбора и передачи энергетических данных. Отличительной особенностью системы являются зашифрованный канал связи, а также двунаправленная связь между шлюзом данных и клиентской сетью.

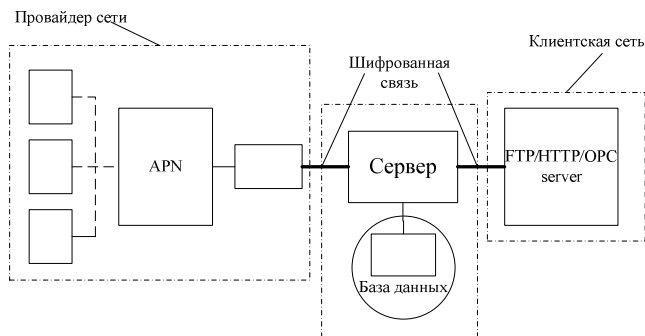


Рис. 4. Сетевая архитектура системы сбора и передачи энергетических данных

Для создания канала передачи данных между провайдером и сервером необходимы могут использоваться настройки, приведенные в табл. 2.

Таблица 2

Параметры настройки сети

Сервер	Параметр	Провайдер
Настройка сети		
IP-адрес 1	IP-адреса с равным уровнем доступа	IP-адрес 2
3DES	Алгоритм шифрования сообщений	3DES
MD5	Целостность сообщений	MD5
Предварительный ключ	Метод проверки подлинности	Предварительный ключ
Group 2	DH –группа	Group 2
3.600	Время жизни (с)	–
IPSec Параметры		
ESP	Механизм шифрования	ESP
3DES	ESP Transform	3DES
MD5	Целостность данных	MD5
84.600 с	Ассоциация безопасности (SA), срок службы	84.600 с

На рис. 5 показана архитектура надежного серверного взаимодействия в системе сбора и передачи энергетических данных. Сеть построена на базе синхронизированной системы серверов Master/Slave (Ведущий/Ведомый). Ведущими являются серверы, с которыми происходит взаимодействие, а ведомым – резервируемый сервер. При работе ведущих серверов всю работу берут ведомые серверы (пунктирной линией показаны данные, направляемые с резервного сервера).

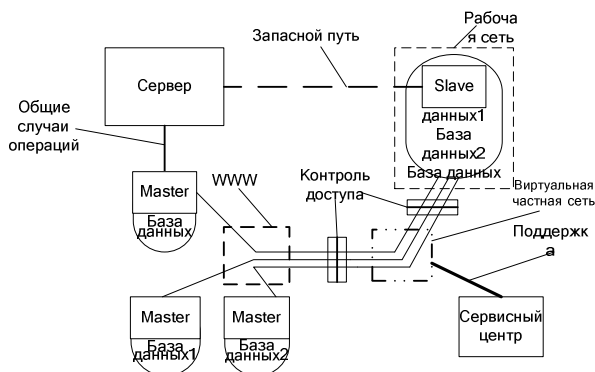


Рис. 5. Серверное взаимодействие в системе сбора и передачи энергетических данных

Заключение. Таким образом, функциональные особенности устройства удаленной телеметрии и управления VIDA350 позволяют реализовать на его основе многошинную сетевую архитектуру сбора данных, используя типовые стандарты обмена информацией в промышленных сетях. Показана наиболее эффективная многоуровневая реализация архитектуры сбора и передачи энергетических данных на базе VIDA350. Приведены основные его характеристики и настройки. Использование предложенной архитектуры во многом упрощает сбор данных и повышает эффективность организации многошинных систем учета энергетических ресурсов.

Библиографический список

1. Забара В.И. Энергосбережение и энергоучёт в электрических сетях // Промышленная автоматизация в России. – URL: <http://www.industrialauto.ru> (дата обращения: 22.11.15).
2. Документация компании «ENVIDATEC» GmbH. – URL: envidatec.com.

ЭНЕРГОСБЕРЕГАЮЩАЯ ЛОГИКА

Аспирант А.С. Прохоров

Пермский национальный исследовательский
политехнический университет

В статье рассмотрена концепция энергосберегающей логики программируемых логических интегральных микросхем.

Ключевые слова: энергосбережение, энергоэффективность, ПЛИС, «зелёная» логика.

ENERGY-SAVING LOGIC

Graduate Student A.S. Prokhorov

Perm National Research Polytechnic University

In this article discusses the concept of energy-saving logic programmable logic integrated circuits.

Keywords: energy saving, energy efficiency, FPGA, “green” logic.

Существующая логика программируемых логических интегральных схем (ПЛИС) в настоящее время по большей части распределена в пространстве – по кристаллу кремния в виде конфигурируемых логических блоков (КЛБ), состоящих из логических элементов (ЛЭ), программируемых локальных и глобальных матриц соединений. Хотя эта логика и занимает относительно небольшой процент площади кристалла по сравнению с памятью, но от неё во многом зависит энергоэффективность всей ПЛИС. В настоящее время вычисление логических (булевых) функций и автоматных отображений в ЛЭ ПЛИС осуществляется аппаратно, что не позволяет в полной мере реализовать энергосберегающие технологии в ПЛИС.

Предлагаются следующие положения концепции «зелёной» логики.

Первое из положений – это распределение логики из пространственной во временную область, что влечет за собой снижение энергопотребления за счёт отключения части аппаратуры, если такое отключение возможно¹. Для этого необходимо многократное использование одного и того же КЛБ в пределах одной загрузки файла конфигурации.

¹ Тюрин С.Ф., Харченко В.С. «Зелёная» программируемая логика: концепция и элементы реализации для FPGA проектов // Системи обробки інформації: збірник наукових праць. – Х.: Харківський університет Повітряних Сил ім. Івана Кожедуба, 2013. – Вип. 9(116). – 230 с.

Второе – оперативное перераспределение функций между такими новыми КЛБ с целью достижения максимального эффекта снижения энергопотребления, не превышая заданных временных ограничений.

Третье положение – оперативная оптимизация каждого конкретного такого перераспределения для данных условий эксплуатации перед загрузкой файла конфигурации в ПЛИС.

Четвертое положение – использование элементов с избыточным базисом для борьбы со сбоями при работе на пониженном напряжении питания.

И, наконец, в-пятых, создание предлагаемых энергоэффективных программно-аппаратных логических элементов может быть эмулировано в существующие ПЛИС с использованием соответствующих систем проектирования. Но может быть поставлена задача разработки новых архитектур, ориентированных на энергоэффективные применения.

Таким образом, рассмотрена существующая логика ПЛИС, выявлена основная проблема, которая влияет на энергоэффективность всей ПЛИС, предложена концепция «зелёной» логики ПЛИС FPGA и предложены технические решения, обеспечивающие поддержку такого подхода.

СИМ-МОДЕЛЬ ЛОКАЛЬНОГО СЕГМЕНТА АКТИВНО-АДАПТИВНОЙ СЕТИ HIL MICROGRID

**Студент А.В. Чудинов,
канд. техн. наук, доцент А.В. Кычкин**
Пермский национальный исследовательский
политехнический университет

В статье рассматриваются перспективы использования СИМ-модели для интеграции информационных компонентов системы активно-адаптивной сети в рамках стенда полунатурного моделирования Hardware-in-the-Loop (HIL) MicroGrid. Для обеспечения эффективного информационного обмена в единой активно-адаптивной сети необходима общая модель электроэнергетической системы, для описания которой целесообразно использовать стандарты МЭК 61968 и МЭК 61970. Данные стандарты предполагают описание классов, объектов и их атрибутов. Для обозначения информационной связи между объектами классов используются наследование, агрегирование и двунаправленные ассоциации. Работа выполнена в рамках гранта Президента Российской Федерации по государственной поддержке молодых российских ученых – кандидатов наук, МК–5279.2014.8 «Синтез эффективных технологий удаленного мониторинга и управления состоянием интеллектуальной электроэнергетической системы с активно-адаптивной сетью».

Ключевые слова: общая информационная модель, активно-адаптивная сеть, МЭК 61970, полунатурное моделирование.

CIM MODEL OF THE HIL MICROGRID LOCAL SEGMENT

**Student A.V. Chudinov,
Ph.D. in Technical Sciences, Associate Professor A.V. Kychkin**
Perm National Research Polytechnic University

The article discusses the prospects of using CIM model for the integration of information system components active – adaptive network within the stand HIL Hardware-in-the-Loop (HIL) MicroGrid. To ensure the effective exchange of information in a single active – adaptive network requires a general model of the power system to which it is advisable to use the described standards IEC 61968 and IEC 61970. These standards require the description of classes, objects and their attributes. To indicate the data communication between objects of classes using inheritance, aggregation, and bi-directional association. Work done under the grant of the President of the Russian Federation on gosu state support of young Russian scientists - candidates, MK-5279.2014.8 "Synthesis efficient technologies remote monitoring and management of the state of intellectual power system with active-adaptive grid".

Keywords: Common Information Model (CIM), active – adaptive network, IEC 61970, Hardware-in-the-Loop (HIL).

Система NIL MicroGrid предназначена для имитации работы активно-адаптивной сети (ААС), включая сеть линий электропередач, к которой подключены модели внешних генераторов, потребители и измерительные контроллеры. Эффективная работа локального сегмента ААС требует выработки и использования общих информационных методов по обмену информацией между ее объектами, создания специальных интерфейсов для унифицированного подключения к сетям; поддержки различных станций управления для возобновляемых источников энергии; наличия достаточного объема информации о текущем состоянии электрической сети, ее элементов (включая векторные измерения), внешней среды (освещенность, осадки, ветровые нагрузки). Для отработки выполнения этих функций на стенде NIL MicroGrid необходимо наладить обмен данными между моделями генераторов, потребителей и системы имитации, аппаратное обеспечение которой выполнено на базе модулей NI 7851R с формированием низковольтных гармонических сигналов промышленной частоты с применением технологии ПЛИС.

Для описания общих механизмов обмена информацией в ААС целесообразно применение международных стандартов Common Information Model (CIM) и Generic Interface Definition (GID). CIM-модель – это обобщенная информационная абстрактная модель описания всех основных объектов ААС, которая все множество элементов системы представляет стандартным образом в виде описания объектов, их свойств и связей между ними. CIM значительно упрощает интеграцию объектов и различных приложений ААС, выполненных независимыми производителями [1].

CIM служит общим языком общения ААС и позволяет эффективно описать оборудование, объекты и процессы потребления и генерации электрической энергии в виде единой объектно-ориентированной модели (элементы ААС представлены в виде совокупности объектов, каждый из которых является экземпляром определенного класса, а классы образуют иерархию наследования). Модель CIM использует UML для описания сущностей, под которыми понимается как физическое оборудование (бытовой потребитель), так и логические объекты модели сети (узлы, ветви), эквивалентные модели участков сети или ее объектов и логические структуры (например, график нагрузки). Число классов, описывающих различные сущности CIM-модели, велико, однако классы могут быть сгруппированы по смыслу

в пакеты. Например, пакет «Топология» включает классы, которые служат для описания соединений оборудования сети; пакет «Модель нагрузки» содержит классы для описания графиков нагрузок потребителей и их моделей [2].

Стандарт GID определяет интерфейсы, которые могут связывать приложения ААС между собой, с хранилищами данных и с интеграционной платформой IIL, определяет общий технический механизм, позволяющий приложениям взаимодействовать и предоставлять свои данные [1] (таблица). Базовые технологии стандартов: OMG Data Acquisition from Industrial Systems (DAIS), OMG Historical Data Access From Industrial Systems (HDAIS), OPC Foundation Simple Eventing.

GID-интерфейсы

Наименование	Назначение
Generic Data Access (GDA)	Работа с данными и метаданными CIM-модели. Предоставление пользователю возможности получения и модификации данных, навигации по CIM-модели без знания этой модели, получение свойств классов, связанных с заданным объектом
Generic Eventing and Subscription (GES)	Публикация сообщений и получение этих сообщений определенными подписчиками
High Speed Data Access (HSDA)	Передача больших объемов данных из SCADA-систем. Позволяет запрашивать информацию о схеме (классах) и об экземплярах этих классов, содержащих данные реального времени
Time Series Data Access (TSDA)	Передача архивных данных. Позволяет запрашивать информацию о схеме (классах) и об экземплярах этих классов, содержащих временные ряды. С помощью TSDA можно получать информацию, организовывать публикацию и подписку на изменения

Обзор классов и атрибутов общей информационной модели электроэнергетических систем в соответствии со стандартом МЭК 61970. Основными используемыми международными стандартами для описания CIM-моделей энергообъектов и энергосистем являются стандарты МЭК 61968 и МЭК 61970, [3]. Стандарт МЭК 61968 определяет требования, архитектуру интеграции и интерфейсы для основных элементов системы управления распределением (DMS) и других связанных внешних IT-систем. В МЭК 61968 представлены разделы, описывающие оборудование, не охвачены разделы, связанные с производственной, финансовой деятельностью энергетических предприятий. МЭК 61970 – комплекс стандартов

создания автоматизированных систем технологического управления энергосистемой. Этот стандарт также построен на базе CIM-представления, которое обеспечивает абстрактную модель для полной энергосистемы на UML, основные классы объектов электроэнергетики представлены на рис. 1.

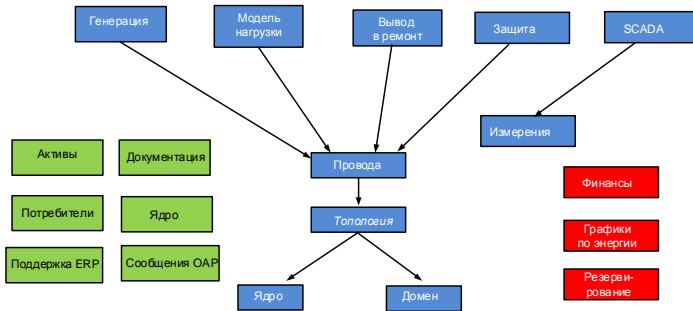


Рис. 1. Основные классы объектов стандарта МЭК 61970

Ядро – пакет, содержащий описания базовых классов, от которых наследуются классы, включенные в остальные пакеты. Домен содержит определения типов данных, включенных в каждый класс. Тип данных кроме значения может также описывать единицу измерения. Например, тип Напряжение служит для описания типа и единицы измерения напряжения. Топология совместно с классами пакета Ядро позволяет моделировать граф сети. Пакет Провода является расширением пакетов Ядро и Топология, которые моделирует информацию об электрических характеристиках сети и оборудования. Вывод в ремонт является расширением пакетов Ядро и Провода, который моделирует информацию о текущей конфигурации сети. Защита является расширением пакетов Ядро и Провода для моделирования устройств защиты. Измерения содержит классы описания данных измерений. Классы пакета Модель нагрузки отвечают за моделирование потребителей электроэнергии любого уровня в виде графиков нагрузки и связанных с ними данных. В пакет включены классы для описания данных, влияющих на прогноз потребления. Генерация делится на две части – Производство и Динамика. Пакет Производство содержит информацию по моделям экономического распределения нагрузки и определения резерва, который используется для выбора сос-

тава генерирующего оборудования и прогноза нагрузки. В пакете Динамика моделируется тепловая часть оборудования станции [4].

Принципы моделирования ААС с использованием абстрактных моделей CIM. Разработка модели взаимодействия генераторов и потребителей в ААС очень трудоемка из-за большого разнообразия электрооборудования. Кроме того, в модели должны быть реализованы импорт и экспорт данных для обмена с другими приложениями и для работы с внешней сетью. Для хранения CIM-модели ААС используется специализированный программный комплекс, позволяющий хранить, редактировать, верифицировать, импортировать и экспортировать модель ААС, а также обмениваться ее фрагментами с другими подсистемами. CIM-модель локального сегмента должна иметь общую среду передачи данных и общее CIM-описание данных и интерфейсов.

Каждому объекту присваивается уникальный идентификатор ресурса (universal resource identifier, URI). В результате создается стандартизованное описание инфраструктуры энергетической сети. Модель CIM предусматривает возможность добавления новых атрибутов в существующие классы, а также новых классов объектов, уточняющих уже существующие классы или связанных с ними [4].

Для наладки информационного обмена разрабатываются стандарты представления CIM-модели в XML-протокол для хранения и управления информацией, который подходит для обмена данными CIM-модели через XML (МЭК 61970 часть 503 CIM XML Model Exchange Format). В XML можно представить классы объектов электроэнергетики и сами объекты со значениями их атрибутов, в том числе со ссылками на другие объекты [5].

Поскольку объекты ААС должны обмениваться между собой информацией в режиме реального времени, то было бы крайне полезно создать единую CIM-модель объектов, описывающую инфраструктуру ААС. Полученную CIM-модель можно выгрузить в XML и загрузить в любое приложение, созданное в соответствии со стандартом CIM. Если это приложение к тому же поддерживает GID-интерфейс, то оно может использоваться как объектно-ориентированное хранилище данных, обращаясь через GID-интерфейс к объектам.

Построение CIM модели IHL MicroGrid. CIM использует терминологию объектно-ориентированного программирования как классы, объекты, атрибуты и наследование. В сегменте ААС классами являются: Нагрузка, Генератор, ЛЭП, Коммутационное оборудова-

ние, Контроллер, Релейная защита и автоматика. Основными свойствами класса являются инкапсуляция, полиморфизм и ассоциации. Инкапсуляция означает сосредоточие всех свойств класса как его атрибутов. Описание атрибута класса вне класса или внутри другого класса недопустимо. Полиморфизм означает, что одно и то же символическое имя атрибута может использоваться в разных классах, но имя класса должно быть уникальным. Ассоциация означает возможность связи классов между собой, т.е. любая пара классов может быть связана ассоциацией, которая, в свою очередь, также является классом.

Для класса Нагрузка объектами являются электрический привод (атрибуты – активное и реактивное сопротивления, температура, скорость вращения); системы телевизор, компьютер, электрическая плита (атрибут – активная нагрузка), системы кондиционирования, холодильник и стиральная машина (атрибуты – активная нагрузка, реактивная мощность).

Для класса Генератор объектами являются ветрогенераторы (атрибуты – скорость ветра, плотность воздуха, диаметр ротора), батарея солнечных элементов (атрибуты – интенсивность солнца, область элементов, КПД).

Для класса ЛЭП с управляемым изменением характеристик атрибуты – активные и реактивные составляющие сопротивлений линии, а также системы контроля состояния (стрел провеса, гололедообразования, систем защиты от разрядов и перенапряжений); коммутационные аппараты (объекты: выключатель, разъединитель) с высокой разъединяющей способностью и большим коммутационным ресурсом; атрибуты микроконтроллеров, такие как память, напряжение питания, частота, являются примитивным типом.

Для обозначения связи между объектами классов будем использовать ассоциации, позволяющие одному классу получить информацию об атрибутах и ассоциациях другого класса, и агрегирование, показывающее, что один класс является частью другого. Выделяется тип ассоциации – наследование, при котором один класс имеет открытый доступ ко всем свойствам другого класса. Из рис. 2 видно, что классы AAC НН MicroGrid являются абстрактными объектами, не имеющими привязки к реальному размещению, обобщенный класс – коммутационное оборудование.

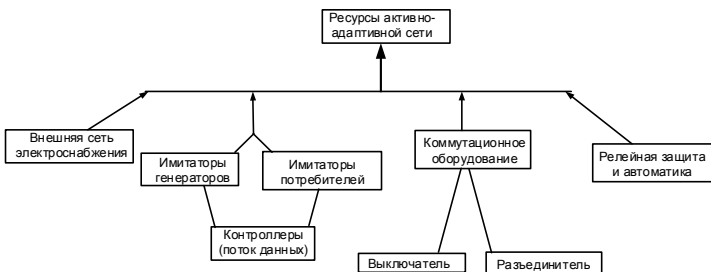


Рис. 2. Схема наследования в NIL MicroGrid

При агрегировании класс включает в себя другой класс, но не является потомком. Агрегирование классов NIL MicroGrid приведено на рис. 3.

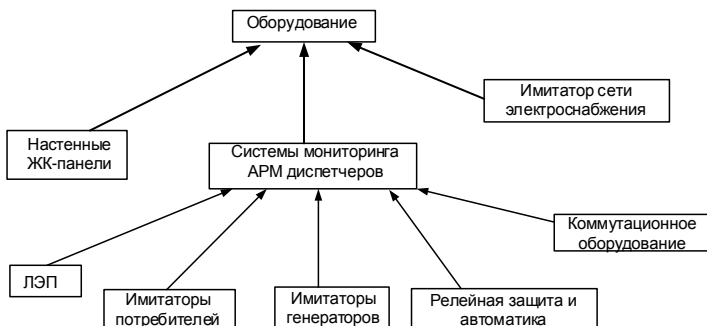


Рис. 3. Схема агрегирования в NIL MicroGrid

При построении двунаправленной ассоциации классы остаются независимыми, но ссылаются друг на друга. При построении ассоциации в NIL MicroGrid выделяется класс «куда», имеющий кратность «1...n», и класс «откуда», кратности «0, 1». На рис. 4 приведен пример ассоциации с кратностью.

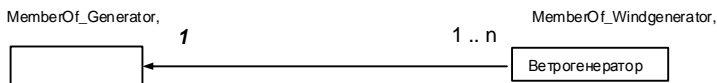


Рис. 4. Двунаправленная ассоциация при описании генераторов локального сегмента AAC NIL MicroGrid

Класс «Генератор» может содержать один «Ветрогенератор» или несколько (1..n), а агрегирование указывает на то, что «Генератор» включает класс «Ветрогенератор».

Таким образом, CIM предлагает возможности, позволяющие полно описать информационный уровень взаимодействия сегмента ААС IHL MicroGrid с помощью объектно-ориентированного подхода в виде классов, атрибутов и их типов. Связи между объектами классов получены с помощью наследования, агрегирования и двунаправленных ассоциаций, что значительно упрощает процессы разработки и анализа топологически, функционально и технически сложных реализаций ААС.

Библиографический список

1. Becker D. Common Information Model (CIM): CIM 10 Version. – Final Report, EPRI, 2001.

2. Федоров В. Стандарты обмена данными в электроэнергетике [Электронный ресурс] // Открытые системы. – 2005. – № 9. – URL: <http://www.osp.ru/os/2005/09/380385/> (дата обращения: 12.10.2015).

3. Багрянцев А.А., Машалов Е.В., Неуймин В.Г. Применение CIM-модели для представления коммутационных схем в задачах расчета и оптимизации установившихся режимов // Энергосистема: управление, конкуренция, образование: материалы III Междунар. науч.-практ. конф. – Екатеринбург, 2008.

4. МЭК 61970 Словарь локализованных терминов общей информационной модели. – М., 2011.

5. DeVos A., Widergren S.E. XML for CIM model exchange // IEEE Publishing Edt. – The 22nd International Conference on Power Industry Computer Applications, 2001.

Секция 4

**ТЕЛЕКОММУНИКАЦИОННЫЕ СИСТЕМЫ И СЕТИ.
ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ**

РАЗРАБОТКА ПРОСТЕЙШЕЙ СИСТЕМЫ ОБНАРУЖЕНИЯ ВТОРЖЕНИЙ

Канд. техн. наук, доцент А.Н. Кокоулин,
студенты А.С. Бадртдинов, М.А. Тихоненко
Пермский национальный исследовательский
политехнический университет

В учебных целях разработана программа, выполняющая функции простейшей системы обнаружения вторжений – определение пакетов, соответствующих установленному фильтру.

Ключевые слова: система обнаружения вторжений, tcpdump, golang.

DEVELOPING A SIMPLE INTRUSION DETECTION SYSTEM

Ph.D. in Technical Sciences, Associate Professor A.N. Kokoulin,
Students A.S. Badrtdinov, M.A. Tikhonenko
Perm National Research Polytechnic University

The program is designed for educational purposes, which serves as a simple intrusion detection system – defining packages corresponding to the filter.

Keywords: intrusion detection system, tcpdump, golang.

Обнаружение вторжений и несанкционированных воздействий в сети является актуальной проблемой современного информатизированного общества. Считается, что начало активному исследованию в этой области было положено в 1980 г. статьей Джеймса Андерсона «Мониторинг угроз компьютерной безопасности». Несколько позже, в 1987 г. это направление было развито публикацией статьи «О модели обнаружения вторжения» Дороти Деннинг. Она обеспечила методологический подход, вдохновивший многих исследователей и заложивший основу для создания коммерческих продуктов в области обнаружения и предотвращения вторжений [1].

В данной статье представлена разработанная в учебных целях программа-пример элементарной системы обнаружения вторжений, основанная на анализаторе сетевого трафика tcpdump. В качестве языка программирования выбран язык Go, совмещающий в себе производительность, близкую к производительности языка C, и вы-

сокую скорость разработки, схожую, например, с языком программирования Python.

Система обнаружения вторжений – программное или аппаратное средство, предназначенное для выявления фактов несанкционированного доступа в информационную систему или сеть либо несанкционированного управления ими в основном через Интернет. Они обеспечивают дополнительный уровень защиты компьютерных систем путем фильтрации поступающих пакетов [2].

Системы обнаружения вторжений используются как для выявления неблагоприятных для компьютерной системы пакетов, так и для проверки работы сети и выявления ошибок и недочетов.

Go – язык общего назначения с широкими возможностями и понятным синтаксисом. Благодаря мультиплатформенности, надежной, хорошо документированной стандартной библиотеке и ориентированности на удобные подходы к самой разработке Go является идеальным языком для первых шагов в программировании [3].

Tcpdump (от TCP и англ. dump – свалка, сбрасывать) – утилита UNIX, позволяющая перехватывать (capture) и анализировать сетевой трафик, проходящий через компьютер, на котором запущена данная программа [4]. Позволяет фильтровать поступающие пакеты с помощью большого числа встроженных функций.

В программе, листинг которой представлен далее, используется следующая команда запуска утилиты tcpdump: «tcpdump -i eth0 -n -x src not hostname». Описание команды: «tcpdump» – название утилиты; «-i eth0» – выбор интерфейса eth0 для захвата проходящих сетевых пакетов; «-n» – отображает IP-адрес вместо имени хоста; «-x» – делает распечатку пакета в шестнадцатеричной системе, полезно для более детального анализа пакета, количество отображаемых данных зависит от параметра -s; «src not hostname» – отбрасываются исходящие пакеты от машины, на которой запущена программа.

```
package main
import (
    "bufio"
    "fmt"
    "os/exec"
    "os"
    "regexp"
```

```

)
func main() {
    filters := [2]string{"(ya|yandex)\\.ru",
"(fb|facebook)\\.com"}
    hostname, _ := os.Hostname()
    cmd := exec.Command("tcpdump", "-i", "eth0",
"-n", "-x", "src", "not", hostname)
    out, err := cmd.StdoutPipe()
    if err != nil {
        panic(err)
    }
    defer out.Close()
    bufOut := bufio.NewReader(out)
    if err = cmd.Start(); err != nil {
        panic(err)
    }
    for {
        result, _, err := bufOut.ReadLine()
        if err != nil {
            panic(err)
        }
        for _, filter := range filters {
            match, _ := regexp.MatchString(filter,
string(result))
            if match {
                fmt.Println("WARNING:", filter)
            }
        }
    }
}

```

Алгоритм работы программы заключается в следующем: программа ищет соответствия сетевых пакетов, захваченных утилитой tcpdump, и элементов массива фильтров, содержащего регулярные выражения для поиска.

Результат работы программы представлен на рис. 1. На этом же рисунке видны и использованные простые тестовые регулярные выражения ("(ya|yandex)\\.ru", "(fb|facebook)\\.com") [5].

```
WARNING: (ya|yandex)\.ru
WARNING: (ya|yandex)\.ru
WARNING: (fb|facebook)\.com
WARNING: (fb|facebook)\.com
WARNING: (fb|facebook)\.com
WARNING: (fb|facebook)\.com
WARNING: (fb|facebook)\.com
WARNING: (fb|facebook)\.com
WARNING: (fb|facebook)\.com
```

Рис. 1. Вывод программы

Фрагмент результата выполнения команды, описанной выше, представлен на рис. 2.

```
root@HomePC:~# tcpdump -i eth0 -n -x src not HomePC
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
01:13:05.304132 IP 87.240.134.101.80 > 192.168.1.2.41167: Flags [P.], seq 181545
221:181545552, ack 1377381792, win 47, options [nop,nop,TS val 687242139 ecr 119
76636], length 331
 0x0000: 4528 017f 1aa8 4000 3706 87a9 57f0 8665
 0x0010: c0a8 0102 0050 a0cf 0ad2 2905 5219 2da0
 0x0020: 8018 002f 4d3a 0000 0101 080a 28f6 7b9b
 0x0030: 00b6 bfb3 4854 5450 2f31 2e31 2032 3030
 0x0040: 204f 4b0d 0a53 6572 7665 723a 2041 7061
 0x0050: 6368 650d 0a44 6174 653a 204d 6f6e 2c20
 0x0060: 3232 204a 756e 2032 3031 3520 3230 3a31
 0x0070: 333a 3035 2047 4d54 0d0a 436f 6e74 656e
 0x0080: 742d 5479 7065 3a20 7465 7874 2f6a 6176
 0x0090: 6173 6372 6970 743b 2063 6861 7273 6574
 0x00a0: 3d55 5446 2d38 0d0a 436f 6e74 656e 742d
 0x00b0: 4c65 6e67 7468 3a20 3132 350d 0a43 6f6e
 0x00c0: 6e65 6374 696f 6e3a 206b 6565 702d 616c
 0x00d0: 6976 650d 0a50 7261 676d 613a 206e 6f2d
 0x00e0: 6361 6368 650d 0a43 6163 6865 2d43 6f6e
 0x00f0: 7472 6f6c 3a20 6e6f 2d73 746f 7265 0d0a
 0x0100: 0d0a 5b7b 2274 7322 3a22 3238 3734 3136
 0x0110: 3430 3622 2c22 6576 656e 7473 223a 5b5d
 0x0120: 7d2c 7b22 7473 223a 2231 3935 3037 3436
 0x0130: 3839 3122 2c22 6576 656e 7473 223a 5b5d
 0x0140: 7d2c 7b22 7473 223a 2231 3433 3237 3839
 0x0150: 3736 3222 2c22 6576 656e 7473 223a 5b22
 0x0160: 3233 3c21 3e6f 6666 6c69 6e65 3c21 3e34
 0x0170: 3532 3031 3030 333c 213e 3122 5d7d 5d
01:13:05.309139 IP 192.168.1.2.41167 > 87.240.134.101.80: Flags [P.], seq 1:1009
, ack 331, win 392, options [nop,nop,TS val 11982620 ecr 687242139], length 1008
 0x0000: 4500 0424 9712 4000 4006 ffc1 c0a8 0102
 0x0010: 57f0 8665 a0cf 0050 5219 2da0 0ad2 2a50
 0x0020: 8018 0188 a416 0000 0101 080a 00b6 d71c
 0x0030: 28f6 7b9b 504f 5354 202f 696d 3636 3820
 0x0040: 4854 5450 2f31 2e31 0d0a 486f 7374 3a20
 0x0050: 7134 2e71 7565 7565 7634 2e76 6b2e 636f
 0x0060: 6d0d 0a43 6f6e 6e65 6374 696f 6e3a 206b
 0x0070: 6565 702d 616c 6976 650d 0a43 6f6e 7465
 0x0080: 6e74 2d4c 656e 6774 683a 2032 3339 0d0a
 0x0090: 4f72 6967 696e 3a20 6874 7470 3a2f 2f71
```

Рис. 2. Вывод утилиты tcpdump

Система обнаружения вторжений – лишь один из инструментов защиты важных данных, и он не должен рассматриваться как замена любого из других защитных механизмов.

Библиографический список

1. Бобров А. Системы обнаружения вторжений [Электронный ресурс]. – URL: <http://www.icmm.ru/~masich/win/lexion/ids/ids.html> (дата обращения: 02.06.2015).

2. Система обнаружения вторжений [Электронный ресурс]. – URL: https://ru.wikipedia.org/wiki/Средство_обнаружения_вторжений (дата обращения: 02.06.2015).

3. Приступая к работе. Введение в программирование на Go [Электронный ресурс]. – URL: <http://golang-book.ru/chapter-01-getting-started.html> (дата обращения: 02.06.2015).

4. Tcpdump [Электронный ресурс]. – URL: <https://ru.wikipedia.org/wiki/Tcpdump> (дата обращения: 04.06.2015).

5. Регулярные выражения [Электронный ресурс]. – URL: https://ru.wikipedia.org/wiki/Регулярные_выражения (дата обращения: 02.06.2015).

О ПРИМЕНЕНИИ SIEM-СИСТЕМ ДЛЯ ОБЕСПЕЧЕНИЯ БЕЗОПАСНОСТИ КОРПОРАТИВНЫХ СЕТЕЙ

**Студент В.И. Борисов,
канд. техн. наук, доцент А.С.Шабуров**
Пермский национальный исследовательский
политехнический университет

В статье анализируются проблема роста интенсивности информационного обмена в современных корпоративных сетях, а также обусловленная подобным ростом задача выявления инцидентов информационной безопасности. Предлагается решение проблемы анализа информационных массивов осуществлять на основе внедрения SIEM-систем.

Ключевые слова: корпоративная сеть, инцидент, SIEM-система, DLP-система, лог-файл, бессигнатурный метод корреляции.

ABOUT APPLICATION OF THE SIEM-SYSTEMS FOR ENSURING SAFETY OF CORPORATE NETWORKS

**Student V.I. Borisov,
Ph.D. in Technical Sciences, Associate Professor A.S. Shaburov**
Perm National Research Polytechnic University

In this article analyzes the problem of growing intensity of information exchange in modern enterprise network, as well as due to this growth, the task of detecting information security incidents. The solution of the problem of the analysis of information arrays is proposed to carry out on the basis of introduction of SIEM-systems.

Keywords: enterprise network, incident, SIEM-system, DLP-system, Log-file, assignatures.

В настоящее время достаточно актуальной является проблема количества информации, обрабатываемой в информационных системах, объемы которой растут в геометрической прогрессии. Интенсивность информационного обмена автоматизированных и телекоммуникационных систем обусловлена развитием и разнообразием бизнес-процессов и задач, решаемых в рамках систем управления предприятиями и организациями различных форм собственности. Вместе с тем растет и количество вредоносной информации, такой как вирусы, троянские программы и т.д., а также разнообразие злоумышленных информационных атак на ресурсы систем: фишинг, брутфорс и т.п. При этом варианты и пути проникновения вредоносной информации в корпоративные

информационные системы также постоянно видоизменяются и модифицируются, что делает задачу обеспечения информационной безопасности подобных систем достаточно сложной. Интенсивность информационных событий в корпоративных сетях может достигать нескольких миллионов в день, поэтому возникает проблема нахождения и локализации вредоносной информации в огромных информационных массивах. При этом обработка подобных событий в ручном режиме не представляется возможной, так как потребовала бы значительных человеческих и временных затрат, а также недопустимых с точки зрения эффективности аппаратно-программных ресурсов. Решение данной проблемы целесообразно на основе применения SIEM-систем.

SIEM (System information event management) – системы, появившиеся в результате слияния SEM-систем и SIM-систем. Основным функциональным отличием данных систем является то, что SEM-системы предназначены для анализа информации в режиме реального времени, а SIM-системы анализируют уже накопленную информацию [1].

Основной функцией SIEM-систем является анализ информации, поступающей от разных источников, таких как системы DLP, средства антивирусной защиты информации, межсетевые экраны, системы учета трафика, сканеры уязвимости и т.д. На основе анализа данных из этих источников выявляются отклонения от нормального функционирования, заданного критериями безопасности, и в случае обнаружения происходит оповещение администратора безопасности.

Кроме того, типовая SIEM-система может использоваться:

- для анализа информации, поступающей от различных источников;
- предоставления доказательной базы при расследовании инцидентов информационной безопасности;
- предоставления структурированной информации, необходимой при аудите информационной безопасности;
- обеспечения непрерывности работы сервисов путем обнаружения сбоев в их работе;
- структуризации информационно-телекоммуникационной системы.

Перед SIEM-системой ставятся основные задачи обеспечения информационной безопасности:

- объединение и хранение журналов событий;
- анализ событий и разбор инцидентов;

- обработка и корреляция событий;
- оповещение об инцидентах и инцидент-менеджменте.

В общем SIEM-система состоит из следующих основных компонентов (рис. 1):

- агенты, собирающие данные из различных источников;
- серверы-коллекторы, аккумулирующие информацию с агентов;
- сервер корреляции;
- сервер баз данных.



Рис. 1. Структура основных компонентов SIEM-систем

Функционирование SIEM-системы можно разбить на несколько уровней (рис. 2):

- сбор лог-файлов и формирование необходимых данных от различных источников;
- нормализация данных, заключающаяся в приведении событий с одинаковым смыслом к общему формату;
- корреляция событий системы, важных для обеспечения безопасности, путем нахождения связей между ними, например, подбор паролей, заражение вредоносным кодом, аномальная активность в системе, изменение критических параметров системы и т.п.
- организация хранения лог-файлов;
- реагирование на инциденты, в том числе уведомления о важных событиях для информационной безопасности;
- визуализация инцидентов, формирование отчетных документов.



Рис. 2. Уровни SIEM-систем

Наиболее важным этапом работы SIEM-системы для принятия решения может считаться процесс корреляции событий, важных с точки зрения безопасности. Рассмотрим данный процесс подробнее.

После получения информации от источников система начинает анализировать эту информацию. В большинстве SIEM уже есть стандартный набор правил корреляции (Rule Based Reasoning) [3]. Данные правила состоят из определенных наборов условий и сценариев действий. Данные правила корреляции основаны на бессигнатурных методах корреляции, т.е. система сама определяет появление нежелательных процессов и обеспечивает их фиксирование и подсчет. Кроме того, существуют сигнатурные методы корреляции, в которых правила определения инцидентов в конкретную группу устанавливаются оператором SIEM-системы [3]. При этом важно, что SIEM-системы не способны предотвратить инциденты. Они лишь обрабатывают произошедшие события и анализируют их, а на основании статистических данных можно своевременно выявить бреши в информационной безопасности и предпринять меры по защите информации для будущих инцидентов.

SIEM-система также позволяет существенно облегчить работу для руководства любой компании. Все инциденты сохраняются,

а значит, есть ответственный за обработку этого инцидента и сроки обработки инцидента. На основании статистических данных можно судить об эффективности работы подразделения по защите информации и соответствующих средств защиты. Возможность автоматического составления отчетов с наглядными таблицами и диаграммами способствует снижению сложности обоснования затрат на информационную безопасность.

Основной потребитель SIEM-систем на сегодняшний день – банковская сфера. Это обусловлено тем, что:

1. Банкам необходимо регулярно проводить аудиты соответствия требованиям информационной безопасности.

2. Банки работают с важной информацией, утечка которой повлечет за собой большие финансовые потери.

3. Банки в состоянии обеспечить приобретение и функционирование (аппаратные мощности) SIEM-систем, для чего необходимы большие финансовые вложения.

4. Ежедневно в банковских системах происходит значительное количество событий, как правило, важных с точки зрения информационной безопасности.

5. Наличие SIEM-систем обусловлено требованиями документов, таких как SOX, PCI DSS, ISO 27001, COBIT и др.

Развитие рынка информационных услуг, а также функциональных возможностей SIEM-систем обуславливает возрастающий спрос на их внедрение. С 2009 по 2014 г. объем российского рынка подобных систем вырос 9 до 20 млн долларов. Эксперты прогнозируют 20%-ный рост в 2015 г. в сравнении с 2014 (рис. 3).

Эффективность внедрения SIEM-систем подтверждается статистическими данными предприятий различных секторов экономики. Например, ежедневный объем анализируемых операций финансовой организации достигает порядка 30 млн событий для выявления порядка 30 из них, которые могут нанести ущерб информационной инфраструктуре. Энергетическая компания анализирует ежедневно порядка 2 млрд событий для выявления 20–25 инцидентов. Обработка подобных информационных массивов в ручном режиме, а также принятие эффективных решений по противодействию инцидентам информационной безопасности не представляются возможными.

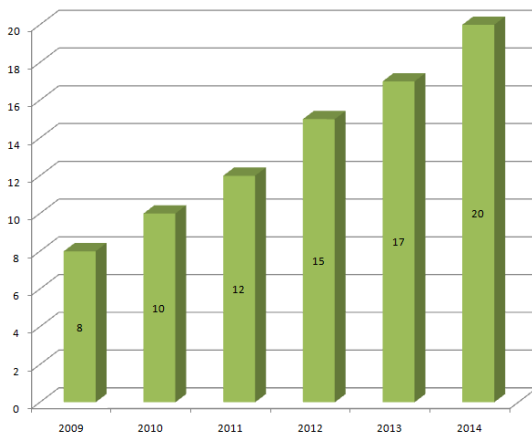


Рис. 3. Объемы рынка SIEM-систем в России:
■ – российский рынок SIEM, млн \$

Таким образом, использование SIEM-систем позволяет свести к минимуму затраты на обработку событий как в финансовом плане, так и в плане времени обработки информации, что обуславливает достаточно интенсивное внедрение подобного класса систем для обеспечения безопасности корпоративных сетей.

Библиографический список

1. Дрозд А. Обзор SIEM-систем [Электронный ресурс]. – URL: http://www.anti-malware.ru/analytics/Technology_Analysis/Overview_SECURITY_systems_global_and_Russian_market (дата обращения: 19.09.2015).
2. Кузнецов А., Федоров А. Современные тенденции развития SIEM-решений [Электронный ресурс] // Storage News. – 2013. – № 2(54). – URL: http://www.ntc-vulkan.ru/images/stories/publication/Vulkan_IS_54-9_final.pdf (дата обращения: 19.09.2015).
3. Олеся Шелестова. Корреляция SIEM. Сигнатурные методы [Электронный ресурс] // SecureLab 17 октября 2012. – URL: <http://www.securitylab.ru/analytics/431459.php> (дата обращения: 19.09.2015).

ПОСТРОЕНИЕ МОДЕЛИ ПРИЛОЖЕНИЯ УЗЛА ПРОМЫШЛЕННОЙ РАСПРЕДЕЛЁННОЙ ИНФОРМАЦИОННО- УПРАВЛЯЮЩЕЙ СИСТЕМЫ

Студент В.А. Васиньков

Пермский национальный исследовательский
политехнический университет

В статье предлагается имитационная модель приложения узла распределённой информационно-управляющей системы для оценки информационного запаздывания контроля и управления техническими объектами и технологическими процессами.

Ключевые слова: имитационная модель, прикладная подсистема, приложение узла, распределённая информационно-управляющая система, сетевая система управления, Fieldbus.

DEVELOPMENT MODEL APPLICATION NODE FIELDBUS SYSTEM

Student V.A. Vasinkov

Perm National Research Polytechnic University

The paper proposes a simulation model of the node application distributed control system for the evaluation of the information delay control and management of technical objects and technological processes.

Keywords: simulation model, application subsystem, application site, Distributed Information Management System, network management system, Fieldbus.

Современное производство невозможно представить без использования компьютерных распределённых информационно-управляющих систем и сетей. Это обусловлено известной целесообразностью и высокой эффективностью их применения в автоматизированных системах для контроля и управления техническими объектами и технологическими процессами [1]. Современное развитие и усложнение технологий функционирования автоматизированных систем вызваны, в частности, ужесточающимися требованиями в области скорости обработки технологических данных и качества контроля и управления. Таким образом, особо актуальным становится решение задачи проектирования быстродействующих распределённых информационно-управляющих систем и основополагающей задачи количественной оценки вероятностно-временных характеристик [2]. Поэтому настоящая работа посвящена

анализу промышленной сети как основы распределённой ИУС и, в частности, моделированию значимой её прикладной подсистемы и анализу её характеристик.

Основой современных распространённых централизованных информационно-управляющих систем является мощное вычислительное устройство, обрабатывающее данные с подключённых датчиков и управляющее исполнительными механизмами [3]. Централизованный подход, основанный на едином вычислительном центре, включающем большое количество кабельных линий, характеризуется множеством известных проблем на этапах ввода в эксплуатацию и эксплуатации системы. Эффективной альтернативой являются децентрализованные ИУС, основой которых являются промышленные Fieldbus-сети. По сравнению с централизованной системой они имеют ряд преимуществ, а именно: простота подключения на основе единственной проводной линии связи (промышленной шины), возможность создания крупных распределённых систем контроля и управления распределёнными объектами и процессами, низкая эксплуатационная стоимость, высокая скорость работы за счёт распределения задач среди вычислительных узлов, большая надёжность, возможность передачи и обработки большого количества данных существующих сетей связи [3]. Таким образом, проектирование и анализ таких систем представляют особый интерес.

Процесс проектирования промышленной сети с требуемой производительностью должен осуществляться с использованием адекватных моделей функционирования и методов количественной оценки вероятностно-временных характеристик сети. При проектировании системы контроля и управления объектами важными являются задачи оценки, учёта и планирования информационного запаздывания, вызванного распределённым взаимодействием элементов системы (датчиков, контроллеров, исполнительных механизмов). Информационное запаздывание негативно влияет на качество управления и контроля проектируемой системы. Так, управление инерционными, т.е. медленными процессами, не должно характеризоваться частым измерением, что приведёт к перегрузке сети неважной технологической информацией. Контроль и управление малоинерционными процессами должны производиться с учётом вычислительного запаздывания, которое не должно превышать запаздывание объекта-регулятора, регулятора-объекта. В иных случаях несогласованное с контуром управления информаци-

онное запаздывание вызовет нарушение устойчивости и требуемого качества управления. Это определяет актуальность поставленной задачи количественной оценки вероятностно-временных характеристик промышленной сети РИУС.

В настоящий момент указанная проблема влияния информационно-го запаздывания решается на этапе эксплуатации системы. Решение проблемы требует привлечения аппаратно-программных средств анализа характеристик системы, что определяет высокие экономические и трудовые затраты. Поэтому для устранения указанного недостатка целесообразными являются адекватный расчёт и учёт информационного запаздывания на этапе проектирования Fieldbus-сети.

Информационное запаздывание между элементами системы контроля и управления (датчиками, регуляторами, контроллерами) является результатом совокупности сетевых задержек передачи и обработки технологической информации транспортной и прикладной подсистемами fieldbus. Большое количество работ оценки сетевых задержек связано с оценкой транспортной подсистемы, определяющей задержки передачи по сети между датчиками и исполнительными механизмами. На фоне увеличения скорости передачи данных актуальным становится учет вычислительной задержки, вносимой прикладной подсистемой (приложением) узла промышленной сети.

Для оценки вычислительной задержки прикладной подсистемы целесообразным является использование имитационного моделирования, позволяющего учесть весь необходимый спектр особенностей и параметров функционирования для получения адекватных результатов. Выполненный анализ систем моделирования MatLab (SciLab), MathCad, GPSS определил нецелесообразность их применения. Это обусловлено ограничениями дискретно-событийного моделирования, свойственными процессному подходу к разработке модели. Преимуществом дает использование агентного подхода при разработке сложных и адекватных моделей промышленной сети, реализованного, в частности, в применяемой СИМ AnyLogic [5]. Данная система моделирования характеризуется полнофункциональностью объектно-ориентированного подхода к разработке модели и лишена недостатка графического интерфейса разработки, ограничивающего возможность быстрого масштабирования модели.

Для моделирования и оценки характеристик прикладной подсистемы кратко рассмотрим принципы её функционирования.

Последовательность выполнения прикладных задач по контролю значений технологических параметров объекта контроля и управления выполняется по циклическому алгоритму и зависит от происходящих событий. События определяются через операторы условий WHEN, представляющих собой логические выражения проверки значений контролируемых параметров. Когда выражение проверки WHEN истинно, начинается выполнение задачи TASK. Некоторые задачи могут быть определены приоритетными. Допускается использование в программе 80 операторов WHEN-TASK, из них приоритетных до 10 [5]. Алгоритм функционирования планировщика определён циклической дисциплиной обслуживания с приоритетами (рис. 1).

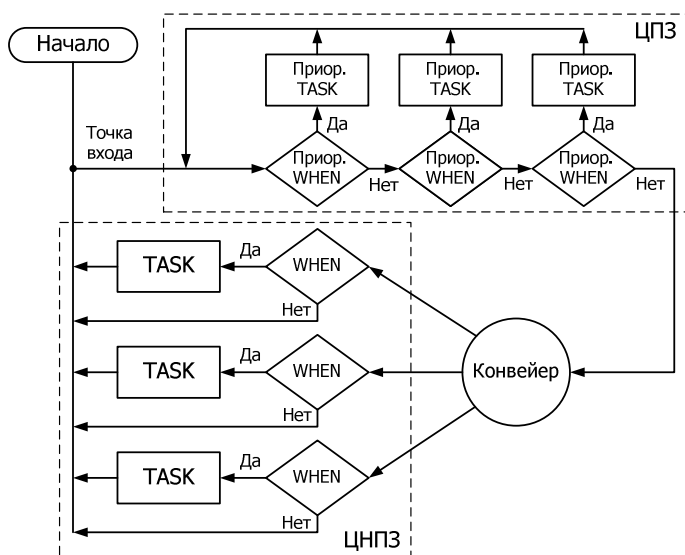


Рис. 1. Алгоритм работы планировщика задач

Начало работы планировщика составляет цикл приоритетных задач (ЦПЗ). Последовательно проверяются приоритетные условия WHEN. Если проверяемое условие истинно, то выполняется соответствующая задача, а после её выполнения порядок проверки начинается сначала. ЦПЗ длится до тех пор, пока результат последовательной проверки всех приоритетных условий не окажется ложным. По окончании ЦПЗ выполняется проверка одного, в порядке последовательности,

неприоритетного условия. Вне зависимости от результатов проверки неприоритетной задачи начинается новый ЦПЗ. Таким образом, полный цикл неприоритетных задач (ЦНПЗ) подразумевает прохождение, как минимум, N ЦПЗ, равное числу неприоритетных условий, и может быть предсказуемо. В случае высокой частоты срабатывания приоритетных условий время реакции на неприоритетные условия становится очень большим и становится непредсказуемым [8].

В соответствии с изложенными принципами функционирования и структурой модель приложения узла можно представить тремя основными состояниями: точкой входа (pointOfEntry), проверкой when, задачей (task) (рис. 2).

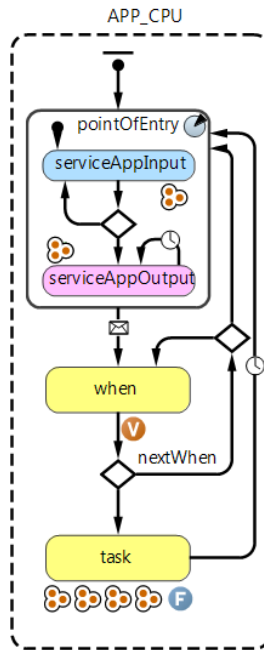


Рис. 2. Модель приложения узла

Процесс функционирования модели начинается с инициализации исходных параметров, хранимых в информационных массивах: данных о времени последнего выполнения задач, данных о характеристиках (интервалах времени) проверок, количестве прикладных модулей различного приоритета.

В состоянии точки входа «pointOfEntry» осуществляются прием и обработка информации из входных сетевых сообщений (пакетов). В начальном состоянии «serviceAppInput» происходит получение пакета, назначение обслуживающего модуля, установка времени события. На выходе из состояния «serviceAppInput» имитируется задержка чтения, записи и обработки. Если остались пакеты, осуществляется возврат к состоянию «serviceAppInput», иначе – по переходу в состояние «serviceAppOutput». В состоянии «serviceAppOutput», при наличии сформированных информационных сообщений, их отправляют на дальнейшую обработку другим подсистемам и удаляют из внутренней памяти приложения. Когда все сообщения будут отправлены, происходит переход в цикл обслуживания задач, к первой приоритетной проверке условия «when».

В состоянии «when» выполняется проверка условия текущего модуля приложения, его приоритета. На выходе из «when» имитируется задержка проверки условия и определяется результат проверки путём сравнения интервала времени с момента последнего выполнения задачи модуля с типовым средним конфигурационным временем событий. Если проверка «when» прошла успешно, выполняется задача «task», иначе осуществляется переход к следующей проверке «when». Далее, если выполнялась обработка неприоритетного модуля, управление передаётся в состояние «pointOfEntry», иначе управление алгоритма переходит к приоритетной проверке «when».

В состоянии «task» производится выполнение задачи, сопровождаемой задержкой. При обработке входящего сообщения собирается статистика и обновляется время срабатывания задачи «task», формируется ответ – информационное сообщение, если того требует тип сообщения. Если тип модуля приложения «исходящий», то обновляется время срабатывания задачи «task» и вырабатывается новое информационное сообщение. После выполнения задачи «task» имитируется задержка её выполнения и осуществляется переход к состоянию точки входа «pointOfEntry» для завершения обработки задачи.

Разработанная модель приложения узла промышленной сети позволяет оценить значимую часть информационного запаздывания между распределёнными элементами системы контроля и управления, что способствует корректному синтезу системы на этапе планирования fieldbus-сети.

Библиографический список

1. Дитрих Д., Лой Д., Швайцнер Г.-Ю. Д 492 ЛОН-технология. Построение распределенных приложений: пер. с нем. / под ред. О.Б. Низамутдинова. – Пермь: Звезда, 1999. – 424 с.
2. Современные технологии автоматизации [Электронный ресурс]. – URL: <http://www.cta.ru/cms/f/340834.pdf> (дата обращения: 11.11.2015).
3. Ковалёв А.В., Хнаду Архитектура построения систем управления и сбора автоматизации [Электронный ресурс]. – URL: http://refereed.ru/ref_6afef7f597dd5dcfdd8035c008001a99.html (дата обращения: 11.11.2015).
4. Даденков С.А., Кон Е.Л. Исследование производительности алгоритма доступа к среде predictive p-persistent CSMA протокола // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2012. – № 6. – С. 217–230.
5. Даденков С.А., Кон Е.Л. Анализ моделей и методов агентного и дискретно-событийного имитационного моделирования // Известия СПбГЭТУ «ЛЭТИ». – 2015. – № 5. – С. 35–41.
6. Даденков С.А., Кон Е.Л. Метод расчета числа узлов-соперников за канал в сети со случайным множественным доступом, построенной на основе нейрочипов // Нейрокомпьютеры: разработка, применение. Радиотехника. – 2015. – № 10. – С. 26–32.
7. Даденков С.А., Кон Е.Л., Южаков А.А. Алгоритм проектирования промышленной информационно-управляющей сети требуемой производительности (на примере технологии LonWorks) // Вестник Поволж. гос. технол. ун-та. Сер. Радиотехнические и инфокоммуникационные системы. – 2014. – № 5(24). – С. 35–43.
8. Даденков С.А., Кон Е.Л. Оценка степени влияния некоторых факторов на производительность LonWorks-сети // Образовательные ресурсы и технологии. – 2014. – № 2(5). – С. 72–76.

АНАЛИЗ ПРОТОКОЛА ДОСТУПА И ПЕРЕДАЧИ ТЕХНОЛОГИИ ДЕЦЕНТРАЛИЗОВАННОЙ АВТОМАТИЗАЦИИ EIB/KNX

Студентка Д.Н. Гайнанова

Пермский национальный исследовательский
политехнический университет

В работе производится анализ принципов функционирования алгоритма множественного доступа и передачи узлами технологической информации по разделяемой среде распределённой управляющей системы с технологией EIB/KNX.

Ключевые слова: избегание коллизий, метод случайного множественного доступа, телеграмма, CSMA/CA, EIB/KNX.

ANALYSIS ACCESS PROTOCOLS AND TECHNOLOGY TRANSFER DISTRIBUTED CONTROL SYSTEM EIB/KNX

Student D.N. Gaynanova

Perm National Research Polytechnic University

The paper analyzes the principles of functioning of the algorithm of multiple access nodes and transmission technology information on a shared medium of a distributed control system technology EIB/KNX.

Keywords: prevent collision, method of random multiple access, telegram, CSMA/CA, EIB/KNX.

Введение. На рынке средств автоматизации зданий существует целый класс технологических платформ – открытых систем децентрализованной автоматизации. Наиболее известной и распространённой на сегодняшний день открытой системой в области автоматизации зданий является сетевая технология децентрализованной автоматизации EIB/KNX. Стандарт EIB/KNX может использоваться для автоматизации как промышленных зданий и офисов, так и обычных жилых домов [1–2]. Протокол доступа и передачи измерительной и управляющей информации CSMA (множественный доступ с контролем несущей), нашедший применение в сети KNX, используется и в других промышленных сетях реального времени, например, CAN, Industrial Ethernet, LonWorks, Modbus/TCP и др. Представленное определяет интерес к исследованию протокола доступа и сравнительному анализу с другими. Анализ позволит определить преимущества и недостатки алгоритма, его эффективность при различной нагрузке

на сетевой канал (количество узлов, создающих нагрузку и интенсивность) [2–10]. Это позволит использовать алгоритм более эффективно для решения класса определённых задач, в том числе связанных с реальным временем. Для решения данной задачи в статье выполняются анализ алгоритма доступа, выделение этапов его функционирования, что необходимо для дальнейшей реализации модели.

Технология доступа узлов к разделяемой среде передачи.

Процесс обмена данными между устройствами по шине передачи является событично управляемым. Данные посылаются в шину отдельными пакетами (телеграммами) друг за другом. Таким образом, в одно и то же время по шине передаётся только один пакет данных от одного конкретного шинного устройства. Из соображений надёжности для осуществления обмена информационными телеграммами и для доступа к шине применяется метод децентрализованного доступа CSMA/CA (Carrier Sense Multiple Access/Collision Avoidance). Рассмотрим подробнее алгоритм доступа к среде передачи данных (рис. 1).

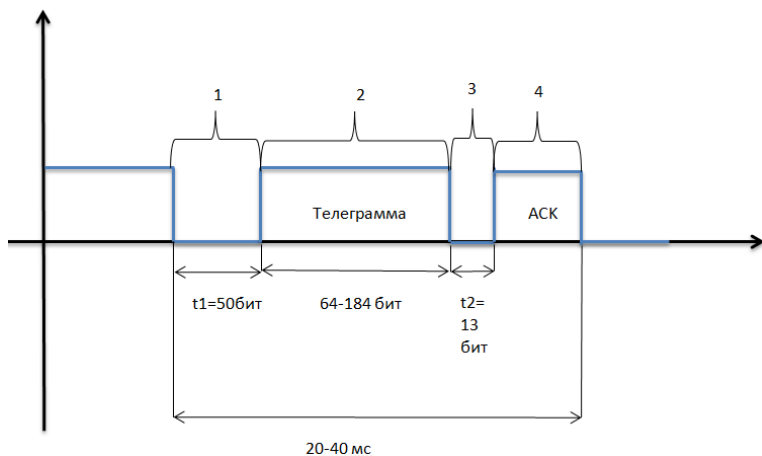


Рис. 1. Временная диаграмма доступа и передачи телеграммы

Алгоритм доступа к среде основан на следующих этапах:

1. Проверка активности канала и межтелеграммный интервал. Узел следит за состоянием канала (прослушивает несущую) и выжидает с момента последней передачи фиксированный временной интервал продолжительностью $t_1 = 50$ бит времени для определения того, что канал свободен.

2. Передача информационной телеграммы. Узел, в случае наличия телеграммы, начинает побитовую передачу. В случае, если у узла нет данных для передачи, то он не участвует в доступе к каналу и ожидает появления информации, после чего осуществляет только проверку неактивности канала в течение интервала t_1 (этап 1) и может начать этап передачи. Телеграмма передаётся последовательно отправляемыми пакетами размером 11 бит, включающими служебный или пользовательский байт данных телеграммы и служебные биты пакета, в том числе контроля чётности. После передачи каждого пакета выжидается интервал, равный двум битам времени. Таким образом, задержка передачи одного пакета составляет 13 бит времени. Для передачи телеграммы могут потребоваться от 8 и до 23 пакетов, в зависимости от типа и объёма передаваемых в телеграмме пользовательских данных. Общая структура телеграммы представлена на рис. 2.

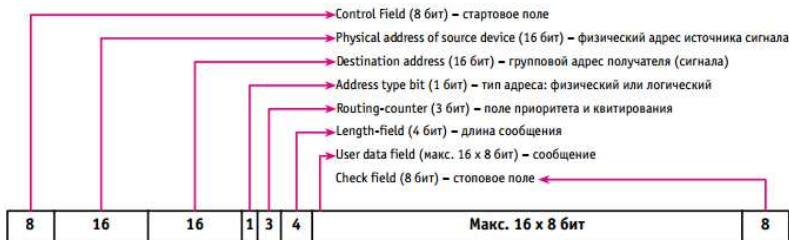


Рис. 2. Структура телеграммы

3. Интервал ожидания подтверждения. После передачи телеграммы узел ожидает, в течение минимального интервала времени $t_2 = 13$ бит времени, подтверждение принятия телеграммы. Если в течение 50 бит времени узел не получает телеграмму, то узлы распознают условие освобождения канала и начинают доступ к каналу, начиная с этапа 1 алгоритма. Узел, не получивший подтверждение, участвует в повторной передаче телеграммы ограниченное число раз (см. ниже счётчик передач).

4. Передача подтверждающей телеграммы. Узел-получатель телеграммы проверяет телеграмму при помощи контрольного байта, чтобы проверить корректность полученных данных, и отправляет соответствующие подтверждение. Подтверждающая телеграмма состоит из 8 бит информации и передаётся в составе 1 пакета.

Важными при анализе алгоритма асинхронного множественного доступа узлов к разделяемой среде передачи являются анализ ситуаций, при которых возможны столкновения телеграмм (коллизии) при передаче, и механизмы их предотвращения. Механизм избегания коллизий алгоритма основан на разделении двух физических уровней канала передачи: доминантного (приоритетного) – уровень логического 0 и рецессивного (неприоритетного) – уровень логической 1 [1]. После очередной передачи множество узлов стремится получить доступ к каналу и начинает одновременную передачу телеграммы (этап 2 алгоритма, см. рис. 1). Каждый узел передаёт бит служебной информации – приоритета и прослушивает среду передачи. Если узлом был передан неприоритетный бит 1, а принят из канала связи приоритетный бит 0, то узел прекращает передачу, в ином случае – продолжает её. Таким образом, узлы разделяют среду передачи по приоритетам. Специфика соперничества по приоритетам заключается в том, что узлы с телеграммами соперничают только в течение конкретных бит приоритета, определяющих лишь три уровня приоритета (низкий, высокий, тревога), и не соперничают битами другой служебной и пользовательской информации. Таким образом, соперничество узлов с равным уровнем приоритета за доступ и передачу по каналу завершается «ничьей», и узлы продолжают передавать данные, следствием чего является коллизия, распознаваемая (в зависимости от опции обнаружения коллизии узла) в конце передачи телеграммы. Вне зависимости от того, обнаружена коллизия или нет, по истечении интервала ожидания подтверждения (этап 3 алгоритма), равного интервалу контроля состояния канала t_1 , узлы, не получив подтверждения, будут повторять передачу определённое количество раз (конфигурируемая величина счётчика передачи, по умолчанию равная 3), каждый раз при этом попадая в коллизии. Успешно будет передана лишь та телеграмма, для которой задано большее количество попыток передачи, при условии, что в канале за этот интервал времени отсутствуют другие соперники с данным уровнем максимального приоритета. Очевидным является тот факт, что выделение индивидуальных приоритетов в рамках сетевого канала позволило бы решить проблему разделения канала на основе приоритетов. Проблема разделения канала может наблюдаться и для узлов с различным уровнем приоритета в условиях, не соответствующих требованиям проектирования длинной линии связи, когда наблюдается временное запаздывание между реально отправляемым и принимаемым битами.

Другим важным фактором, влияющим на временные задержки передачи, помимо рассмотренных задержек на передачу в условиях соперничества, коллизии и ожидания менее приоритетными узлами очереди передачи, является тип подтверждающей телеграммы, определяющий необходимость принятия дополнительных мер для обеспечения повторной доставки информации. Существуют три типа подтверждающей телеграммы:

- Тип NAK – информационная телеграмма принята с ошибками. Принятие такого подтверждения требует повторной передачи исходной телеграммы в рамках нового цикла соперничества после определения свободного канала.

- Тип BUSY – шина занята. Узел-отправитель телеграммы ожидает освобождения канала (этап 1) и осуществляет повторную передачу телеграммы, соперничая с другими узлами канала.

- Тип ACK – информационная телеграмма принята успешно. Узел-отправитель удаляет исходную телеграмму.

В случае когда узел не получает подтверждения в течение времени освобождения канала t_1 , то узел осуществляет повторную передачу, ограниченную числом раз счётчика передачи.

В работе был представлен анализ протокола множественного доступа узлов к разделяемой среде и передачи технологической информации в промышленной сети EIB/KNX. Результаты проведенного анализа будут использованы для разработки имитационной модели алгоритма доступа и исследования его количественных характеристик при различной загруженности сетевого канала. Сравнение полученных результатов с известными протоколами позволит определить эффективность применения алгоритма в условиях различной нагрузки на канал сети и обоснованно применять его в новых решениях децентрализованной автоматизации для организации взаимосвязи датчиков, контроллеров и исполнительных механизмов.

Библиографический список

1. Дитрих Д., Кастнер В., Саутер Т., Низамутдинов О. EIB – Система автоматизации зданий: пер. с нем. / под ред. О.Б. Низамутдинова, М.В. Гордеева; Перм. гос. техн. ун-т. – Пермь, 2001. – 378 с.

2. Кремлев А.С., Титов А.В., Щукин А.Н. Проектирование систем интеллектуального управления домашней автоматикой. Элементы теории и практикум: учеб. пособие. – СПб.: Изд-во НИУ ИТМО, 2014. – 96 с.

3. Даденков С.А., Кон Е.Л. Анализ моделей и методов агентного и дискретно-событийного имитационного моделирования // Известия СПбГЭТУ «ЛЭТИ». – 2015. – № 5. – С. 35–41.

4. Даденков С.А., Кон Е.Л. Метод расчета числа узлов-соперников за канал в сети со случайным множественным доступом, построенной на основе нейрочипов // Нейрокомпьютеры: разработка, применение. – 2015. – № 10. – С. 26–32.

5. Кокоулин А.Н. Разработка эффективных методов надежного хранения и передачи многомерных массивов графических данных в информационно-управляющих системах // Нейрокомпьютеры: разработка, применение. – 2013. – № 11. – С. 70–75.

6. Кокоулин А.Н. Методы обработки и хранения изображений в распределенных отказоустойчивых системах // Сб. тр. XII Всерос. совещ. по проблемам управления ВСПУ-2014. – М.: Изд-во ИПУ РАН, 2014. – С. 7442–7448.

7. Даденков С.А., Кон Е.Л., Южаков А.А. Алгоритм проектирования промышленной информационно-управляющей сети требуемой производительности (на примере технологии LonWorks) // Вестник Поволж. гос. технолог. ун-та. Сер. Радиотехнические и инфокоммуникационные системы. – 2014. – № 5(24). – С. 35–43.

8. Даденков С.А., Кон Е.Л. Подход к построению аналитической модели информационно-управляющей сети LonWorks на основе нейрочипов // Нейрокомпьютеры: разработка, применение. – 2013. – № 11. – С. 64–69.

9. Даденков С.А., Кон Е.Л. Оценка степени влияния некоторых факторов на производительность LonWorks-сети // Образовательные ресурсы и технологии. – 2014. – № 2(5). – С. 72–76.

10. Даденков С.А., Кон Е.Л. Исследование производительности алгоритма доступа к среде predictive p-persistent CSMA протокола // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2012. – № 6. – С. 217–230.

МОДЕЛЬ ФУНКЦИОНИРОВАНИЯ ТРАНСПОРТНОЙ ПОДСИСТЕМЫ РАСПРЕДЕЛЁННОЙ ИНФОРМАЦИОННО-УПРАВЛЯЮЩЕЙ СИСТЕМЫ

Студент А.И. Горожанкин

Пермский национальный исследовательский
политехнический университет

В статье рассматривается влияние информационных задержек в распределённой информационно-управляющей системе. Предлагается имитационная модель транспортной подсистемы ИУС для выполнения оценки информационных задержек.

Ключевые слова: Fieldbus-сеть, имитационная модель, информационное запаздывание, сетевая система управления, LonWorks, алгоритмы доступа PpP-CSMA, промышленные информационно-управляющие системы.

MODEL FUNCTIONING OF THE TRANSPORT SUBSYSTEM NETWORKED CONTROL SYSTEM

Student A.I. Gorozhankin

Perm National Research Polytechnic University

This article examines the impact of information delays in distributed control system. Proposed simulation model transport subsystem DCS to perform the evaluation of information delays.

Keywords: Fieldbus-network simulation model, information delay, network the control system, LonWorks, access method PpP-CSMA, networked control system.

В современном мире роль автоматизации очень важна. Основной целью автоматизации технологических процессов и технических объектов является повышение эффективности производственного процесса. Эффективность характеризуется увеличением объёмов и качества выпускаемой продукции при определённых экономических затратах на этапах проектирования и эксплуатации автоматизированной системы. Основой автоматизированной системы, призванной обеспечить заданные технические характеристики (надёжности, быстродействия, безопасности), является информационно-управляющая система (ИУС). С ростом количества датчиков, увеличением площади территории, на которой расположена автоматизированная система, и усложнением алгоритмов управления становится эффективным применение распределённых ИУС, т.е. систем, которые состоят из

совокупности контроллеров и модулей обработки, ввода-вывода, размещенных на большие расстояния [1–5]. Одной из распространенных концепций построения децентрализованных ИУС является подход на основе FieldBus-технологии, определяющей построение системы автоматизации на базе промышленной шины передачи данных (промышленной сети).

Сложность корректной реализации ИУС связана с особенностями реализации составляющих её контуров контроля и управления процессами в рамках промышленной сети. Реализация информационного взаимодействия распределенных элементов системы управления (датчиков, регуляторов и объекта управления) характеризуется информационным запаздыванием (рис. 1) [6–10], негативно влияющим на качество управления. Запаздывание датчика-регулятора вносится техническими средствами канала передачи информации. Вычислительное запаздывание вносится алгоритмом обработки технологической (измерительной) информации и расчёта управляющих воздействий. Запаздывание регулятора-объекта вносится техническими средствами канала передачи информации к исполнительному механизму [11].

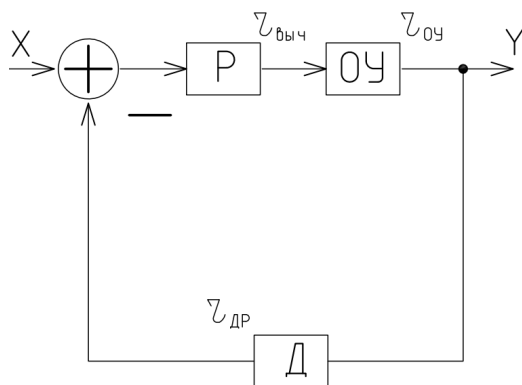


Рис. 1. Схема системы управления: ОУ – объект управления; Р – регулятор, Д – датчик

Указанные виды информационного запаздывания (см. рис. 1) являются результатом реализации дисциплин взаимодействия элементов сетевой системы управления. Проблема качества управления состоит в отличии реальных значений информационного запаздывания от принимаемых на этапе синтеза системы управления. Отличие

обусловлено, как правило, полным отсутствием принятия во внимание возможного отклонения величины запаздывания от планируемого, а также отсутствием методов оценки запаздывания. Результатом данной проблемы является несоответствие характеристик качества проектируемой системы требованиям. Решение данной проблемы на этапе эксплуатации автоматизированной системы определяет ряд экономических и трудовых затрат, связанных с необходимостью применения дорогостоящих аппаратно-программных средств количественной оценки быстродействия (задержек) промышленной сети и перепроектирования системы управления и/или сети. Исходя из представленного, целесообразным является решение проблемы на этапе планирования FieldBus-сетей. Это позволит определить через задержки передачи и обработки технологической информации в сетевой системе управления величины запаздываний и определить настройки элементов системы (регулятора, датчика).

Для решения поставленной задачи в работе анализируется распределённая промышленная шина LonWorks. Для разработки модели функционирования сети и анализа её временных характеристик используется система имитационного моделирования (AnyLogic), что позволит учесть большое количество особенностей функционирования протоколов для получения адекватных результатов.

Наиболее важной подсистемой промышленной сети, вносящей запаздывание и влияющей на качество управления, является транспортная подсистема.

Подсистема отвечает за организацию информационного взаимодействия элементов сетевой системы управления. Информационное взаимодействие производится путём подготовки технологической информации, получением доступа элемента системы управления к сетевому каналу и передачи. Доступ и передача производятся согласно протоколу случайного множественного доступа (predictive p-persistent CSMA) узлов к сети (рис. 2).

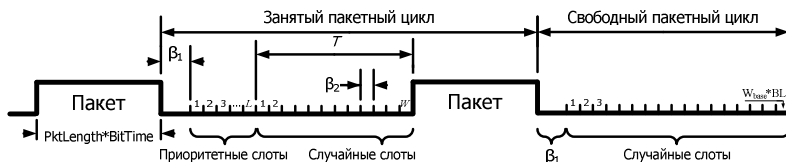


Рис. 2. Временная диаграмма работы алгоритма доступа

Этапы алгоритма доступа (см. рис. 2):

1. Проверка активности канала. В самом начале пакетного цикла узел проверяет активность передачи в канале. Если канал свободен и узел имеет пакет для передачи, то начинает реализацию процедуры доступа к каналу.

2. Межпакетный интервал. Для того чтобы определить освобождение канала после передачи, каждый узел сети должен выждать минимальный межпакетный интервал β_1 , значение которого определяется типом используемой среды передачи.

3. Приоритетный интервал. Данный интервал доступа используется узлами для передачи приоритетных пакетов. Узел выжидает фиксированное для него количество приоритетных временных слотов L , находящихся в диапазоне от 0 до 127, длительностью β_2 каждый [1].

4. Случайный интервал. Для того чтобы начать передачу пакета, канальный уровень должен обеспечить разность времени начала передачи узлами сети, чтобы избежать ситуации одновременной отправки пакетов – коллизии [1].

5. Передача пакета. Модель транспортной подсистемы построена путём создания агентов класса «Узел», структура которых определена алгоритмом доступа. Это позволяет воспроизвести процесс независимого коммуникационного доступа узлов к сети.

Модель алгоритма доступа и передачи представлена диаграммой состояний и переходов с тремя основными состояниями (рис. 3):

1. Состояние «controlOfDataToSend» – состояние контроля данных к отправке. Состояние выполняет функции по синхронизации и установке начальных характеристик пакетного цикла передачи.

2. Состояние «sendingPacket» – составное состояние отправки пакета, которое моделирует доступ узла к сетевому каналу для передачи пакета, согласно протоколу PpP-CSMA.

3. Состояние «receivingPacket» – состояние ожидания приёма пакета, которое реализует функции по завершении пакетного цикла. Вариантом завершения цикла является: успешная/неудачная передача пакета, завершение свободного пакетного цикла.

Основная часть модели транспортной подсистемы с алгоритмом доступа определена в состоянии передачи пакета «sendingPacket». В рамках данного состояния реализуется механизм алгоритма доступа PpP-CSMA (см. рис. 2). Соревнование за канал начинается в состоянии «contention», в котором инициализируется значение количества

элементарных слотов доступа. Спустя интервал задержки времени доступа (количество выбранных слотов доступа, умноженных на продолжительность элементарного тайм-слота) реализуется проверка занятости канала «transmission», по результатам которой узел переходит в состояние проигрыша соперничества «contentionOfLoss», если передача пакета уже ведется другими узлами, либо в состояние обработки коллизии «collisionOfTransmit», или в состояние обработки успешной передачи «successOfTransmission», в зависимости от того, произошла коллизия или нет.

Разработанная модель транспортной подсистемы промышленной сети РИУС позволяет оценить вероятностно-временные характеристики и информационное запаздывание системы. Это способствует корректному синтезу системы контроля и управления промышленной сети на этапе её планирования.

Библиографический список

1. Дитрих Д., Лой Д., Швайнциер Г.-Ю. Д 492 ЛОН-технология. Построение распределенных приложений: пер. с нем. / под ред. О.Б. Низамутдинова. – Пермь: Звезда, 1999. – 424 с.

2. Роль автоматизированных информационных систем и их влияние на эффективность работы организации [Электронный ресурс]. – URL: http://edu.dvgups.ru/METDOC/ITS/STRPRO/INF_ТЕН_STR/METHOD/SULDIN/frame/5_1.htm (дата обращения: 10.11.2015).

3. Промышленные Fieldbus-сети [Электронный ресурс]. – URL: <http://www.pcweek.ru/industrial/article/detail.php?ID=49447> (дата обращения: 10.11.2015).

4. Автоматизация технологических процессов [Электронный ресурс]. – URL: https://ru.wikipedia.org/wiki/Автоматизация_Технологических_процессов (дата обращения: 10.11.2015).

5. Денисенко В.В. Компьютерное управление технологическим процессом, экспериментом, оборудованием. – М.: Горячая линия-Телеком, 2009. – 608 с.

6. Даденков С.А., Кон Е.Л. Оценка степени влияния некоторых факторов на производительность LonWorks-сети // Образовательные ресурсы и технологии. – 2014. – № 2(5). – С. 72–76.

7. Даденков С.А., Кон Е.Л. Исследование производительности алгоритма доступа к среде predictive p-persistent CSMA протокола //

Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2012. – № 6. – С. 217–230.

8. Даденков С.А., Кон Е.Л., Южаков А.А. Алгоритм проектирования промышленной информационно-управляющей сети требуемой производительности (на примере технологии LonWorks) // Вестник Поволж. гос. технолог. ун-та. Сер. Радиотехнические и инфокоммуникационные системы. – 2014. – № 5(24). – С. 35–43.

9. Даденков С.А., Кон Е.Л. Анализ моделей и методов агентного и дискретно-событийного имитационного моделирования // Известия СПбГЭТУ «ЛЭТИ». – 2015. – № 5. – С. 35–41.

10. Даденков С.А., Кон Е.Л. Метод расчета числа узлов-соперников за канал в сети со случайным множественным доступом, построенной на основе нейрочипов // Нейрокомпьютеры: разработка, применение. – 2015. – № 10. – С. 26–32.

11. Крушель Е.Г., Степанченко И.В. Информационное запаздывание в цифровых системах управления: монография. – Волгоград: Изд-во ВолгГТУ, 2004. – 124 с.

АНАЛИТИЧЕСКОЕ И ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ САМОМАРШРУТИЗИРУЮЩЕГОСЯ НЕЙРОСЕТЕВОГО АНАЛОГО-ЦИФРОВОГО ПРЕОБРАЗОВАТЕЛЯ

Магистрант А.В. Елтышев

Пермский национальный исследовательский
политехнический университет

В статье рассматриваются аналитическая и имитационная модели самомаршрутизирующегося нейросетевого аналого-цифрового преобразователя. Обсуждаются результаты, полученные с помощью этих моделей, и принципы их построения в зависимости от различных параметров преобразователя.

Ключевые слова: самомаршрутизация, нейронная сеть, аналого-цифровой преобразователь, имитационная модель, аналитическая модель, вероятность отказа.

SELF-ROUTING NEURAL ANALOG-TO-DIGITAL CONVERTER ANALYTIC AND IMITATION MODELING

Master Student A.V. Eltyshv

Perm National Research Polytechnic University

In this article analytic and imitation models of self-routing neural analog-to-digital converter are considered. Results with receiving those models and formation principles with different parameters are discussed.

Keywords: self-routing, neural network, analog-to-digital converter, imitation model, analytic model, probability of failure.

Цели данной работы – построение аналитической и имитационной моделей упрощенной нейронной сети, применяющейся в самомаршрутизирующемся нейросетевом аналого-цифровом преобразователе, а также сравнение результатов моделирования, полученных с помощью данных моделей, для определения степени их соответствия.

Аналого-цифровой преобразователь (АЦП) является одной из составляющих любой системы автоматического управления, так как для получения данных об объекте необходимо привязать показания датчиков, как правило, аналоговых к управляющему микроконтроллеру, который может оперировать только цифровыми данными. Одной из приоритетных задач в таком случае становится повышение отказоустойчивости АЦП с целью обеспечения надежности всего комплекса автоматического управления, что наиболее важно для уда-

ленных автономных объектов управления (космический спутник или автоматическая нефтедобывающая станция и т.п.) или объектов управления повышенной опасности (ядерные электростанции, различные системы боевого назначения и т.п.). Одним из возможных и перспективных направлений в разработке отказоустойчивых АЦП является применение нейронных сетей (НС) [1]. Но для получения количественных оценок эффективности применения НС необходимо разработать модель, позволяющую рассчитывать значение вероятности отказа в зависимости от параметров применяемой НС.

На текущем этапе за основу модели взята упрощенная НС (рис. 1), состоящая только из основных нейронов (ОН). Это объясняется тем, что именно ОН в нейронной сети обеспечивают процесс проведения аналого-цифрового преобразования, а остальные нейроны имеют вспомогательный характер и отвечают за реализацию алгоритмов маршрутизации сигналов в НС [2].

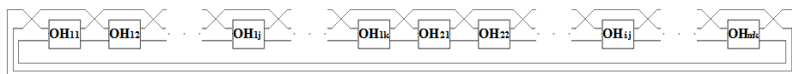


Рис. 1. Упрощенная структурная схема НС

Упрощенная НС представляет собой кольцо из последовательно соединенных ОН без дополнительных связей, при этом каждый нейрон может стать первым в формировании индивидуального АЦП (ИАЦП), т.е. измерять старший разряд входного сигнала, а к нему может присоединиться некоторое число ОН, чтобы получить ИАЦП соответствующей разрядности.

Чтобы получить общий вид зависимости вероятности отказа от параметров НС АЦП, было решено построить аналитическую модель на основе теории систем массового обслуживания (СМО). В ходе исследования были выделены основные параметры НС, которые могут повлиять на вероятность отказа системы в проведении измерения:

- количество нейронов в НС (n);
- количество входных сигналов, для которых необходимо формировать ИАЦП (N);
- минимальная разрядность входной заявки (q_{\min});
- максимальная разрядность входной заявки (q_{\max});
- интенсивность входного потока (λ);
- интенсивность выходного потока (μ);
- интенсивность выхода из строя нейронов ($\lambda_{\text{отк}}$).

Так как существует вероятность выхода из строя ОН без возможности его восстановления, то аналитическая модель представляет собой сеть массового обслуживания (СМО), состоящую из определенного количества СМО, число которых соответствует числу комбинаций количества отказавших нейронов и их расположения относительно друг друга. Сама система (СМО), граф которой имеет векторный вид, определяется состояниями S , в которых она может пребывать, при этом каждое состояние S соответствует количеству заявок k_m , требующих определенное количество ОН для их обработки $m \in [q_{\min} - q_{\max}]$. В силу неоднородности потока (входная заявка способна требовать произвольное число $m \in [q_{\min} - q_{\max}]$) и равновероятности входных заявок (поступление заявки с разной разрядностью в систему одинаково $P(m) = \frac{1}{q_{\max} - q_{\min} + 1}$), интенсивность входного потока заявок в начальном состоянии СМО S_0 будет иметь вид $\lambda_0 = NP(m) \frac{\lambda}{N}$, так как вероятности поступления заявок на измерение сигнала одинаковы для каждого входа. С ростом количества поступивших заявок интенсивность переходов между состояниями СМО будет снижаться, так как с одного входа может одновременно поступить только одна заявка. Интенсивность $\lambda_i = (N - k_m)P(m) \frac{\lambda}{N}$ представляет собой интенсивность перехода СМО из состояния S_i в смежное состояние во время поступления новой заявки при наличии в системе k_m заявок, уже поступивших в систему с других входов. Таким образом, количество состояний S будет ограничено с одной стороны количеством входов N , а с другой – количеством k_m поступающих заявок. Исходя из того, что нейроны расположены последовательно, то интенсивность обслуживания заявок (обратных переходов) определяется по формуле $\mu_i = \frac{k_m \mu_k}{m}$ за время измерения $t_{\text{из}k}$, где μ_k – интенсивность обслуживания k -го нейрона

$$\left(\mu_k = \frac{1}{t_{\text{из}k}} \right).$$

Входной поток заявок будет ограничен следующими свойствами: поступление заявок не зависит от времени (стационарный поток), свойства новой заявки не зависят от предыдущих (без последствия) и вероятность появления одновременно двух или более заявок на элементарном интервале времени бесконечно мала по сравнению с вероятностью появления одной заявки. Исходя из перечисленных свойств, входной поток НС является неоднородным простейшим пуассоновским потоком.

Расчет вероятности отказа СМО разработанной модели выполняется путем построения системы линейных алгебраических уравнений (СЛАУ) по полученному векторному графу, затем выводятся формулы для определения вероятности каждого состояния системы, в результате вероятность отказа в данном макросостоянии СеМО выражается суммой состояний, в которых происходит отказ в обслуживании, т.е. количество доступных ОН не удовлетворяет потребностям входных заявок для проведения измерения.

Поскольку масштаб построения графа и проведение расчетов с помощью СЛАУ с ростом значений начальных параметров резко возрастают, было решено построить модель СеМО в среде разработки AnyLogic и для проверки ее адекватности сравнить графики зависимости вероятности отказа от интенсивности входного потока, а также от интенсивности обслуживания с графиками, полученными вручную. Результаты показали, что обе модели идентичны друг другу, что позволяет с помощью модели, построенной в AnyLogic, исследовать большие СеМО и собирать статистику для аналитической модели НС АЦП даже при больших значениях входных параметров.

Среда AnyLogic для разработки масштабируемой аналитической модели была выбрана не случайно, а с учетом того, что параллельно в этой среде разрабатывается имитационная модель НС АЦП. Выбор этой среды для имитационного моделирования связан с рядом уникальных возможностей AnyLogic для реализации агентного имитационного моделирования, которое положено в основу имитационной модели НС АЦП, обзор и сравнение AnyLogic с другими комплексами приведены в работе [3].

Чтобы логика работы имитационной модели АЦП соответствовала работе реального самомаршрутизирующегося нейросетевого АЦП, необходимо прописать следующие алгоритмы:

- формирование ИАЦП;

- генерация входного потока заявок;
- обслуживание и удаление из системы заявок;
- выход ОН из строя.

Для их реализации наиболее удобным является агентный метод имитационного моделирования, анализ и сравнение которого с другими методами представлены в работе [4]. Структура имитационной модели, построенной с помощью данного метода моделирования, показана на рис. 2.

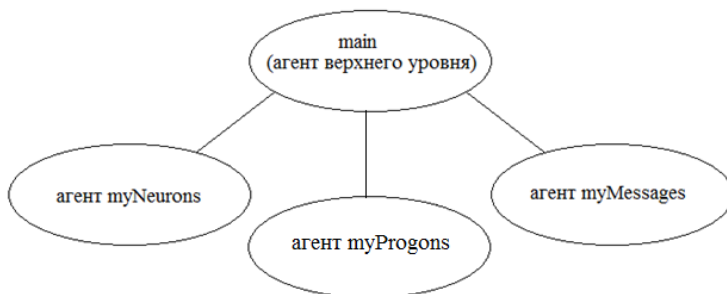


Рис. 2. Структура имитационной модели

Агенты представляют собой части модели, выполняющие прописанные функции и имеющие определенные параметры. Таким образом, имитационная модель состоит из агента `main`, который является агентом верхнего уровня, он содержит в себе агенты `myNeurons`, `myProgons` и `myMessages` и обеспечивает взаимодействие между ними. Кроме того, в этом агенте происходят генерация входных заявок со случайным количеством требуемых разрядов и запуск прогонов, функции которых описываются в агенте `myProgons`.

В агенте `myNeurons` заложены логика выхода из строя ОН и параметры, присущие этим нейронам:

- номер нейрона в НС (от 0 до n);
- исправность, начальное значение которого `true`, т.е. нейрон исправен;
- занятость, начальное значение которого `false`, т.е. нейрон свободен.

Таким образом, в начале каждого прогона мы имеем полностью исправную и свободную НС, в которую начинают попадать входные заявки с заданной интенсивностью.

Агент `myMessages` описывает логику формирования ИАЦП в НС: размещение заявки последовательно по исправным и незанятым нейронам. Дисциплина обслуживания заявки – система с отказом и ожиданием. Это означает, что в случае получения отказа (заявка не смогла разместиться в НС) входная заявка уходит в ожидание и через фиксированное время пытается снова расположиться в НС до тех пор, пока она не сможет сформировать ИАЦП или не истечет время ее «жизни», которое зависит от разрядности этой заявки и интенсивности обслуживания. Также может произойти поломка ОН, который входит в сформированный ИАЦП, тогда данный ИАЦП «разрушается», а заявка отправляется в ожидание. По истечении времени «жизни» заявка удаляется независимо от того, находится она в обслуживании или в ожидании.

Агент `myProgons` является вспомогательным и выполняет функцию сброса всех счетчиков и параметров в начальные состояния при достижении временем моделирования заданного значения времени одного прогона. Это позволяет получать более быстро (автоматизация процесса перебора значений начальных параметров) и точно (с увеличением числа прогонов) усредненные статистические данные (классический метод Монте-Карло).

Расчет вероятности отказа осуществляется путем отношения времени, при котором система была в состоянии отказа (хотя бы одна из входных заявок находилась в состоянии ожидания), ко времени моделирования каждого прогона. Результатом имитационного моделирования стало получение графиков зависимостей вероятности отказа от различных начальных параметров НС АЦП.

Сравнение результатов аналитического и имитационного моделирования НС АЦП показывает, что модели имеют очень близкий результат при одинаково заданных начальных параметрах, что указывает на адекватность получаемых данных.

В ходе проделанной работы был получен общий вид зависимостей вероятности отказа упрощенной НС самомаршрутизирующегося нейросетевого АЦП от начальных параметров, по которым можно определить степень зависимости вероятности отказа от этих параметров, и поставлены следующие задачи:

- внедрить в имитационную модель НС дополнительные связи между ОН;

- описать в имитационной модели нейроны дополнительного слоя НС (коммутаторы (Км)) [2] как агенты, смоделировав алгоритм маршрутизации сигналов внутри НС;
- проверить полученные моделированием значения, сравнив их с реальными вероятностями отказа нейросетевого АЦП;
- разработать методику расчета необходимого количества ОН и Км для построения АЦП, обладающего заданной отказоустойчивостью при известных начальных параметрах.

Библиографический список

1. Посягин А.И., Южаков А.А. Структура аналого-цифрового преобразователя на основе нейронной сети // Прикладная фотоника. – 2012. – № 1–4. – С. 223–238.
2. Посягин А.И., Южаков А.А. Самомаршрутизирующийся аналого-цифровой преобразователь на основе двухслойной нейронной сети. Нейрокомпьютеры: разработка, применение. – 2013. – № 11. – С. 076–081.
3. Даденков С.А., Кон Е.Л. Анализ моделей и методов агентного и дискретно-событийного имитационного моделирования // Известия СПбГЭТУ «ЛЭТИ». – 2015. – № 5. – С. 35–41.
4. Даденков С.А., Кон Е.Л. Подход к построению аналитической модели информационно-управляющей сети LonWorks на основе нейрончиков // Нейрокомпьютеры: разработка, применение. – 2013. – № 11. – С. 64–69.

О ВНЕДРЕНИИ ДОВЕРЕННОЙ ИНФОРМАЦИОННОЙ СРЕДЫ ГОРОДСКОЙ ИНФРАСТРУКТУРЫ НА ОСНОВЕ ПРОГРАММНО-АППАРАТНОГО КОМПЛЕКСА «СРЕДСТВО ОБЕСПЕЧЕНИЯ ДОВЕРЕННОГО СЕАНСА "МАРШ!"»

**Студентка Е.Е. Журилова,
канд. техн. наук, доцент А.С. Шабуров**
Пермский национальный исследовательский
политехнический университет

В данной статье рассмотрена проблема обеспечения информационной безопасности городской инфраструктуры за счет внедрения доверенной информационной среды. Приведена обобщенная структура доверенной аппаратно-программной среды. Выявлена проблема доверия к системам безопасности из-за массового применения технологий и средств управления иностранного производства. Рассмотрены функциональные особенности, а также возможности внедрения программно-аппаратного комплекса «Средство обеспечения доверенного сеанса "МАРШ!"» для создания доверенной информационной среды городской инфраструктуры.

Ключевые слова: критически важный объект, доверенная информационная среда, недеklarированная возможность, базовый доверенный модуль, программно-аппаратный комплекс.

ON THE IMPLEMENTATION OF THE TRUSTED INFORMATION ENVIRONMENT URBAN INFRASTRUCTURE BASED ON PAC SODS "MARSH!"»

**Student E.E. Zhurilova,
Ph.D. in Technical Sciences, Associate Professor A.S. Shaburov**
Perm National Research Polytechnic University

In this article was reviewed the problem of information security in urban infrastructure through the introduction trusted information environment. It shows generalized structure of trusted hardware and software environment. The problem of trust in the security system because of the massive use of technologies and management of foreign manufacture was revealed. It was considered the functional features, and also the possibility of the introduction of hardware-software complex "means providing a trusted session" MARSH!" to create a trusted information environment of urban infrastructure.

Keywords: critical object, trusted information environment, undeclared opportunity, basic trusted module, software and hardware system.

В настоящее время в связи с урбанизацией городов и поселений появляются и развиваются новые отрасли науки, техники, социальных отношений [1]. В то же время активные темпы информатизации

городской инфраструктуры значительно усложнили решение ряда проблем обеспечения функционирования информационно-управляющих и телекоммуникационных систем.

Усложнение и расширение областей использования различных технических и программных решений в инфраструктуре города обуславливают и значительно увеличивают вероятности угроз безопасности информации для управляющих систем, а также степень опасности последствий негативного информационного воздействия.

Особенно важно создание безопасной информационной среды для систем управления критически важными инфраструктурными объектами. К критически важным объектам, безусловно, можно отнести те, которые обеспечивают функционирование городской инфраструктуры, в том числе электрические и тепловые сети, коммуникации, транспортные и т.п.

Воздействие на управляющие системы критически важных объектов будет одним из наиболее эффективных по результатам негативного и разрушительного влияния в ходе ведения кибернетических и информационных войн. При этом подготовка к проведению подобных информационных атак может вестись заблаговременно, посредством внедрения вредоносных программ для получения потенциальной возможности контроля и уничтожения важных систем и объектов.

Повышение уровня информационной безопасности управляющих и телекоммуникационных систем зависит от множества факторов, среди которых внедрение и поддержание доверенной информационной среды.

В общем случае доверие к информационной среде подразумевает уверенность в том, что продукт информационных технологий соответствует требованиям безопасности информации. Это может быть обеспечено как на основе априорных знаний или опыте эксплуатации продукта, или системы, так и с помощью их специальных исследований и определения свойств, существенных для сохранения состояния безопасности.

В настоящий момент можно выделить три класса угроз для информационных систем, без внедренной системы доверия и защиты:

1. Сбой работы информационной системы в случае умышленно заложенных незапротоколированных функций в системотехнике или при ошибках разработчиков системотехники.

2. Несанкционированная подмена данных без вмешательства оператора за счет недокументированных специальных функций, встроенных в систему управления.

3. Несанкционированная подмена данных с участием оператора или других лиц.

Задача снижения вероятности проявления подобных угроз подразумевает использование только соответствующих требованиям безопасности продуктов информационных технологий. Возможность обеспечения необходимой функциональности и доверия к активам, с одной стороны, а также необходимый уровень защиты от угроз безопасности информации, с другой, делает доверенные информационные системы незаменимыми для городской инфраструктуры.

Обобщенная структура доверенной аппаратно-программной среды представлена на рис. 1.

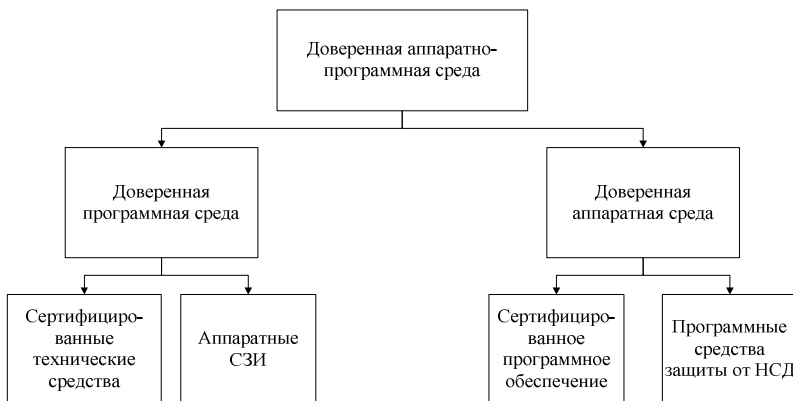


Рис. 1. Структура доверенной аппаратно-программной среды

Сегодня разработано множество разнообразных программно-технических решений с различным функционалом как российского, так и зарубежного производства.

Однако использование зарубежных средств создания доверенной информационной среды делает актуальным вопрос доверия к элементной базе и может ставить под угрозу защищенность такой системы. По оценкам специалистов, 90 % аппаратно-программных средств и операционных систем, используемых в городских инфраструктурах, разработаны и произведены за рубежом и поставляются

в Россию без исходного кода. Это обстоятельство делает невозможным проверку таких комплексов на наличие недекларированных возможностей (НДВ) [2].

Наличие исходного кода и возможность проведения сертификации также не гарантируют полного отсутствия угроз. В испытательных лабораториях, как правило, строятся тестовые примеры, для которых уровень покрытия 75 % является показателем сравнительно высокого качества проводимого тестирования. Проведение же более тщательных проверок требует большого количества времени, что приводит к устареванию оборудования и утрате конкурентоспособности [3].

Это подводит к необходимости использования отечественных средств создания доверенной информационной среды. В настоящее время существуют компании, поставляющие продукцию, способную составить конкуренцию зарубежным программно-техническим средствам, а возможность проверки исходных кодов на НДВ является еще одним аргументом в пользу импортозамещения.

Так, положительным примером практического создания и использования доверенной информационной среды является совместное решение ФГУП РСВО и НТЦ «НАО-Про», реализованное в г. Санкт-Петербурге и в г. Севастополе.

Для обеспечения доверенной загрузки целостной информационной среды при удалённом доступе и изолированном сетевом соединении с сервером приложений, например, используется система построения доверенной среды «ПОСТ».

Еще одной российской компанией «Элвис-плюс» была разработана технология базового доверенного модуля (БДМ). БДМ-технология предназначена для построения доверенной вычислительной среды в различных информационных системах [5].

В 2014 г. на выставке по информационной безопасности «InfoSecurity Russia» компания «Setec» продемонстрировала линейку продуктов «Тринити», предназначенную для построения IT-инфраструктуры, защищенных «тонких клиентов». Главным преимуществом разработанного программного обеспечения является возможность работы пользователей в доверенной информационной среде.

Еще одним продуктом для обеспечения работы в доверенной среде является программно-аппаратный комплекс (ПАК) «Средство обеспечения доверенного сеанса (СОДС) «МАРШ!».

СОДС «МАРШ!» выглядит как USB устройство, однако содержит внутри активное микропроцессорное устройство с многоконтурной криптографической подсистемой, проверенной защищенной операционной системой «Linux», браузером, специальной подсистемой управления к памяти и многим другим, что позволяет обеспечить полноценную криптографическую защиту информации во время доверенного сеанса связи [6]. Это позволяет безопасно работать с доверенными сервисами с любого компьютера, избежав при этом ограничений, связанных с необходимостью специального оборудования компьютера, а также ограничений, накладываемых организационными требованиями в отношении программного обеспечения, прав доступа, времени работы и т.д.

Принцип работы данного комплекса представлен на рис. 2. Для начала работы пользователь загружает «МАРШ!», подготовленный для использования как загрузочное устройство, обеспечивая тем самым доверенную среду. Следом запускается браузер и соответствующее программное обеспечение (стрелка 1). Устанавливается доверенная связь с сервером центральной информационной системы (стрелка 2). Сервер ИС выполняет авторизацию пользователя и проверяет право доступа к сервисам ИС, после этого идет соединение с требуемым сервисом ИС (стрелка 3).



Рис. 2. Структурная схема работы ПАК СОДС «МАРШ!»

Доверенный сеанс в браузере обеспечивает защищённый информационный обмен, гарантируя достоверность передаваемой информации, что, в свою очередь, позволяет выполнить требования Федерального закона № 63 «Об электронной подписи» [7].

Загрузка операционной системы производится из раздела памяти, защищенного от записи, не задействовав тем самым жесткий диск в процессе загрузки. Загруженная конфигурация ОС ограничивает действия пользователя внутри данной системы, делает недоступными ее органы управления, изолирует рабочую среду от посторонних сетевых соединений, завершает сеанс связи сразу после завершения работы в браузере.

Доверенный сеанс связи с сервером устанавливается через VPN-шлюз, защищенное соединение на основе криптографических алгоритмов, используя закрытые ключи и сертификаты, хранящиеся в защищенной памяти «МАРШ!».

Если рассматривать данную систему с позиции управления доступом, то «МАРШ!» выглядит как память, состоящая из разделов ReadOnly (содержит ПО), ReadWriteHidden (содержит обновления и дополнения и информацию о VPN), AddOnly (содержит журнал событий) и стандартных разделов общего доступа ReadWrite. Количество разделов может быть разным, деление осуществляется на производстве и не доступно пользователям.

ПАК СОДС «МАРШ!» имеет преимущество в первую очередь перед импортными системами, поскольку производится и сертифицируется в России и по российским стандартам. Среди конкурентных же продуктов его выделяют возможность доступа через одну точку авторизации, мобильность, достаточный уровень защиты и приемлемую цену.

Таким образом, с ростом и урбанизацией городов возникает необходимость обеспечения информационной безопасности и создания доверенной информационной среды с использованием отечественных аппаратно-программных комплексов. Программно-аппаратный комплекс «Средство обеспечения доверенной связи “МАРШ!”» является одним из возможных вариантов, удовлетворяющих требованиям безопасности информации.

Библиографический список

1. Чернова Е.Б. История развития социологического направления урбанистики. – СПб.: Реферат, 2012.
2. Хабибуллин И.В. Основные проблемные вопросы создания доверенной программно-аппаратной среды для АСУ органов военного и государственного управления // Вопросы кибербезопасности. – 2014. – № 3(4).

3. Мэн Д. Russian researchers expose breakthrough U.S. spying program [Электронный ресурс]. – URL: <http://www.reuters.com/article/2015/02/16/usa-cyberspying-idUSL1N0VN15J20150216?feedType=RSS&feedName=marketsNews> (дата обращения: 06.11.2015).

4. Бородакий Ю.В. Доверенная среда – основа гарантированной безопасности [Электронный ресурс] // Информационная безопасность. – 2013. – № 2. – URL: <http://www.itsec.ru/articles2/Oborandteh/doverennaya-sreda--osnova-garantirovannoy-bezopasnosti> (дата обращения: 03.11.2105).

5. Элвис-Плюс [Электронный ресурс]. – Зеленоград, 2010. – URL: <http://elvis.ru/products/bdm/> (дата обращения: 03.11.2015).

6. СОДС «МАРШ!» [Электронный ресурс]. – URL: <http://www.sodsmarsh.ru/pages/page/show/2> (дата обращения: 03.11.2105).

7. Федеральный закон от 06.04.2011 № 63-ФЗ (ред. от 28.06.2014) «Об электронной подписи» (с изм. и доп., вступ. в силу с 01.07.2015) // Доступ из справ.-правовой системы КонсультантПлюс.

ВОПРОСЫ ИНТЕГРАЦИИ СМАРТ-КАРТ В ОС WINDOWS

Канд. техн. наук, доцент А.Н. Кокоулин,
студенты С.Д. Зее, А.И. Тур

Пермский национальный исследовательский
политехнический университет

В данной статье рассматривается проблема аутентификации, которая на сегодняшний день становится актуальной во многих фирмах и удалённых сервисах. Одним из наиболее удобных способов является использование смарт-карт. Этот способ позволяет не только подключаться к учётным записям Windows, но и защищать информационные ресурсы организации.

Ключевые слова: смарт-карта, аутентификация, Windows.

SMART-CARDS INTEGRATION IN WINDOWS OS

Ph.D. in Technical Sciences, Associate Professor A.N. Kokoulin,
Students S.D.Zee, A.I. Tur

Perm National Research Polytechnic University

The headline of the article is considering of the problem of authentication which becomes acute in many firms and remote services. One of the most convenient methods is using a smart cards. This method allows to be connected to accounting entries of Windows and to protect information resources of the organization.

Keywords: smart card, authentication, Windows.

На сегодняшний момент проблема аутентификации становится всё более актуальной. Старые способы аутентификации (логин/пароль) уже устарели, за всё время их использования злоумышленники разработали различные методы атаки и получения доступа. Более надёжные способы аутентификации требуют больших знаний пользователя и не всегда целесообразно их использование. Использование смарт-карт совмещает надёжность и простоту для пользователя.

Смарт-карта представляет собой пластиковую пластину со встроенной микросхемой, позволяющей защищенно хранить информацию о владельце, используемую в процессе аутентификации.

В зависимости от типа карты для её считывания применяются различные устройства. Наиболее распространённый тип смарт-карт – контактные. Для них используются специальные считыватели, позволяющие считать информацию с чипа памяти.

Обладая считывателем и смарт-картой, пользователь может использовать их:

- для входа в Windows по смарт-карте;
- защиты информации на компьютере;
- защиты информации на компьютерах малых рабочих и корпоративных сетей;
- защиты почтовой переписки;
- организации защищенного доступа к рабочим приложениям;
- организации защищенного доступа к интернет-сайтам. [5]

Кроме того, смарт-карты также используются в сотовой связи (SIM-карты) и банковской сфере (банковские карты). Все они схожи по реализации (оптимизированный для криптографии микроконтроллер с повышенным уровнем безопасности) и главной функции (подтверждение подлинности пользователя, запрашивающего доступ, – аутентификация).

Для начала попробуем классифицировать аутентификацию с помощью смарт-карт. Для этого будем использовать три критерия: степень автоматизации, приоритет использования, фактор авторизации.

Если классифицировать данный метод аутентификации по степени автоматизации, то можно рассматривать его как полностью автоматизированный (со стороны системы), так как он не подразумевает дополнительных действий от администратора в случае подачи заявки на вход в систему.

Если же рассматривать по приоритету использования, то такой способ в случае интеграции в безопасность операционной системы, должен являться основным. Это обусловлено тем, что владение смарт-картой является более безопасным, чем просто знание пароля. Однако данный способ имеет недостатки из-за возможных технических проблем при считывании карты или же её утери. Поэтому резервным методом аутентификации стоит оставлять ввод пары логин/пароль.

Среди факторов авторизации выделяют четыре группы – фактор знания («то, что ты знаешь»), вещественный фактор («то, чем ты владеешь»), биофактор («то, что является частью тебя») и социальный фактор («те, кто знают тебя»). В чистой своей форме аутентификация с помощью смарт-карт использует вещественный фактор, т.е. подразумевает наличие у пользователя вещи, удостоверяющей его личность. Однако иногда для повышения уровня защищённости вход

по смарт-карте дополняют обязательным вводом пароля или PIN-кода, что уже будет сочетанием вещественного фактора и фактора знаний.

Смарт-карта представляет собой пластиковую карту с интегрированным в неё процессором и памятью. Общая структура смарт-карты представлена на рис. 1.

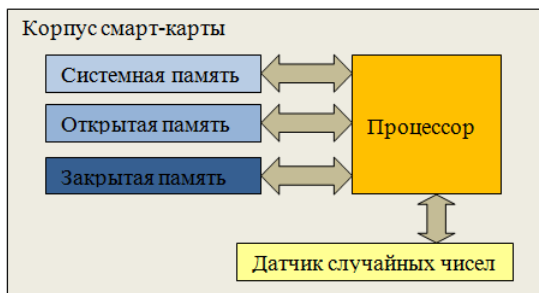


Рис. 1. Структурная схема смарт-карты

В зависимости от того, какой чип установлен на карточке, смарт-карты делятся на несколько видов: контактные, бесконтактные и гибридные. Контактные требуют прямого контакта со считывателем, а вот бесконтактные используют технологию RFID, поэтому достаточно их просто близко поднести к устройству. Гибридные же смарт-карты могут использоваться одинаково успешно любым из этих двух способов [2].

Чип смарт-карты обеспечивают следующие функции:

- хранение закрытых ключей цифровых сертификатов во внутреннем хранилище чипа смарт-карты;
- генерация ключевой пары (с этой целью в чипе смарт-карты содержится датчик случайных чисел, который генерирует при обращении ключевую пару. Закрытый ключ записывается в хранилище чипа смарт-карты и никогда не выходит за пределы памяти чипа. Открытый ключ передаётся процессору, который генерирует запрос на сертификат и посылает его за пределы смарт-карты, например, в центр выдачи цифровых сертификатов);
- аппаратную реализацию симметричных алгоритмов шифрования;
- аппаратную реализацию алгоритмов хеширования;
- аппаратную реализацию ассиметричных алгоритмов.

Для выполнения этих функций в распоряжении процессора карты есть следующие виды памяти:

- системная память – содержит файловую и операционную системы, в которой хранятся данные, необходимые для проверки правильности вводимых PIN-кодов и паролей администратора;

- открытая память – содержит данные, которые возможно прочитать без PIN-кода (имя смарт-карты, идентификатор смарт-карты и другие);

- закрытая память – содержит данные, доступ к которым возможен только по PIN-коду (например, закрытые ключи).

Считыватели смарт-карт, как и сами карты, тоже разделяют на контактные и бесконтактные. При этом почти все эти устройства способны как считывать, так и записать информацию на карту в случае, если управляющий пользователь имеет такие полномочия. Также считыватели можно разделить на стационарные и носимые. Стационарные могут быть выполнены как в виде отдельного внешнего устройства, так и быть встроенными в устройства ввода (например, клавиатуру ПК). Носимые имеют компактные размеры и, как правило, представляют собой контактную рамку.

Для использования смарт-карт необходима тесная интеграция в систему. Примером такой интеграции является программа *SecureSmart*. Эта программа позволяет изменить стандартный вход в систему Windows, поддерживается как локальный вход, так и доменный с архитектурой «клиент-сервер», проверка подлинности осуществляется с помощью смарт-карт или биометрических считывателей (например, сканер отпечатка пальцев), дополнительно используется PIN-код. Она подменяет файл реестра `gina` (`HKEY_LOCAL_MACHINE \ SOFTWARE \ Microsoft \ Windows NT \ CurrentVersion \ Winlogon Value = GinaDLL REG_SZ`) и модифицирует окно входа в систему, добавляя свой собственный интерфейс [3]. Схема взаимодействия и функциональная структура представлены на рис. 2.

Работа в домене требует особой защиты соединения и аутентификации, так как передача зачастую идёт по незащищённым участкам сети. Такую работу обеспечивает протокол Kerberos. Он реализуется следующим способом (рис. 3):

1. Аутентификация со смарт-картой и PIN-кодом (закрытый ключ+сертификат);

2. Запрос TGT к контроллеру домена (DC) с именем отправителя и меткой времени, которые подписаны с использованием ключа и копией сертификата;

3. Для проверки запроса и цифровой подписи центр распределения ключей (KDC) проверяет сертификат отправителя и запрашивает у AD соответствие между сертификатом и пользователем Windows. Если соответствие подтверждено, то генерируется новый сеансовый ключ и TGT для учётной записи;

4. DC отправляет TGT отправителя с защищенной копией сеансового ключа;

5. Для получения копии сеансового ключа отправителя от пакетов, полученных в 4-м пункте, используется закрытый ключ отправителя [1].

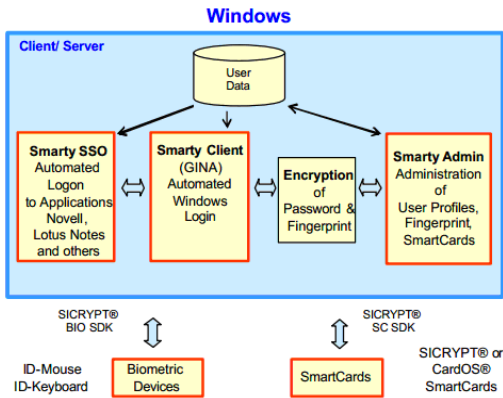


Рис. 2. Функциональная структура Sicyprt SmartY

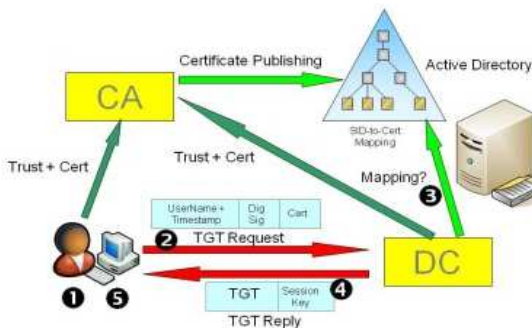


Рис. 3. Аутентификация смарт-карт через Kerberos

Смарт-карты хотя и являются одним из самых надёжных методов аутентификации, однако даже они имеют множество уязвимостей. Существуют три основных вида атак на смарт-карты. Атаки на социальном уровне – это атаки на людей, работающих со смарт-картами, независимо от фазы жизненного цикла смарт-карты.

Атаки на физическом уровне:

1. Пассивные атаки. К ним относятся атаки, основанные на наблюдении за каким-либо физическим параметром реализации, промодулированным ключевой информацией;

2. Активные атаки без проникновения. К этому классу относятся атаки, основанные на генерации случайных аппаратных ошибок во время исполнения криптоалгоритма и последующего их анализа;

3. Активные атаки с проникновением. Здесь речь идет об атаках на микросхемы с проникновением в саму микросхему.

Атаки на логическом уровне: осуществляются за счет классического криптоанализа, известных неисправностей операционной системы смарт-карты, «троянских коней» в исполняемом коде приложений смарт-карты. По статистике данные атаки являются наиболее успешными [4].

Для защиты от вышеописанных угроз применяется множество методов. Вот некоторые из них:

- криптографические сопроцессоры;
- датчики высокой и низкой частоты напряжения, температуры, света;
- аппаратные средства защиты от вскрытия;
- аппаратный контроль целостности EEPROM;
- аппаратный генератор случайных чисел;
- аппаратные средства активного противодействия инженерным атакам;
- программные средства защиты ОС;
- поддержка транзакций на уровне ОС.

Как мы убедились, смарт-карты являются актуальным средством аутентификации, позволяющим организовать надёжную и защищённую работу с компьютером и данными на нём. Но стоимость такого варианта защиты на момент написания статьи, из расчёта на один компьютер, примерно составляет 2000 рублей. Поэтому не всегда экономически выгодно использовать такой метод защиты. Однако

с точки зрения улучшения состояния безопасности организации этот подход считается достаточно рациональным, так как позволяет избавиться от недостатков классических систем аутентификации.

Библиографический список

1. De Clercq Jan. Why does Kerberos smart card login require public key certificates, private keys, and a Certification Authority (CA)? [Электронный ресурс] // Windows IT Pro | Microsoft Windows Information, Solutions, Tools [Официальный сайт]. – URL: <http://windowsitpro.com/security/q-why-does-kerberos-smart-card-login-require-public-key-certificates-private-keys-and-cert> (дата обращения: 01.11.15).

2. Голдовский И.М. Банковские микропроцессорные карты. – М.: Альпина Паблицер, 2010. – 694 с.

3. Даденков С.А., Кон Е.Л. Метод расчета числа узлов-соперников за канал в сети со случайным множественным доступом, построенной на основе нейрочипов // Нейрокомпьютеры: разработка, применение. – 2015. – № 10. – С. 26–32.

4. Даценко П.Е. Атаки на смарт-карты и способы противодействия таковым [Электронный ресурс]. – URL: http://www.rusnauka.com/10_NPE_2009/Informatica/44170.doc.htm (дата обращения: 05.11.15).

5. Кокоулин А.Н., Зее С.Д., Тур А.И. Уязвимости ОС Windows Server 2003 и 2008 // Мир науки и инноваций. – Иваново: Научный мир, 2015. – № 1. – С. 41–46.

ПОСТАНОВКА ЗАДАЧИ ИССЛЕДОВАНИЯ ЭФФЕКТИВНОСТИ ПРОТОКОЛОВ МНОЖЕСТВЕННОГО ДОСТУПА

Студентка Ю.В. Лихачева

Пермский национальный исследовательский
политехнический университет

В статье выполняется постановка актуальной задачи исследования эффективности протоколов множественного доступа. Решение анализируемых в работе задач позволит повысить сетевую производительность.

Ключевые слова: метод множественного доступа, производительность, протокол, промышленная сеть, эффективность, LonWorks.

PROBLEM RESEARCH EFFICIENCY MULTIPLE ACCESS PROTOCOL

Student Y.V. Likhachev

Perm National Research Polytechnic University

In this article, we set the actual problem of research on the effectiveness of multiple access protocols. The solution analyzed problems will improve network performance.

Keywords: multiple access technique, performance, protocol, industrial network, efficiency, LonWorks.

Исторически первым подходом к управлению доступом устройств к разделяемой среде (Media access control, MAC) передачи в сети связи является подход случайного множественного доступа. Предложенный в 1970 г. алгоритм доступа Aloha, а также его модификации получили широкое распространение в различных сетях и в дальнейшем составили основу современных алгоритмов управления случайным доступом. Простота объединения в сеть устройств путём их соединения общей средой передачи и необходимость в повышении её полезной пропускной способности помогли развитию алгоритмов множественного случайного доступа с контролем несущей (Carrier Sense Multiple Access, CSMA) с обнаружением коллизий (Collision Detection, CD) и избеганием коллизий (Collision Avoidance, CA). Широкое распространение протоколы доступа получили в локальных вычислительных (Local Area Network, LAN) проводных сетях с технологией Ethernet (IEEE 802.3), а в дальнейшем и беспроводной связи (например, IEEE 802.11). В последние десятилетия модификации алгоритмов CSMA успешно применяются и в промышленных (fieldbus) локальных сетях управления (Local Operation Network, LON), в том числе с полностью асинхронной

передачей: проводных Can, LonWorks, EIB/KNX, DeviceNet, ProfiNet, Ethernet/IP, Modbus/TCP, Powerlink, а также беспроводных сенсорных сетях One-Net, ZigBee, Z-Wave, Wibree, WirelessHART, IEEE 802.15.4 и др. Растущие требования к быстродействию сетей и необходимость обеспечения детерминированности вероятностно-временных характеристик промышленных систем реального времени актуализируют решение задач анализа, исследования и оптимизации существующих алгоритмов множественного доступа CSMA путём: рекомендаций к использованию их параметров при различной загрузке сетевого канала; обоснованной разработки новых модификаций алгоритмов.

Алгоритмы доступа с контролем несущей могут быть классифицированы по признаку настойчивости передающих устройств (при наличии у них информации для передачи). Все разновидности являются частным случаем общего алгоритма с p -уровнем настойчивости (p -Persistent, p -CSMA), где p -вероятность начала передачи сетевым устройством, имеющим информацию, сразу по освобождению канала. Настойчивые протоколы доступа 1 -CSMA характеризуются передачей информации сразу же после освобождения канала. Протокол с ненастойчивым поведением характеризуется проверкой занятости канала и в случае занятости откладыванием попытки проверки канала и передачи на случайный интервал времени. Отдельно от указанных (абсолютно настойчивых и ненастойчивых) протоколов выделяют алгоритмы с вероятностным уровнем настойчивости, обеспечиваемым случайной задержкой передачи с момента освобождения канала. Известные результаты свидетельствуют о различной эффективности протоколов, в том числе в зависимости от количества устройств в сети и создаваемой ими загрузки канала (рисунок).

Протоколы с настойчивой передачей обеспечивают производительность (малую задержку доступа) при низкой загруженности сетевого канала, а протоколы с ненастойчивой передачей позволяют сократить время доступа (количество попыток передачи, см. рисунок) для успешной передачи за счёт избегания коллизий при средней и высокой загруженности. Протоколы с вероятностной настойчивостью обеспечивают лучшую производительность (время доступа и передачи) на определённых диапазонах загрузки канала.

Эффективность p -CSMA протоколов, с одной стороны, и известные проблемы производительности, с другой, определяют актуальность их исследования и повышение эффективности, реализуемой рекомендациями к применению параметров протоколов и их модификациями

к определённым условиям эксплуатации. Особый интерес представляет анализ протоколов с постоянным и переменным уровнем настойчивости, соответственно называемых чистым p -CSMA и протоколом с прогнозированием нагрузки predictive p -Persistent CSMA.



Рис. Влияние контроля несущей на эффективность использования канала

Исследованию вероятностно-временных характеристик указанных протоколов посвящено множество работ, из которых наиболее ёмкими в открытой печати являются публикации [1–5]. В представленных работах особое внимание уделяется оценке производительности и эффективности функционирования протоколов в условиях различной нагрузки на канал, создаваемой узлами сети, анализу влияния на производительность таких параметров, как приоритеты, ширина соревновательного окна, алгоритм прогнозирования нагрузки на канал и др. В работе [2] показан поиск оптимального значения параметра базового соревновательного окна протоколов, обеспечивающего лучшую производительность на различных диапазонах загруженности сетевого канала. Стоит отметить, что достоинством производимых оценок и работ является применение аналитического подхода, позволяющего производить оперативные вычисления и строить необходимые для анализа графические зависимости. Однако аналитические выражения получены в условиях ряда допущений и ограничений, способствующих понижению адекватности выполняемых оценок. Данную исследовательскую работу предлагается продолжить с использованием современных средств имитационного моделирования [6]. Особый интерес для исследования представляют следующие направления:

– анализ влияния на производительность базового количества слотов доступа (базовой ширины соревновательного окна) и определение оптимальной (по различным критериям) ширины под конкретную загрузку сетевого канала.

– анализ влияния на производительность вероятностного закона выбора слотов доступа для передачи в условиях различной загруженности сетевого канала и количества соперничающих узлов.

– анализ эффективности изменения величины приращения ширины соревновательного окна при увеличении прогнозируемой нагрузки.

Решение сформулированных задач позволит количественно оценить эффективность предлагаемых подходов к оптимизации производительности сети.

Библиографический список

1. Даденков С.А., Кон Е.Л. Подход к построению аналитической модели информационно-управляющей сети LonWorks на основе нейро-чипов // Нейрокомпьютеры: разработка, применение. – 2013. – № 11. – С. 64–69.

2. Даденков С.А., Кон Е.Л. Исследование производительности алгоритма доступа к среде predictive p-persistent CSMA протокола // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2012. – № 6. – С. 217–230.

3. Даденков С.А., Кон Е.Л. Метод расчета числа узлов-соперников за канал в сети со случайным множественным доступом, построенной на основе нейрочипов // Нейрокомпьютеры: разработка, применение. – 2015. – № 10. – С. 26–32.

4. Даденков С.А., Кон Е.Л., Южаков А.А. Алгоритм проектирования промышленной информационно-управляющей сети требуемой производительности (на примере технологии LonWorks) // Вестник Поволж. гос. технол. ун-та. Сер. Радиотехнические и инфокоммуникационные системы. – 2014. – № 5(24). – С. 35–43.

5. Даденков С.А., Кон Е.Л. Оценка степени влияния некоторых факторов на производительность LonWorks-сети // Образовательные ресурсы и технологии. – 2014. – № 2(5). – С. 72–76.

6. Даденков С.А., Кон Е.Л. Анализ моделей и методов агентного и дискретно-событийного имитационного моделирования // Известия СПбГЭТУ «ЛЭТИ». – 2015. – № 5. – С. 35–41.

МАТЕМАТИЧЕСКАЯ МОДЕЛЬ АТАК НА ИНФОРМАЦИОННЫЕ РЕСУРСЫ КОРПОРАТИВНЫХ АВТОМАТИЗИРОВАННЫХ СИСТЕМ

**Аспирант, ассистент В.С. Лужнов,
канд. техн. наук, доцент А.Н. Соколов**
Южно-Уральский государственный университет (Национальный
исследовательский университет), г. Челябинск

В статье рассматриваются основные типы специализированных математических моделей, на основе которых возможно проведения анализа защищенности автоматизированных систем. Описываются их свойства и способ формализации на основе иерархических деревьев. Подробно рассматривается математическая модель атак на информационные ресурсы автоматизированных систем и описывается ее формализация рассмотренным способом.

Ключевые слова: информационная безопасность, автоматизированные системы, защищенность, математические модели, безопасность автоматизированных систем, анализ защищенности автоматизированных систем.

MATHEMATICAL MODEL OF ATTACKS ON INFORMATION RESOURCES OF THE CORPORATE AUTOMATED SYSTEMS

**Graduate Student, assistant V.S. Luzhnov
Ph.D. in Technical Sciences, Associate Professor A.N. Sokolov**
South Ural State University (National Research University), Chelyabinsk

This article focuses on the main types of specialized mathematical models on the basis of which it is possible to analyze the security of automated systems. It describes their properties and method of formalization, based on a hierarchical tree. Discussed in details a mathematical model of attacks on information resources of automated systems and a way of its hierarchical formalization.

Keywords: information Security, automated systems, security, mathematical models, security of automated systems, security analysis of automated systems.

С ростом уровня автоматизации процессов управления на предприятиях (организациях) и повсеместным внедрением автоматизированных систем (АС) различного уровня и назначения актуальной становится проблема разработки средств защиты информационных ресурсов АС от внутренних и внешних угроз информационной безопасности. Для снижения трудоемкости работ по созданию, внедрению и оценке эффективности средств защиты применяются специализированные математические модели, выступающие в качестве

эталонных, на основе которых возможно осуществление анализа защищенности конкретной АС. Существуют три основных типа таких моделей [1]:

- модели информационных атак, предназначенные для воспроизведения необходимых свойств и характеристик атак. Модели этого типа позволяют в лабораторных условиях провести исследование характеристик определённой атаки для того, чтобы определить, какие средства защиты могут использоваться для нейтрализации атаки;

- модели процесса обнаружения информационных атак, позволяющие формально описать процесс выявления атак на информационные ресурсы АС;

- модели процесса оценки рисков информационной безопасности АС, которые позволяют определить эффективность применения всей системы обеспечения безопасности АС в целом.

Для практической реализации в виде комплекса алгоритмов, который в дальнейшем может быть использован в рамках создания специализированного программного обеспечения, особый интерес представляют модели информационных атак на ресурсы автоматизированных систем [2].

Для эффективного использования моделей атак при исследовании возможных действий нарушителя по отношению к АС они должны обладать следующими основными свойствами [1, 3]:

- универсальность, позволяющая использовать модель для представления различных типов атак вне зависимости от источника, объекта и средства реализации атаки;

- расширяемость, обеспечивающая возможность добавления в модель новых характеристик атаки. Это свойство позволяет пользователю изменять состав характеристик моделируемых атак в зависимости от среды АС, в которой они рассматриваются;

- формализуемость – свойство, которое указывает на возможность использования математического аппарата для описания параметров модели;

- простота, определяющая степень возможности пользователя воспринимать структуру и способы реализации моделируемой атаки;

- многофакторность, позволяющая учитывать три основных параметра моделируемой информационной атаки: уязвимость, активизируемую атакой, способ реализации атаки и её возможные последствия.

В целях обеспечения математического аппарата исследований и моделирования описанные модели могут быть формализованы в виде иерархического дерева (связного ациклического графа) [4] $T = \langle M, N \rangle$, где M – множество вершин дерева, N – множество дуг дерева. Каждая вершина дерева T ассоциируется с определённым действием нарушителя, а корень дерева обозначает конечную цель информационной атаки. На графе T имеется возможность составить множество возможных путей T_p , где каждый путь $t_p \in T_p$ представляет собой последовательность дуг $(n_{p1}, n_{p2}, \dots, n_{pn})$ вида $n_{pk} = (m_i, m_j)$, $m_i, m_j \in M$, при этом конечная вершина дуги n_{pk} одновременно является начальной вершиной дуги n_{pk+1} . В качестве начальной вершины пути могут выступать листья дерева T , а в качестве конечной вершины – корень дерева T . Таким образом, каждый элемент множества T_p описывает один из возможных сценариев проведения информационной атаки.

Частным случаем моделей, базирующимся на описанном способе формализации, выступает модель атак на информационные ресурсы. Предлагаемая модель атак на информационные ресурсы в своей основе состоит из трех базовых множеств: V – множество уязвимостей информационных ресурсов автоматизированной системы, A – множество способов реализации атак на информационные ресурсы, C – множество последствий реализации атак на информационные ресурсы. Основные положения рассмотренной модели приведены в [5].

Для описания связей, существующих между элементами множеств A , V и C , необходимо определить n -арное алгебраическое отношение (тернарное при $n = 3$) W на множестве: $W = A \times V \times C$.

Тогда элемент (a, v, c) , принадлежащий отношению W , где $a \in A, v \in V, c \in C$, в рамках модели представляет собой логическую структуру вида «Атака на информационные ресурсы, которая реализуется способом a через эксплуатацию уязвимости v , приводящая к последствию c ».

С каждой уязвимостью $v_i \in V$ связано множество A_i , которое является подмножеством множества A и включает в себя атаки на информационные ресурсы, эксплуатирующие уязвимость v_i . При этом справедливо отношение: $0 < |A_i| < |A|$, т.е. уязвимость v_i не

может эксплуатироваться для реализации всех атак на информационные ресурсы из множества A . При этом не существует такой уязвимости, на основе которой не могло быть реализовано ни одной атаки на информационные ресурсы.

С каждой информационной атакой $a_j \in A$ связано множество V_j , которое является подмножеством множества V и включает в себя уязвимости, эксплуатируемые атакой a_j . При этом справедливо отношение $0 < |V_j| < |V|$, т.е. атака a_j не может эксплуатировать одновременно все уязвимости информационных ресурсов автоматизированной системы. При этом не существует такой атаки, которая бы не эксплуатировала ни одной уязвимости информационных ресурсов автоматизированной системы.

С каждой информационной атакой $a_j \in A$ связано множество C_j , которое является подмножеством множества C и включает в себя последствия, вызываемые атакой a_j . При этом справедливо отношение $0 < |C_j| < |C|$, т.е. атака a_j не может привести одновременно ко всем последствиям, входящим в множество C . При этом атака a_j не может не приводить ни к одному последствию.

С каждым последствием $c_k \in C$ связано множество A_k , которое является подмножеством множества A и включает в себя атаки на информационные ресурсы автоматизированной системы, которые приводят к последствию c_k . При этом справедливо отношение $0 < |A_k| < |A|$, т.е. не существует такого последствия, к которому бы не привело ни одной атаки. При этом последствие не может быть следствием реализации всех атак, входящих в множество A .

Описанная математическая модель, с учетом уже указанного способа формализации, может быть представлена в виде графа $G = \langle L, E \rangle$, где L – множество вершин графа, а $E \in L^2$ – множество дуг графа. Для графа G определено отношение $T \in \{E \times W\}$, которое каждой дуге из множества E ставит в соответствие один или более элементов отношения W . Использование отношения T позволяет интерпретировать каждую дугу графа G как один из типов моделируемой

атаки на информационные ресурсы автоматизированной системы. При этом в отношении T одной дуге $e \in E$ может соответствовать одновременно несколько элементов множества W только при условии, что эти элементы обозначают атаки, приводящие к одним и тем же последствиям, т.е.:

$$(\forall e \in E), (\forall w' \in W), (\forall w'' \in W) \exists (e, w') \in T, \exists (e, w'') \in T \leftrightarrow c' = c'',$$

где $w' = (a', v', c')$, $w'' = (a'', v'', c'')$ – элементы, принадлежащие множеству W , a' и a'' – способы реализации атак, v' и v'' – уязвимости, c' и c'' – последствия реализации атак.

В каждую вершину графа G может входить одновременно несколько дуг только при условии, что в отношении T каждой такой дуге соответствуют элементы множества W , описывающие атаки на информационные ресурсы автоматизированной системы, которые приводят к одинаковым последствиям. Таким образом, вершины графа G могут объединять различные этапы атаки на информационные ресурсы, приводящие к идентичным последствиям.

Пример описанного графа G приведен на рисунке.

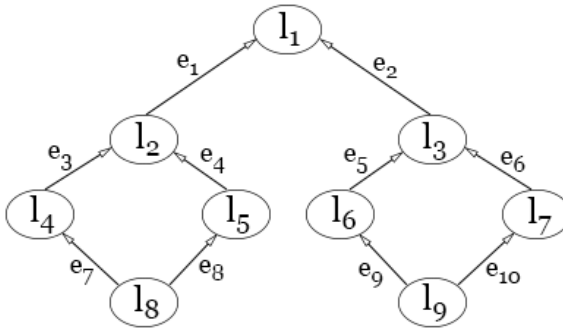


Рис. Пример графа G : $l_1 \dots l_9$ – вершины графа G ,
 $e_1 \dots e_{10}$ – дуги графа G

К графу применимо отношение T :

$$T = \{(e_1, (a_1, v_2, c_1)), (e_2, (a_2, v_1, c_1)), (e_3, (a_2, v_1, c_2)), (e_4, (a_3, v_4, c_2)), \\ (e_5, (a_4, v_3, c_3)), (e_6, (a_5, v_5, c_3)), (e_7, (a_5, v_6, c_4)), (e_8, (a_6, v_7, c_5)), \\ (e_9, (a_7, v_7, c_6)), (e_{10}, (a_8, v_3, c_7))\}.$$

Описанная математическая модель атак на информационные ресурсы автоматизированных систем может использоваться для реализации на ее основе алгоритмов и, как следствие, программного обеспечения, позволяющего как проводить анализ защищенности автоматизированных систем, так и в специализированном виде предотвращать эксплуатацию уязвимостей информационных ресурсов направленными атаками.

Библиографический список

1. Сердюк В.А. Разработка и исследование математических моделей защиты автоматизированных систем от информационных атак. – М.: Изд-во РГТУ им. К.Э. Циолковского, 2004.

2. Лужнов В.С., Соколов А.Н. Программные средства анализа защищенности автоматизированных систем. Проблемы и перспективы // Безопасность информационного пространства: сб. тр. XIII Всерос. науч.-практ. конф. студ., аспирантов и молодых ученых. – Челябинск: Изд. центр ЮУрГУ, 2015. – С. 120–124.

3. Васильев В.В., Симак Л.А., Рыбникова А.М. Математическое и компьютерное моделирование процессов и систем. – Киев: Изд-во ИПМЭ, 2008.

4. Шнайер Б. Моделирование угроз безопасности [Электронный ресурс]. – URL: <https://www.schneier.com/paper-attacktrees-ddj-ft.html> (дата обращения: 19.10.2015).

5. Сердюк В.А. Математическая модель оценки уровня защищенности сетей передачи данных // Моделирование. Теория, методы и средства: материалы II Международ. науч.-практ. конф. – Новочеркасск, 2002. – С. 31–34.

АНАЛИЗ СТОЙКОСТИ СИММЕТРИЧНЫХ ШИФРОВ ГОСТ Р 34.12-2015

Ассистент Е.А. Маро

Южный федеральный университет, г. Таганрог

В работе проведено моделирование алгебраического анализа шифра ГОСТ 34.12-2015 $n = 64$ (Магма), построенного на основании сети Фейстеля. Приведены численные характеристики нелинейной системы второй степени, описывающей преобразования в фиксированных блоках замены размером 4 на 4 бита.

Ключевые слова: алгебраический криптоанализ, симметричные блочные алгоритмы шифрования, многопроцессорные вычислительные системы.

ANALYSIS OF SYMMETRIC CIPHERS GOST R 34.12-2015

Assistant E.A. Maro

Southern Federal University, Taganrog

Article shows an algebraic analysis of three rounds GOST 34.12-2015 $n = 64$ (Magma), which is built on the basis of a Feistel network. Characteristics of the nonlinear system of the second degree, which describes the fixed S-boxes (size of 4 by 4 bits).

Keywords: algebraic cryptanalysis, a symmetric block cryptographic algorithms, multiprocessor computing systems.

Идея описать алгоритм шифрования системой линейных уравнений возникла давно, предположения о возможности реализации подобной атаки были высказаны в работе Шеннона [1]. Однако до недавнего времени она оставалась всего лишь идеей, так как упиралась в необходимость хранения большого числа переменных и решения больших систем уравнений. В настоящий момент сравнительно мало исследований, которые бы показывали возможность вскрытия полного реального алгоритма шифрования. В работах [2–4] приводились расчеты возможности алгебраической атаки на блочный шифр DES, позднее в статье [5] описана алгебраическая атака на 6 раундов шифра DES при известном одном шифротексте.

В работе [6] описан способ представления алгоритма блочного шифрования AES в виде одного уравнения с большим числом неизвестных над полем $GF(2)$. Для 128-битного шифра AES данное уравнение содержало 2^{50} неизвестных и на практике не представляло угрозы для стойкости шифра. Полученные авторами результаты

показали, что возможность представления алгоритма блочного шифрования в виде полиномиального уравнения еще не свидетельствует об его уязвимости к алгебраическим атакам.

В работах [7, 8], например, приводятся дальнейшие исследования алгебраического анализа стандарта AES. Так, в работе [7] проведены атаки на упрощенные алгоритмы шифрования Serpent, Present и AES, в ходе которых была продемонстрирована их потенциальная уязвимость к алгебраическим атакам. Работа [8] содержит результаты алгебраического анализа полной версии алгоритмов AES-128 и Cryptomeria путем нахождения базиса Гребнера. В статье [9] приводятся результаты алгебраического анализа методом SAT-solving 8 раундов блочного алгоритма шифрования PRINTCipher-48 при известных двух текстах и анализ 9 раундов при выполнении некоторых допущений. При рассмотрении алгебраического анализа ГОСТ 28147-89 следует выделить работу [10], позволившую выполнить атаку в 2^{39} раз быстрее, чем атака методом грубой силы, при известных 2^{64} текстах. Однако в данной статье автор пока не раскрывает используемый алгоритм атаки на ГОСТ 28147-89. В работах [11–13], посвященных криптоанализу нового стандарта шифрования ГОСТ Р 34.12-2015, приведены описания атак методом анализа ошибок (fault analysis attacks), методом «человек посередине» (MitM attack) на 5 раундов шифрования, временная атака (timing attack) на реализованные в CUDA шифры, подобные AES.

Применительно к алгоритмам блочного шифрования алгебраические атаки можно представить в виде двух этапов. На первом этапе необходимо представить алгоритм шифрования и по возможности некоторую дополнительную информацию об алгоритме в виде систем полиномиальных уравнений над полем GF(2) или другим конечным полем. Второй этап заключается в решении системы уравнений и получении из решения системы ключа шифрования.

В открытой печати не было найдено данных по изучению и моделированию алгебраических атак на новый национальный стандарт симметричного шифрования ГОСТ Р 34.12 - 2015. В ходе выполнения работы рассмотрен алгебраический метод eXtended Linearization (XL) применительно к описанному в стандарте алгоритму шифрования $n = 64$ (далее называемый шифр «Магма», построенный на основе сети Фейстеля).

Шифр Магма состоит из 32 раундов шифрования, построенных по принципу сети Фейстеля. Блок открытого текста разбивается на

две равные части по 32 бита каждая. Над правой частью открытого текста выполняется раундовое преобразование, состоящее из трех операций:

- сложение с раундовым ключом по модулю 2^{32} ;
- замена в восьми фиксированных S-блоках;
- циклический сдвиг влево на 11 позиций.

Блоки замены имеют вид:

$\pi 0' = (12, 4, 6, 2, 10, 5, 11, 9, 14, 8, 13, 7, 0, 3, 15, 1)$;

$\pi 1' = (6, 8, 2, 3, 9, 10, 5, 12, 1, 14, 4, 7, 11, 13, 0, 15)$;

$\pi 2' = (11, 3, 5, 8, 2, 15, 10, 13, 14, 1, 7, 4, 12, 9, 6, 0)$;

$\pi 3' = (12, 8, 2, 1, 13, 4, 15, 6, 7, 0, 10, 5, 3, 14, 9, 11)$;

$\pi 4' = (7, 15, 5, 10, 8, 1, 6, 13, 0, 9, 3, 14, 11, 4, 2, 12)$;

$\pi 5' = (5, 13, 15, 6, 9, 2, 12, 10, 11, 7, 8, 1, 4, 3, 14, 0)$;

$\pi 6' = (8, 14, 2, 5, 6, 9, 1, 12, 15, 4, 11, 0, 13, 10, 3, 7)$;

$\pi 7' = (1, 7, 14, 13, 0, 5, 8, 3, 4, 15, 10, 6, 9, 12, 11, 2)$.

При анализе шифра Магма (аналогичный шифру ГОСТ 28147-89 с фиксированными блоками замены) проанализированы используемые блоки замены и сформированы системы булевых уравнений для каждого из восьми узлов замены. Алгебраическая иммунность блоков замены шифра Магма равна 2. Опираясь на теорему, доказательства которой приведены в работе [5], для любого блока замены размером $n \times m$ бит: $F(x_1, \dots, x_n) \rightarrow (y_1, \dots, y_m)$, и для любого подмножества T из t всех возможных одночленов (2^{n+m}), если выполняется условие $t > 2^n$, то существует по меньшей мере $t - 2^n$ линейно независимых уравнений, содержащих одночлены из множества T и выполняющихся с вероятностью 1. Для блоков замены, приведенных в стандарте ГОСТ Р 34.12-2015, получены 168 линейно независимых уравнения, связывающих входы и выходы каждого S-блока (работу одного блока замены описывает 21 уравнение). В общем виде уравнения, описывающие преобразования в 4-битном блоке замены, можно задать формулой:

$$\sum \alpha_{i,j} x_i x_j + \sum \beta_{i,j} y_i y_j + \sum \gamma_{i,j} x_i y_j + \sum \delta_i x_i + \sum \epsilon_i y_i + \eta = 0,$$

где $x_i x_j$ – комбинация входных битов S-блока; $y_i y_j$ – комбинация выходных битов S-блока; $x_i y_j$ – комбинация входных и выходных битов; x_i и y_i – соответственно входные и выходные биты S-блока; η – коэффициент, принимающий значения 0 или 1.

Реализована параллельная программа формирования системы уравнений для восьми блоков замены. Программа реализована в среде

Python, с использованием OpenMP. Составление и проверка системы уравнений для восьми блоков на ПК (Intel Core i5 2,8 ГГц, 8 Гбайт) потребовали около 20 ч (74453 с).

Для двух раундов шифрования Магма система уравнений, содержит 336 уравнений, 128 неизвестных (входные и выходные значения блоков замены) и 448 произведений неизвестных. Рассмотрим варианты решения составленной системы. Метод линеаризации не может быть применим без дополнительных преобразований, так как число уравнений меньше, чем число мономов. Однако учитывая структуру алгоритма шифрования Магма, представим входные (X_1 и X_2) и выходные (Y_1 и Y_2) значения блоков замены в каждом раунде через известные открытый текст (OT) и шифротекст (C) по формулам: $X_1=OT_R \oplus K_1$, $Y_1=(OT_L \oplus C_R) \ggg 11$, $X_2=C_R \oplus K_2$, $Y_2=(OT_R \oplus C_L) \ggg 11$.

Таким образом, число неизвестных системы сократилось до 64 вместо 128, а число произведений неизвестных в системе уменьшится до 160.

Для трех раундов шифра исходная система содержит 504 уравнения, 192 неизвестных и 672 произведений неизвестных. Используя метод XL, выполняется умножение уравнений каждого блока замены на одночлены первой степени. Таким образом, сформируем дополнительно 4032 уравнения. Система после умножения содержит 4536 уравнений, 192 неизвестных и 2016 произведений неизвестных. Часть из вновь полученных уравнений не являются линейно независимыми, поэтому они не рассматриваются при решении системы, приведенной к линейному виду.

При теоретическом рассмотрении анализа N-раундов шифра можно составить $168N$ уравнений с $64N$ неизвестными и $224N$ произведений неизвестных.

Разработчиками алгебраического метода анализа сделано допущение, что под сложностью атаки рассматривается решение системы, приведенной к линейному виду (для метода исключения Гаусса сложность $O(n^3)$, не учитывая сложность подготовительных этапов по составлению системы и получению дополнительных уравнений.

Работа выполнена при поддержке гранта РФФИ №15-37-20007 мол_а_вед.

Библиографический список

1. Claude Shannon Communication Theory of Secrecy Systems // Bell System Technical Journal. – 1949. – № 28(4). – P. 656–715.
2. Schaumuller-Bichl I. Cryptanalysis of the Data Encryption Standard by the Method of Formal Coding, In Cryptography // Proc. Burg Feuerstein 1982, LNCS 149, T. – Beth editor, Springer-Verlag, 1983.
3. Marraro L., Massacci F. Towards the Formal Verification of Ciphers: Logical Cryptanalysis of DES // Proc. Third LICS Workshop on Formal Methods and Security Protocols: Federated Logic Conferences (FLOC-99). – 1999.
4. Massacci F. Using Walk-SAT and Rel-SAT for cryptographic key search // Proc. 16th International Joint Conference on Artificial Intelligence (IJCAI'99). – 1999.
5. Courtois Nicolas T., Bard Gregory V. Algebraic Cryptanalysis of the Data Encryption Standard. – URL: <http://eprint.iacr.org/2006/402/>
6. Ferguson Niels, Schroepel Richard, Whiting Doug. A Simple Algebraic Representation of Rijndael // SAC 2001. LNCS 2259. – Springer. – P. 103.
7. Nicolas T. Courtois and Blandine Debraize Specific S-Box Criteria in Algebraic Attacks on Block Ciphers with Several Known Plaintexts // Second Western European Workshop, WEWoRC 2007. – Bochum, Germany, July 4-6, 2007. – P. 100–113.
8. Weinmann Ralf-Philipp. Algebraic Methods in Block Cipher Cryptanalysis: Dissertation. – 2013.
9. Stanislav Bulygin Algebraic cryptanalysis of the round-reduced and side channel analysis of the full PRINTCipher-48 // LNCS. – 2011. – Vol. 7092. – P. 54–75.
10. Nicolas T. Courtois Algebraic Complexity Reduction and Cryptanalysis of GOST. – URL: <http://www.nicolascourtois.com/papers/gostac11.pdf>
11. AlTawy R., Duman O., Youssef A.M. Fault analysis of Kuznyechik. – URL: <https://eprint.iacr.org/2015/347.pdf>
12. AlTawy R., Youssef A.M. A meet in the Middle Attack on Reduced Kuznyechik. – URL: <https://eprint.iacr.org/2015/096.pdf>
13. Fomin D. A timing attack on CUDA implementations of an AES-type block cipher // 4rd Workshop on Current Trends in Cryptology (CTCrypt 2015). – 2015. – P. 288–302.

К ПРОБЛЕМЕ НАДЕЖНОСТИ ВЫЧИСЛИТЕЛЬНОЙ СЕТИ

Курсант А.А. Олейников, канд. техн. наук, доцент А.В. Тарутин
Пермский военный институт внутренних войск Министерства
внутренних дел России

Рассматривается вопрос необходимости резервирования. Раскрыты классификация локально-вычислительных сетей, цели и методы резервирования. Особое внимание обращено на преимущества и недостатки различных методов резервирования. На основе анализа выработаны наиболее целесообразные методы с учетом поставленной задачи.

Ключевые слова: локальная вычислительная сеть, надежность, резервирование.

TO THE PROBLEM OF COMPUTER NETWORK RELIABILITY

Cadet A.A.Oleynikov,
Ph.D. in Technical Sciences, Associate Professor A.V. Tarutin
Perm Military Institute of Internal Troops of the Ministry
of Internal Affairs of Russia

The article focuses on the need of reservation. It reveals a classification of local area networks, the goals and methods of reservation. Particular attention is drawn to the advantages and disadvantages of various methods of reservation. On the basis of the analysis the most expedient methods were developed taking into account the assigned objectives.

Keywords: local area network, reliability, reservation.

На сегодняшний день в мире существует более 2,5–3 млрд компьютеров, большинство из них объединены в различные информационно-вычислительные сети, от малых локальных сетей в офисах одного предприятия до глобальных сетей типа Интернета.

Тенденция к объединению компьютеров в сети обусловлена рядом важных причин, основные из них – возможность быстрого обмена информацией между пользователями, ускорение передачи информационных сообщений, получение и передача сообщений (факсов, e-mail писем и прочего), не отходя от рабочего места, возможность мгновенного получения любой информации из любой точки земного шара, а также обмен информацией между компьютерами разных фирм производителей, работающих под разным программным обеспечением.

Вопрос классификации вычислительных сетей раскрыт достаточно полно во многих работах. За основу можно принять следующую классификацию [1]:

- по степени территориальной рассредоточенности существуют:
 - крупномасштабные или глобальные сети, охватывающие территорию страны или нескольких стран с расстоянием между звеньями от сотен до нескольких тысяч километров;
 - региональные сети, расположенные в пределах определенного территориального региона (города, области и т. д.);
 - локальные сети, охватывающие достаточно небольшую территорию $R = 1 \dots 10$ км;
- по функциональным, информационным (по способу хранения и доставки информации) и структурным признакам;
- по способу управления ресурсами сети;
- по методу передачи данных.

Рассмотрим, что представляет собой локальная вычислительная сеть (ЛВС).

Локальная сеть представляет собой 2 и более соединённых друг с другом компьютеров, предназначенных для совместного использования ресурсов, подключение к сети периферийных устройств, таких как принтеры, плоттеры, диски, модемы, приводы CD-ROM и другие периферийные устройства. Локальная сеть обычно ограничена территориально одним или несколькими близко расположенными зданиями [2].

Компьютер в составе ЛВС должен иметь следующие компоненты:

- сетевой адаптер;
- кабель;
- сетевую операционную систему (сетевые программы).

Но важно знать, что у каждого элемента ЛВС существует время безотказной работы, и в конечном итоге на участке сети, имеющем наименьшее время на отказ, произойдет сбой передачи данных в связи с прерыванием работы (выход из строя коммутатора, неисправность кабеля и т.д.).

Необходимо учитывать, что для персонала, работающего со статистической информацией, малоизменяющейся в короткий промежуток времени, невозможность передачи информации в определенный момент времени не представит неудобств. С другой же стороны, работники, обрабатывающие оперативную информацию, должны

всегда иметь доступ к вновь поступающей информации и предоставление переработанных данных для принятия решения в реальном времени [3].

Примером могут служить операторы на железных дорогах, диспетчеры авиакомпаний, диспетчеры неотложных служб (скорая помощь, полиция, пожарная служба, вооруженные силы и т.д.). У данных специалистов промедление в получении информации или принятии решения может повлечь гибель большого количества людей и уничтожению материального имущества. С этой целью выполняют резервирование ЛВС.

Резервирование – это повышение надежности объекта введением избыточности, т.е. дополнительных средств и возможностей сверх минимально необходимых для выполнения объектом возложенных на него функций [4].

В различных областях техники используются разные виды резервирования – структурное, функциональное, информационное.

В силовых структурах для оперативного состава в основном применяют структурное резервирование, т.е. используют избыточные (резервные) конструктивные элементы, включенные параллельно основным (рабочим) и дублирующие их. При этом основным называют такой элемент структуры объекта, который минимально необходим для выполнения объектом заданных функций, а резервный обеспечивает работоспособность объекта в случае отказа основного элемента.

Как показано на рис. 1, структурное резервирование может осуществляться разными способами.

При общем резервировании резервируется объект в целом, а при раздельном – его отдельные элементы. Кратностью резервирования называют отношение числа резервных элементов к числу основных,

$$K_p = N_{\text{рез}}/N_{\text{осн}}.$$

По числу резервных элементов различают однократное, двукратное и многократное резервирование. При раздельном резервировании K_p чаще всего бывает дробной величиной, а при общем – целым числом. В силовых структурах, как правило, используется раздельное резервирование в виде дублирования отдельных наименее надежных и наиболее ответственных элементов, например линий видеоконференцсвязи. На один основной элемент обычно приходится один резервный.



Рис. 1. Схема резервирования

При постоянном резервировании резервные элементы участвуют в функционировании объекта наравне с основным, а при резервировании замещением функции основного элемента передаются резервному элементу только после отказа основного.

Термины «горячий», «теплый» и «холодный» резервы весьма удачны, применительно к структурам правопорядка – соответственно «нагруженный», «облегченный», «ненагруженный» резервы (рис. 1).

Преимуществом ненагруженного резерва является возможность сохранить ресурс резервного элемента при нормальной работе остальных. Однако на практике ненагруженный резерв имеет серьезный недостаток – во многих случаях его нельзя ввести в работу тотчас после возникновения отказа основного оборудования, и поэтому могут временно ухудшиться условия поддержания заданной нагрузки. Так, если исправный коммутатор остановлен в резерв, то его ресурс не расходуется, но даже в самой экстренной ситуации потребуется некоторое вполне определенное время для установки и включения. Коммутатор может также работать с относительно малой нагрузкой (так называемый вращающийся резерв), и при необходимости набор нагрузки производится в темпе, ограниченном только динамическими свойствами энергоблока, но ресурс агрегата расходуется постоянно. Структурное раздельное резервирование элемента организуется двумя способами.

При постоянном включении, как показано на рис. 2, резервный элемент включен параллельно основному и работает вместе с ним. При отказе основного элемента сеть сохраняет работоспособность за счет резервного элемента, принимающего на себя всю нагрузку. В этом случае нет необходимости включать резервный элемент и отключать отказавший основной, но резервный элемент изнашивается и расходует свой ресурс надежности вместе с основным.



Рис. 2. Постоянное включение резерва

Надежность системы двух одинаковых элементов (основного и резервного), включенных параллельно:

$$\lambda_{\text{осн}} = \lambda_{\text{рез}} = \lambda = 1/T_{\text{эл.}}$$

Данная система откажет при одновременном отказе обоих элементов. Согласно формуле полной вероятности при независимости событий вероятность отказа системы двух элементов при постоянном включении:

$$Q_c = \prod Q_i = (1 - P_i)^2 = (1 - e^{-\lambda t})^2 = 1 - 2e^{-\lambda t} + e^{-2\lambda t}.$$

Вероятность безотказной работы указанной системы:

$$P_c = 1 - Q_c = 2 \cdot e^{-\lambda t} - e^{-2\lambda t}.$$

Среднее время безотказной работы:

$$T_c = \int P_c dt = \int (2 \cdot e^{-\lambda t} - e^{-2\lambda t}) dt = (-2/\lambda \cdot e^{-\lambda t} + 1/2 \cdot \lambda e^{-2\lambda t}) = 3/2 \cdot T_{\text{эл.}}$$

Таким образом, при постоянном включении среднее время безотказной работы системы увеличилось в 1,5 раза.

При резервировании замещением, как показано на рис. 3, резервный элемент отключен, находится в состоянии готовности заменить отказавший основной элемент (холодной, теплой или горячей готовности). При этом резерв надежности дублирующих элементов сохраняется, повышается общая надежность системы, но требуется осуществить включение резерва, вероятность чего также должна

быть учтена. Введение в эксплуатацию резерва состоит в поиске отказа, отключении отказавшего элемента, подготовке и вводе резервного элемента в работу.



Рис. 3. Резервирование замещением

Количественный анализ показывает, что среднее время безотказной работы систем двух элементов при резервировании замещением увеличивается вдвое, $T_c = 2 \cdot T_{эл}$.

Следовательно, предпочтительнее резервирование замещением элементов. Но преимущества резервирования замещением перед постоянным включением резервного элемента снижаются, поскольку надежность включения менее 1, и утрачиваются по мере приближения ее к $1,5/2 = 0,75$. Кроме того, следует учесть, что резервный элемент частично изнашивается и в нерабочем состоянии.

Библиографический список

1. Олифер В.Г. Компьютерные сети. – 4-е изд. – СПб.: Питер, 2010. – 958 с.
2. Таненбаум Э. Компьютерные сети. – 5-е изд. – СПб.: Питер, 2012. – 960 с.
3. Мелехин В.Ф., Павловский Е.Г. Вычислительные машины, системы и сети. – М.: Академия, 2006. – 556 с.
4. Черкесов Н.Г. Надежность аппаратно-программных комплексов. – СПб.: Питер, 2005. – 479 с.

ПРОЕКТИРОВАНИЕ СИСТЕМ СВЯЗИ СТАНДАРТА GSM-1800 В РАЙОНАХ С МАЛОЙ ПЛОТНОСТЬЮ НАСЕЛЕНИЯ

Студент А.О. Пукса

Омский государственный технический университет

Рассмотрены вопросы проектирования сети связи стандарта GSM-1800 в районах с малой плотностью населения, располагающихся в отдалённых местах нашей страны. Рассмотрен пример проектирования новой сети стандарта связи GSM-1800 в г. Анадырь.

Ключевые слова: GSM-1800, система связи, затухание.

PROJECTING OF COMMUNICATION GSM-1800 IN AN AREA WITH LOW POPULATION DENSITY

Student A.O. Puksa

Omsk State Technical University

This article describes how to design a communication network GSM-1800 in areas with low population density and are located in remote areas of our country. An example of designing a new network communication standard GSM-1800 in the city of Anadyr.

Keywords: GSM-1800, communications systems, attenuation.

Рассмотрим аспекты проектирования систем связи в районах с малой плотностью населения на примере Чукотского автономного округа. Чукотский автономный округ расположен на крайнем северо-востоке России и занимает весь Чукотский полуостров, часть материка и ряд островов [4].

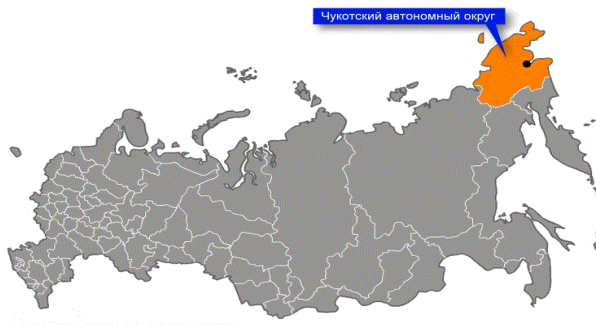


Рис. 1. Географическое расположение г. Анадырь

Спроектируем сеть сотовой связи стандарта GSM-1800 для г. Анадырь и произведём расчёт основных её параметров. Анадырь является самым крупным городом Чукотского автономного округа, его население составляет 13747 человек на 2013 г. Проектируемая сеть сотовой связи будет интегрироваться в уже существующую сеть компании «Теле2 Рус».

Сложности при проектировании сети связи GSM-1800 в г. Анадырь связаны со следующими пунктами:

- удалённое географическое расположение;
- низкая плотность населения;
- сложные погодные условия;
- наличие действующих сетей связи от конкурирующих организаций («ВымпелКом», «МТС» и «МегаФон»).

Начальные данные по распределению голосового трафика выполнены на основании прогнозируемых данных компанией «Теле 2 Рус»:

- проектная мощность сети ПРТС на территории Чукотского автономного округа составит 10 000 абонентов;
- суммарная удельная нагрузка, создаваемая одним абонентом сети, принята равной 0,015 Эрл;
- трафик мобильный – мобильный (внутри сети ПРТС) составляет 92 %;
- трафик на ТЗУС г. Анадырь – 4 % (двусторонние линии);
- трафик на ТМГУС 1/3 г. Хабаровск – 4 % (двусторонние линии).

На рис. 2 изображена спроектированная для г. Анадырь схема стандарта GSM-1800.

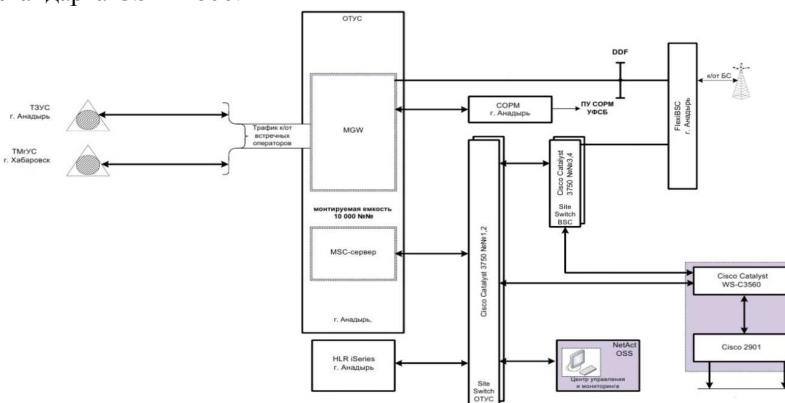


Рис. 2. Спроектированная схема стандарта GSM-1800

Проектируемая сеть предназначена для передачи голосового трафика внутри сети, а также к абонентам и от абонентов других сетей и пользователей PSTN.

Центр коммутации содержит в себе следующие компоненты:

- MGW, необходимый для передачи трафика в различных направлениях для разных типов сетей;

- VLR (визитный регистр местоположения), являющийся базой данных, содержащей информацию о всех абонентах мобильной связи, расположенных в данный момент в зоне обслуживания MSC.

Для функционирования сети необходим HLR (опорный регистр местоположения).[1]

Рассмотренные компоненты являются базисом сети коммутации, для полноценного функционирования которых необходимы вспомогательные элементы.

Шлюз MGW осуществляет передачу трафика в 4 направлениях:

- ТЗУС. Транзитно-зоновый узел связи служит для обработки вызовов к абонентам и от абонентов сети PSTN и мобильных операторов города Анадырь;

- ТМГУС. Транзитный междугородний узел связи обслуживает вызовы к мобильным и стационарным абонентам других регионов, и от них, а также осуществляет связь с биллинг-центром, располагается в г. Хабаровске;

- FlexiBSC. Контроллер базовых станций управляет всеми функциями, относящимися к работе радиоканалов в сети GSM. Это коммутатор большой емкости, который обеспечивает функции хэндовера MS, назначения радиоканалов и сбора данных о конфигурации сот;

- COPM. Это система, необходимая для проведения оперативно-розыскных мероприятий в сетях телефонной, подвижной и беспроводной связи и радиосвязи.

Для обмена данными между MSC-сервером и HLR используется коммутатор (Switch). Также данный коммутатор нужен для обмена информацией с биллинг-центром, SMS-центром, USSD-центром и OSS-центром.

Для связи MGW с транзитно-зоновым узлом связи используется ЦРПЛ «Радиус 15М». Протяжённость линии составляет 14 км. На рис. 3 изображён профиль трассы.

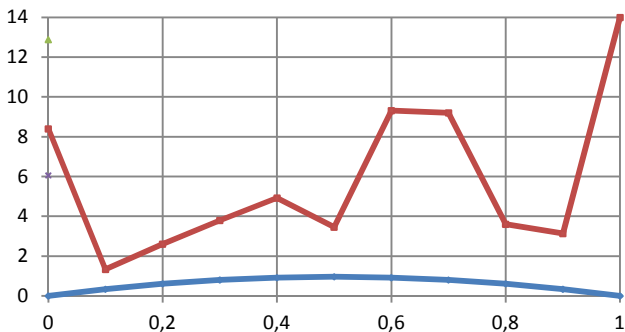


Рис. 3. Профиль трассы

На рис. 4 представлена диаграмма уровней для проектируемой ЦРРЛ. Процент времени, в течение которого множитель ослабления меньше минимально допустимого за счет экранирующего действия препятствий на пролете РРЛ, составляет 0,0001 % [2].

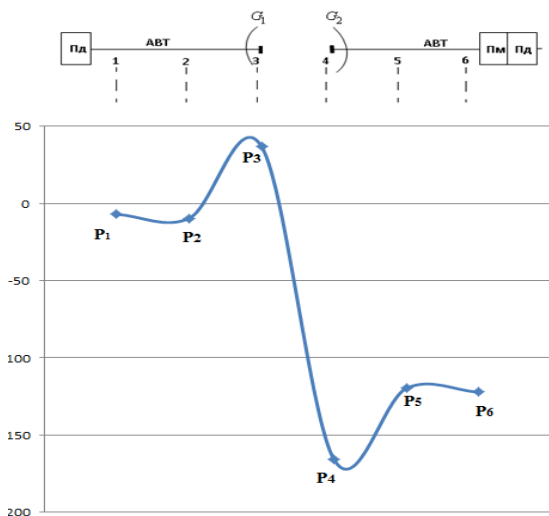


Рис. 4. Диаграмма уровней

Связь ОТУС с ТмГУС осуществляется через спутник «Экспресс АМЗ». Минусом такого решения является удорожание предоставляемых услуг абонентам.

Для схемы, изображённой на рис. 1, был произведён расчёт величины затуханий в спутниковом канале связи. Максимальная величина затуханий сигнала в атмосфере наблюдается в феврале и составляет 5,71 дБ. Затухания в ионосфере составляют 3 дБ. Максимальное затухание сигнала в спутниковом канале вызвано дождём в июле и составляет 43,89 дБ [3].

На рис. 5 представлена зависимость суммарной величины затуханий в спутниковом канале связи от месяца. Наибольшие потери наблюдаются в июле и составляют 46,53 дБ.

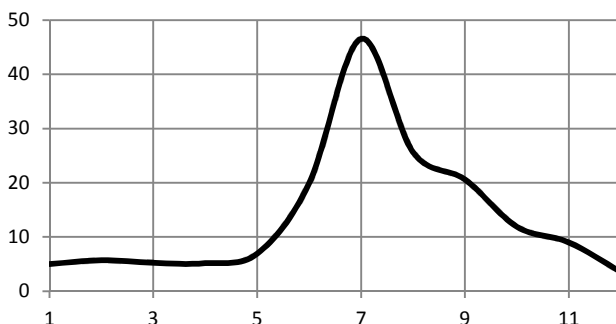


Рис. 5. Суммарные потери

В таблице приведены основные характеристики проектируемой системы коммутации стандарта GSM-1800. Данные по развитию системы были получены на основе анализа статистических данных аналогичных систем в городах со схожими условиями

Основные характеристики проектируемой системы коммутации стандарта GSM-1800

Год	Количество абонентов	Удельная нагрузка на абонента, Эрл	Нагрузка на MGW, Эрл	Количество потоков E1
2015	800	0.015	12	1
2016	1800	0.015	27	2
2017	2600	0.016	41.6	2
2018	3100	0.018	55.8	3
2019	3300	0.019	62.7	3

После 2019 г. рост количества абонентов прекратится и будет вирироваться вокруг 3300 абонентов. Следовательно, максимальная удельная нагрузка на MGW не превысит 70 Эрл, а количество пото-

ков E1 не превысит 4. Всё необходимое оборудование должно выбираться с учётом данных параметров.

Таким образом, помимо всего оборудования для функционирования сети необходимо:

- арендовать спутниковый канал связи;
- арендовать или установить ЦРРЛ.

Запуск сети сотовой связи стандарта GSM-1800 в г. Анадырь в большей степени связан с политикой оператора связи «Теле2 Рус» в создании единого покрытия для территории всей страны.

Библиографический список

1. Попов В.И. Основы сотовой связи стандарта GSM. – М.: Эко-Трендз, 2005. – 296 с.
2. Маковеева М.М. Радиорелейные линии связи: учебник для техникумов. – М.: Радио и связь, 1988. – 312 с.
3. Камнев В.Е., Черкасов В.В., Чечин Г.В. Спутниковые сети связи. – М.: Альпина Паблишер, 2004. – 538 с.
4. Чукотский автономный округ. – URL: https://ru.wikipedia.org/wiki/Чукотский_автономный_округ (дата обращения 19.10.2015).

СНИЖЕНИЕ ПИК-ФАКТОРА СИГНАЛОВ С OFDM С ПОМОЩЬЮ БЛОЧНОГО КОДИРОВАНИЯ

Студент А.О. Пукса

Омский государственный технический университет

Рассматривается метод понижения пик-фактора сигнала с OFDM-модуляцией с помощью блочного кодирования, приведены исследуемая структурная схема, а также результаты исследований. Дано сравнение системы с использованием блочного кодирования и без него.

Ключевые слова: OFDM, пик-фактор, блочное кодирование.

REDUCTION CREST FACTOR SIGNALS WITH OFDM BLOCK CODING

Student A.O. Puksa

Omsk State Technical University

This article describes the method of lowering the crest factor signal with OFDM modulation using block coding, a block diagram is shown under investigation, as well as the results of research. The comparison system using block coding and without.

Keywords: OFDM, peak factor, block coding.

Первый патент, связанный с OFDM-технологией, был зарегистрирован в 1970 г. Робертом Чэнгом, но широкое применение эта технология получила относительно недавно.

Несмотря на все плюсы и минусы данной технологии, OFDM – это ключ к решению проблем, связанных с архитектурами современных сетей связи.

Одним из главных преимуществ таких систем является значительная экономия частотного ресурса. На рис. 1, *а* представлен частотный спектр с использованием мультиплексирования с ортогональным частотным разделением каналов, а на рис. 1, *б* – с частотным разделением каналов. Также использование OFDM-технологии позволяет лучше противостоять межсимвольным помехам. При использовании OFDM возможно отдельно для каждой поднесущей применять разные схемы модуляции.

В данной статье будут представлены результаты моделирования в среде MatLab Simulink систем с использованием блочного кодирования и без него.

В настоящее время технология OFDM существует в следующих стандартах связи: IEEE 802.11, IEEE 802.20, IEEE 802.16 (WiMAX), IEEE 802.16e (Mobile WiMAX), WiBro, Flash-OFDM, LTE, MediaFLO, DVB-C, DVB-C2, DVB-T, DVB-T2, T-DMB, ADSL, ISDB-T, DRM, HIPERLAN/2.

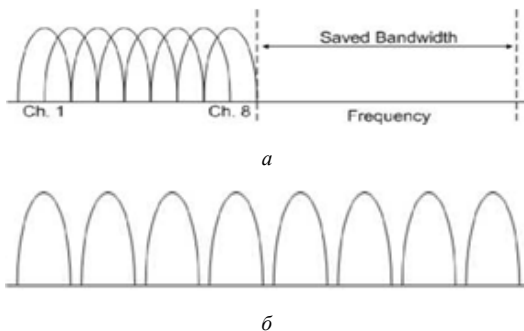


Рис. 1. Частотный спектр

Основным недостатком технологии является высокое значение пик-фактора передаваемых сигналов. «Он возникает из-за того, что OFDM-сигнал состоит из большого числа независимо модулированных по амплитуде и фазе гармоник. При их когерентном (или квази-когерентном) сложении возникают «пики» огибающей, которые характеризуются величиной PAPR (Peak-to-Average Power Ratio), т.е. отношения пиковой мощности сигнала к его средней мощности» [1].

Если не учитывать этот параметр при проектировании, то появится необходимость увеличивать динамический диапазон ЦАП, АЦП, а также усилителя мощности, а это, в свою очередь, приведет к усложнению системы, т.е. к её удорожанию.

Методы понижения пик-фактора сигнала условно можно разделить на 3 группы:

- блочное кодирование;
- ограничение амплитуды сигналов (клиппирование);
- вероятностные методы [1].

«Используя блочные коды, необходимо проанализировать количество сигналов с пик-фактором, не превышающим заданный. На основе этих данных определить необходимую избыточность кода.

Например, если 1/4 всех кодовых слов обладает пик-фактором, не превышающим заданный, то избыточность кода должна составлять 2 бита» [1].

Структурная схема системы с сигналами OFDM и кодером для снижения пик-фактора представлена на рис. 2.

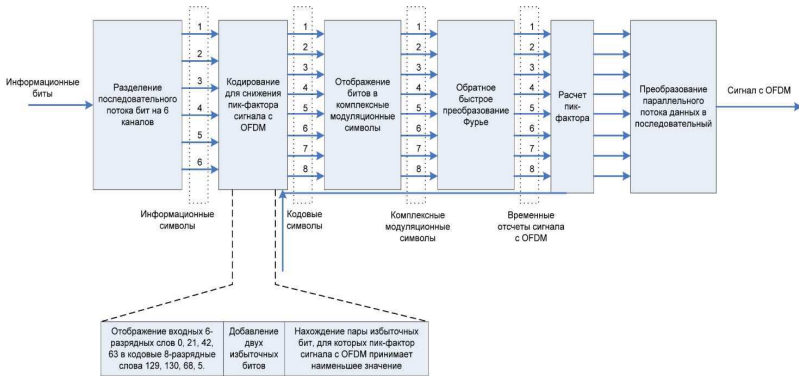


Рис. 2. Структурная схема системы с сигналами OFDM и кодером для снижения пик-фактора

«Декодировать такой код можно по шести младшим битам» [1]. Для этого приёмник после IFFT должен отбросить старшие два бита. На рис. 3 представлены схемы систем с сигналами OFDM, построенные в MatLab Simulink.

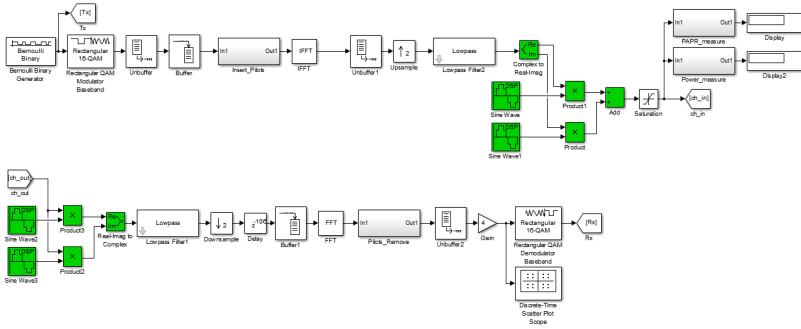


Рис. 3. Схемы для исследования системы с сигналами OFDM

На рис. 4 отображены зависимости величины BER от SNR для сигналов с OFDM с 8 поднесущими и ФМ-2.

Декодирование осуществлялось по младшим шести битам. Энергетический проигрыш системы с использованием блочного кодирования составил 1 дБ. Благодаря введенной избыточности код будет исправлять ошибки.

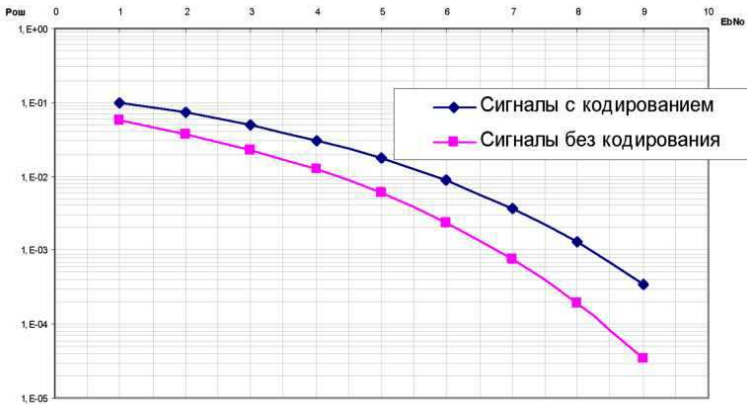


Рис. 4. Зависимость средней вероятности ошибки на бит от отношения сигнал/шум

Применение в системах связи технологии OFDM с использованием блочного кодирования позволяет уменьшить величину пик-фактора сигнала на 4,6 дБ. Снижение пик-фактора сигнала позволит более широко использовать технологию OFDM в современных системах связи, а также сделает процесс взаимодействия с другими стандартами более простым.

Научно исследовательская работа осуществлена в рамках выполнения магистерской диссертации. Научный руководитель – Д.А. Корнеев.

Библиографический список

1. Макаров С.Б., Рашич А.В. Применение блочного кодирования для снижения пик-фактора сигналов с OFDM // Тр. СПбГТУ. – СПб.: Изд-во Политехн. ун-та, 2008. – № 507. – С. 170–178.
2. Рашич А.В. Снижение ПИК-фактора случайных последовательностей многочастотных сигналов путем применения блочного кодирования и спектрально-эффективных методов манипуляции. – СПб., 2009. – 178 с.

О РАЗРАБОТКЕ УЧЕБНО-ЛАБОРАТОРНОГО КОМПЛЕКСА ДЛЯ ПОСТРОЕНИЯ СИСТЕМ ЗАЩИТЫ ИНФОРМАЦИИ ОТ НЕСАНКЦИОНИРОВАННОГО ДОСТУПА НА ОСНОВЕ «АККОРД-АМДЗ 5.5»

Студент А.В. Рангулов

Пермский национальный исследовательский
политехнический университет

Предлагается описание разработанного учебно-лабораторного комплекса на базе программно-аппаратного средства защиты информации от несанкционированного доступа Аккорд-АМДЗ. Применение данного комплекса в процессе подготовки специалистов по защите информации позволит изучать функциональные возможности и проводить исследования современных систем защиты информации.

Ключевые слова: информационная безопасность, несанкционированный доступ, контроллер, программно-аппаратный комплекс, персональный идентификатор, недеklarированные возможности.

ABOUT DEVELOPMENT OF TRAINING-LABORATORY COMPLEX FOR BUILDING SYSTEMS TO PROTECT INFORMATION FROM UNAUTHORIZED ACCESS BASED ON ACCORD-AMDZ 5.5

Student A.V. Rangulov

Perm National Research Polytechnic University

In this article the description of the developed educational and laboratory complex on the basis of a hardware-software security measure of information from unauthorized access of Akkord-AMDZ is considered. Application of this complex in the course of training of specialists in information security will allow to study functionality, to conduct researches of modern systems of information security.

Keywords: information security; unauthorized access; controller; computer appliance; user identifier; undocumented features.

Развитие необходимых практических навыков у будущих специалистов по защите информации является одной из актуальных задач улучшения системы их профессионального обучения.

За счет создания и совершенствования учебно-лабораторной базы на основе действующих программно-аппаратных средств защиты информации (СЗИ) от несанкционированного доступа (НСД) необходимо выработать требуемые компетенции у будущих специалистов в их профессиональной деятельности.

Согласно современным требованиям ФГОС ВПО по направлению подготовки 090303 «Информационная безопасность автоматизированных систем» выпускник должен иметь способность применять программные, программно-аппаратные и технические методы и СЗИ в распределенных автоматизированных системах [1].

В настоящий момент рынок сертифицированных СЗИ от НСД предлагает не менее десятка систем от различных производителей. В связи с возрастающими угрозами информационной безопасности с каждым годом перечень используемых средств постепенно пополняется более модифицированными решениями. Примером серии подобных средств защиты является программно-аппаратный комплекс от НСД – аппаратный модуль доверенной загрузки «Аккорд-АМДЗ», работающий на основе контроллера «Аккорд-5.5».

Термин «доверенная загрузка» означает загрузку операционной системы только после осуществления ряда контрольных процедур, таких как идентификация и аутентификация пользователей, проверка целостности как технических, так и программных средств [2].

Использование данного комплекса для решения учебно-лабораторных задач позволяет будущим специалистам наглядно познакомиться со средством от НСД, которое активно используется многими отечественными государственными и коммерческими организациями.

В составе комплекса поставляются аппаратные средства, включающие в себя одноплатный контролер, устанавливаемый в свободный слот материнской платы, персональный идентификатор, который представляет собой ТМ-идентификатор, а также съемник информации с контактным устройством, обеспечивающий взаимодействие между контроллером и персональным идентификатором пользователя. Помимо этого комплекс включает в себя программные средства, размещенные в энергонезависимой памяти контроллера.

СЗИ «Аккорд-АМДЗ» начинает работу сразу после выполнения кода системного BIOS-компьютера – до загрузки операционной системы и обеспечивает доверенную загрузку ОС, использующих одну из поддерживаемых файловых систем.

Комплекс представляет собой совокупность технических и программных средств, обеспечивающих следующие основные функции защиты от НСД ПЭВМ (АС) [2]:

- идентификация пользователя по персональному идентификатору до загрузки ОС;

- аутентификация пользователя по паролю длиной до 12 символов, вводимому с клавиатуры с защитой от раскрытия пароля до загрузки ОС;
- блокировка загрузки с отчуждаемых носителей (FDD, CD/DVD-ROM, ZIP, USB-накопителей и др.);
- контроль целостности технических, программных средств;
- доверенная загрузка системного и прикладного ПО при одновременной установке на дисках или в логических разделах диска ПЭВМ (PC) нескольких ОС;
- регистрация на ПЭВМ (PC) до 126 пользователей;
- регистрация контролируемых событий в системном журнале, размещенном в энергонезависимой памяти контроллера;
- физическая коммутация управляющих сигналов периферийных устройств в зависимости от уровня полномочий пользователя, позволяющая управлять вводом/выводом информации на отчуждаемые физические носители и устройства обработки данных;
- администрирование встроенного ПО комплекса;
- регистрация, сбор, хранение и выдача данных о событиях, происходящих в ПЭВМ (PC) в части системы защиты от несанкционированного доступа в ЛВС.

Комплекс обеспечивает выполнение основных функций защиты от НСД как в составе локальной ПЭВМ, так и на рабочих станциях ЛВС в составе комплексной системы защиты от НСД ЛВС, в том числе настройку, контроль функционирования и управление комплексом.

СЗИ НСД «Аккорд-АМДЗ» имеет сертификаты соответствия ФСТЭК по второму уровню контроля на отсутствие недеklarированных возможностей. Кроме того, данный комплекс может использоваться для создания автоматизированных систем до класса защищенности 1Б включительно, а также для защиты информации в информационных системах персональных данных до 1 класса включительно, в многопользовательском режиме работы [3].

Функциональные возможности, представленные в СЗИ НСД «Аккорд-АМДЗ», позволяют сформировать ряд учебно-лабораторных задач для успешного освоения принципов работы как самого комплекса, так и способов применения программно-аппаратных решений подобного класса при моделировании внутренних и внешних угроз объекта.

Схема размещения оборудования СЗИ НСД «Аккорд-АМДЗ» в учебной лаборатории приведена на рис. 1.

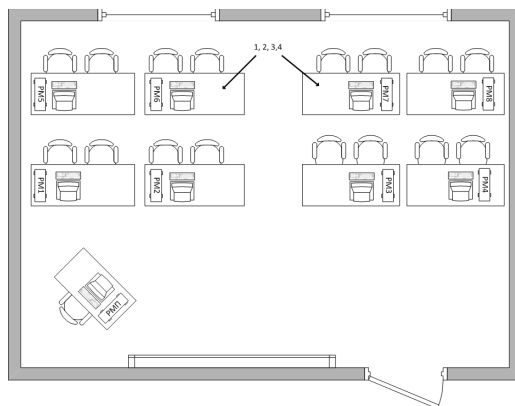


Рис. 1. Схема размещения программно-аппаратного комплекса «Аккорд-АМДЗ» в учебной лаборатории

Оборудование, используемое при проведении лабораторных занятий с использованием СЗИ НСД «Аккорд-АМДЗ»:

- рабочие места обучаемых: РМ1–РМ8;
- одноплатные контроллеры для РМ1–РМ8;
- съемники информации с контактным устройством;
- персональные идентификаторы пользователей (DS 199x («Touch memo») или USB-устройство ПСКЗИ ШИПКА);
- РМП – рабочее место преподавателя.

Учебно-лабораторный комплекс размещен в специализированной лаборатории кафедры автоматики и телемеханики, оборудованной необходимыми для выполнения учебных задач персональными компьютерами.

Поставляемое совместно с «Аккорд-АМДЗ» специальное ПО реализует дискреционный и мандатный методы разграничения доступа и позволяет администратору безопасности информации описать правила разграничения доступа на основе наиболее полного набора атрибутов доступа.

Разработанный учебно-лабораторный комплекс позволяет включить его в программу подготовки студентов по направлению «Информационная безопасность» специальности «Информационная безопасность автоматизированных систем», а также использовать в качестве дополнительной программы для повышения квалификации специалистов по защите информации. Применение программно-аппаратного комплекса «Аккорд-АМДЗ» в ходе лабораторных работ

позволяет подготовить студентов к решению следующих практических задач:

- установка аппаратной части комплекса (плата контроллера) в ПЭВМ;

- регистрация администратора безопасности информации с помощью идентификатора и пароля, а также регулирование следующих параметров: параметры пароля; доступ к устройствам, с помощью которого можно блокировать доступ отдельных пользователей к внешним устройствам; атрибуты доступа, в котором главный администратор устанавливает набор функции администрирования «подчиненным» администраторам;

- регистрация пользователей с помощью идентификатора и пароля, а также регулирование следующих параметров: режим блокировки (для временной блокировки конкретного пользователя системы), временные ограничения (установка ограничений на вход в систему в любой день недели);

- контроль как самой аппаратуры, так и целостности служебных областей жестких дисков, отдельных файлов, а также целостности реестра;

- обзор системного журнала, в который заносятся данные о сеансах работы пользователей с указанием номера идентификатора, а также все попытки несанкционированного доступа к ПЭВМ.

Таким образом, применение современного программно-аппаратного комплекса «Аккорд-АМДЗ» на учебно-лабораторных занятиях расширяет возможности профессиональной подготовки будущих специалистов по защите информации, а также способствует быстрой их адаптации на предприятиях, использующих данное СЗИ от НСД.

Библиографический список

1. Федеральный государственный образовательный стандарт высшего профессионального образования по направлению подготовки 090903 Информационная безопасность автоматизированных систем (квалификация (степень) «специалист») (утв. Приказом Министерства образования и науки РФ от 17 января 2011 г. № 60) // Доступ из справ.-правовой системы КонсультантПлюс.

2. Программно-аппаратный комплекс защиты информации от НСД для ПЭВМ (РС) «Аккорд-АМДЗ». Руководство администратора. – М., 2014. – 48 с.

3. Сертификат Гостехкомиссии №246/7 на комплекс СЗИ НСД Аккорд-АМДЗ №246/7.

ДЕКОДИРОВАНИЕ В КАНАЛАХ СВЯЗИ С ПАКЕТИРУЮЩИМИСЯ ОШИБКАМИ МЕТОДОМ ПОСИМВОЛЬНОГО ПЕРЕМЕЖЕНИЯ И С ИСПОЛЬЗОВАНИЕМ ПОРОЖДАЮЩЕГО ПОЛИНОМА $G(X^i)$

Студент В.П. Рогожин, канд. техн. наук, доцент В.И. Фрейман
Пермский национальный исследовательский
политехнический университет

В статье объектом изучения являются методы декодирования в каналах связи с пакетирующимися ошибками. Целью работы является разработка моделей декодеров в среде MatLab методами посимвольного перемежения и с использованием порождающего полинома $g(x)$. Проработаны устройства, осуществляющие синхронизацию декодера: тактовый ключ, преобразователь из параллельного кода в последовательный.

Ключевые слова: декодер, степень перемежения, порождающий полином, буферный регистр, триггер, устройство управления, тактовый ключ.

DECODING IN COMMUNICATION CHANNELS WITH STACKING ERRORS WITH USING METHOD OF SYMBOL-BY-SYMBOL INTERLEAVING AND GENERATING POLYNOMIAL $G(X^i)$

Student V.P. Rogozhin,
Ph.D. in Technical Sciences, Associate Professor V.I. Freyman
Perm National Research Polytechnic University

In this article the object's research is methods of decoding in network channels with stacking errors. The purpose's article is developing models of decoders in MATLAB by methods: character-oriented interlacing and with using generate polynomial $g(x)$. Devices are investigated which synchronize decoder: clock switch, converter from parallel to serial code.

Keywords: decoder, degree of interlacing, generate polynomial, buffer, trigger, ordering device, clock switch.

Известно, что сообщения в каналах связи практически никогда не передаются без помех. В связи с этим необходимо вводить средства для повышения помехоустойчивости – кодирование. Изучая поведение помехи, строятся математические модели передающихся пакетов. После воздействия помехи на передающийся пакет очень важно получить исходное сообщение – декодирование. Актуальность данной статьи обусловлена необходимостью изучения декодирования

и поведения пакетирующей помехи. В статье итоги декодирования представлены в таблицах переходов. Для исправления воздействий помех на канал связи используют декодирование, что позволяет восстановить исходный сигнал. Для декодирования информационных сообщений методом перемежения существует определенный параметр i – степень перемежения.

Рассмотрим обобщенную структуру декодера $(n \cdot i, m \cdot i)$ -кодов со степенью перемежения 2 в среде MatLab (рис. 1).

В приведенной структуре декодера выделяются i независимых декодеров. Коммутатор (Converter) (см. рис. 1) распределяет построчно посимвольно перемежаемые (n, m, d) -коды, из которых образован $(n \cdot i, m \cdot i)$ -код. Каждый из i декодеров исправляет пакеты ошибок длиной b и меньше.

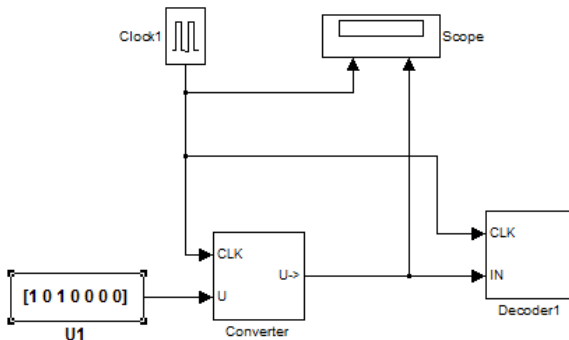


Рис. 1. Декодер циклического (n, m) -кода, исправляющий пакеты ошибок независимым декодированием перемежаемых кодов

Рассмотрим в качестве примера один из независимых декодеров кода $(7, 3, 4)$ с $g(x) = 1 \oplus x \oplus x^2 \oplus x^4$, исправляющий пакет ошибок длиной $b = 2$.

Декодируем $(14, 6)$ -код, исправляющий пакеты ошибок длиной $b \cdot i = 4$ и меньше с посимвольным перемежением степени 2 кода $(7, 3, 4)$, исправляющего пакеты длиной $b = 2$.

Общая структура декодера циклического кода $(14, 6)$ со степенью перемежения $i = 2$ приведена на рис. 2.

Раскроем подробно декодер $(7, 3, 4)$ -кода (рис. 3) и проанализируем его функционирование с помощью таблиц перехода и осциллограмм.

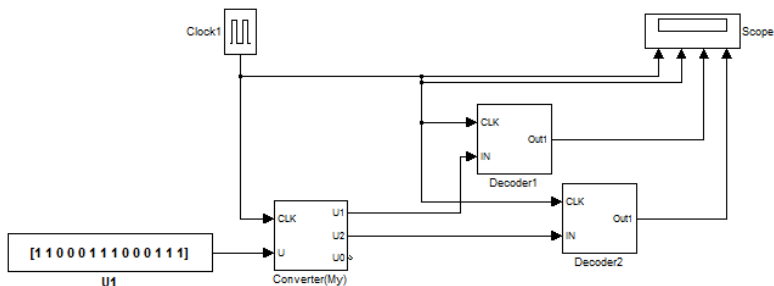


Рис. 2. Декодер циклического (14, 6)-кода, исправляющего пакеты ошибок длиной $b = 4$ и меньше методом независимого декодирования перемежаемых (7, 3, 4)-циклических кодов

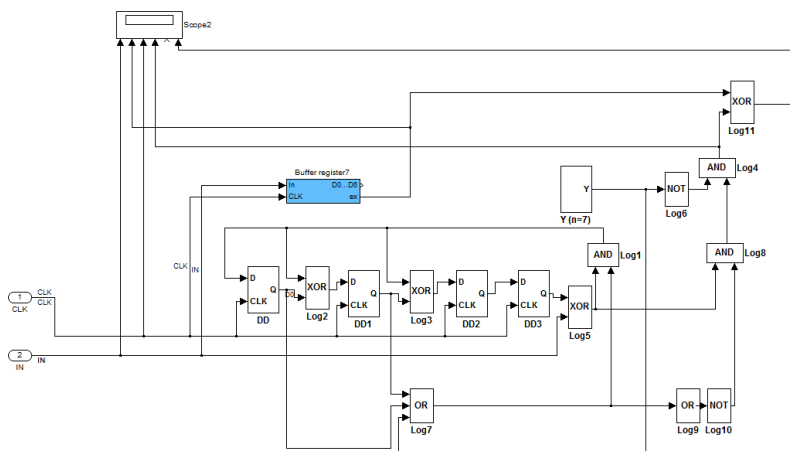


Рис. 3. Декодер циклического (7, 3, 4)-кода, исправляющего в составе декодера (см. рис. 1, 2) пакеты ошибок длиной $b = 2$ и меньше

Пусть вектор V'_1 на выходе коммутатора «Converter» (на входе IN) имеет вид: $V'_1 = V_1 \oplus e = 0011101 \oplus 0011000 = 000010$. Тогда ниже приводится таблица переходов (табл. 1) декодера при исправлении данного пакета.

Из таблицы переходов (см. табл. 1) видно, что на 10-м такте две левые ячейки ГС находятся в 0-м состоянии, а в двух правых ячейках ГС записан пакет ошибок. Начиная с 11-го такта, пакет ошибок выталкивается из ГС и складывается по модулю 2 с вектором V'_1 , т.е. происходит исправление ошибок.

Таким образом, если вектор V' кода (14, 6) поражен пакетом длиной $b = 4$, то этот пакет распределяется между двумя перемежаемыми кодами (7, 3, 4). При этом каждый из них исправляет свой пакет длиной $b = 2$, как это было показано выше [1].

Таблица 1

Таблица перехода декодеров

№	Вх	D_0	D_1	D_2	D_3
		0	0	0	0
1	0	0	0	0	0
2	0	0	0	0	0
3	0	0	0	0	0
4	1	1	1	1	0
5	1	1	0	0	1
6	0	1	0	1	0
7	0	0	1	0	1
8	0	1	1	0	0
9	0	0	1	1	0
10	0	0	0	1	1
11	0	0	0	0	1
12	0	0	0	0	0
13	0	0	0	0	0
14	0	0	0	0	0

1
1

На рис. 4 изображены осциллограммы, отражающие работу декодера. На осциллограммах изображены: на первой – входной вектор V' , на второй – вектор V' на выходе буфера, на третьей – тактовая частота, на четвертой – вектор ошибки «0011000», на пятой – исправленный вектор V' «0011101».

Пусть $i = 3$, циклический код строки (7, 3) (посимвольно перемежаемый код) задан следующим порождающим полиномом: $g(x) = 1 \oplus x \oplus x^2 \oplus x^4$. Тогда $g(x^3) = 1 \oplus x^3 \oplus x^6 \oplus x^{12}$ порождает код $(7 \cdot 3, 3 \cdot 3) = (21, 9)$, исправляющий пакеты ошибок длины $i \cdot b = 3 \cdot 2 = 6$ и меньше.

На рис. 5 представлена структура декодера (21, 9) циклического кода, реализующего способ посимвольного перемежения кода (7, 3, 4) со степенью перемежения $i = 3$.

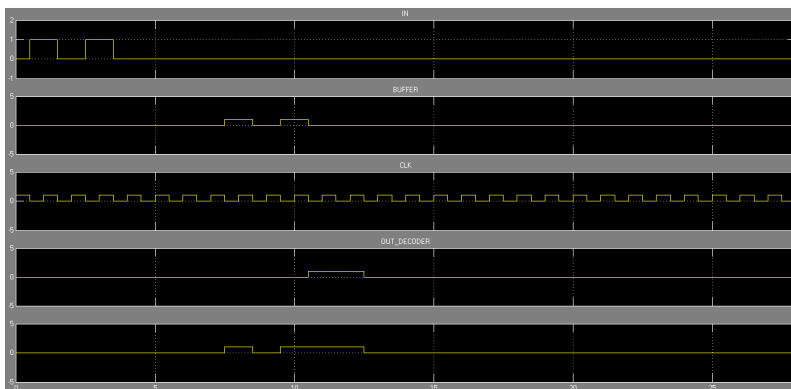


Рис. 4. Осциллограммы декодера циклического (7, 3, 4)-кода, декодера (см. рис. 2) пакеты ошибок длиной $b = 2$ и меньше

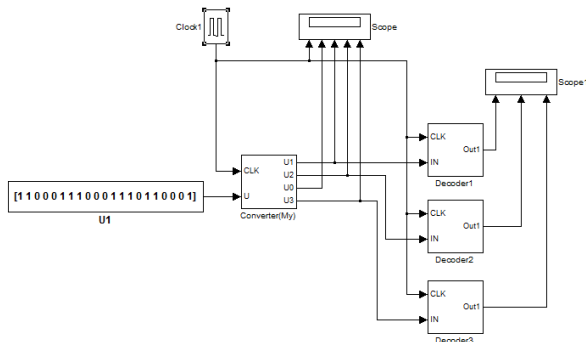


Рис. 5. Декодер посимвольного перемежения ($i = 3$)

Схемы декодеров (decoder1, decoder2, decoder3) для рассматриваемого примера ничем не отличаются от схемы декодера при $i = 2$ (см. рис. 3).

Структура декодера, реализующего указанный способ исправления пакетов ошибок длиной $b \cdot i$ и меньше в коде $(n \cdot i, m \cdot i)$, построенном описанным выше способом посимвольного перемежения (n, m) -кодов с помощью порождающего полинома $g(x^i)$, принципиально не отличается от структуры декодера, приведенного на рис. 1.

На рис. 6 приведена структура декодера кода (14, 6) с порождающим полиномом $g(x^2) = 1 \oplus x^2 \oplus x^4 \oplus x^8$, аналогичная декодеру, представленному на рис. 1.

Проанализируем работу декодера с помощью таблицы переходов (табл. 2) на примере исправления пакета ошибок длиной $b \cdot i = 4$.

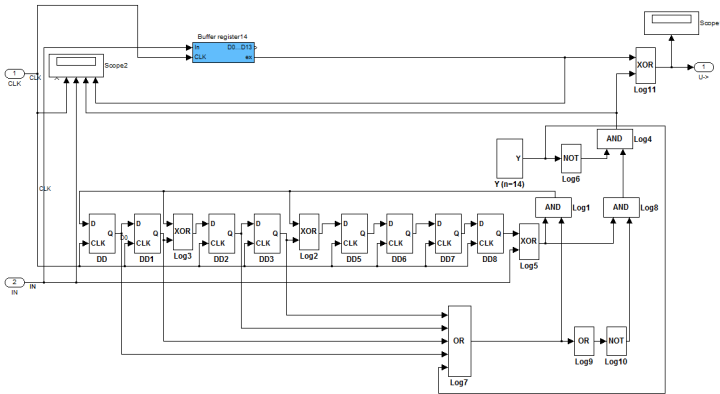


Рис. 6. Декодер циклического (14, 6)-кода, исправляющего пакеты ошибок длиной 4 и меньше с использованием полинома $g(x^2) = 1 \oplus x^2 \oplus x^4 \oplus x^8$

Из таблицы переходов (см. табл. 2) видно, что пакет ошибок длиной 4 поразил 4 старших разряда принятого и записанного в БР кодового вектора.

Таблица 2

Таблица переходов

№	Вх.	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	
1	1	1	0	1	0	0	0	0	0	
2	1	1	1	1	1	1	1	0	0	
3	0	1	1	1	1	1	1	1	0	
4	1	1	0	0	1	0	1	1	1	
5	0	1	1	1	0	0	0	1	1	
6	0	1	1	0	1	1	0	0	1	
7	0	1	1	0	0	0	1	0	0	
8	0	0	1	1	0	0	0	1	0	
9	0	0	0	1	1	0	0	0	1	
10	0	1	0	1	1	0	0	0	0	
11	0	0	1	0	1	1	0	0	0	
12	0	0	0	1	0	1	1	0	0	
13	0	0	0	0	1	0	1	1	0	
14	0	0	0	0	0	1	0	1	1	
15	0	0	0	0	0	0	1	0	1	1
16	0	0	0	0	0	0	0	1	0	1
17	0	0	0	0	0	0	0	0	1	0
18	0	0	0	0	0	0	0	0	0	1
19	0	0	0	0	0	0	0	0	0	

После первых 14 тактов в генераторе синдромов (ГС) четыре младших разряда находятся в 0-м состоянии, а в четырех старших разрядах записан пакет ошибок. Начиная с 15-го разряда (вторые $n = 14$ (15...28) тактов), происходит выталкивание пакета ошибок и сложение по модулю 2 с вектором на выходе БР[2].

Как отмечалось в первом разделе, существуют неподдающиеся исправлению пакеты ошибок, в частности, расположенные в i старших и $b - i$ младших разрядах кодового слова, так как, по сути, они представляют собой пакеты длиной существенно больше, чем $k/2$.

Итак, были рассмотрены способы построения структур декодеров циклических кодов. Были разработаны модели декодеров в среде MatLab методами: посимвольного перемежения и с использованием порождающего полинома $g(x^j)$, проработаны устройства, осуществляющие синхронизацию декодера: тактовый ключ, преобразователь из параллельного кода в последовательный.

Библиографический список

1. Кон Е.Л., Фрейман В.И. Теория электрической связи. Помехоустойчивая передача данных в информационно-управляющих и телекоммуникационных системах: модели, алгоритмы, структуры: учеб. пособие. – Пермь: Изд-во Перм гос. техн. ун-та, 2007. – 317 с.
2. Пахомов Г.И., Фрейман В.И. Теория электрической связи. Основные понятия: учеб. пособие. – Пермь : Изд-во Перм гос. техн. ун-та, 2007. – 115 с.

МЕТОДЫ ЗАЩИТЫ ОТ ИЗМЕНЕНИЙ И ДИЗАССЕМБЛИРОВАНИЯ .NET ПРИЛОЖЕНИЙ

Студент Э.Д. Салимзебаров,
канд. техн. наук, доцент А.Н. Кокоулин
Пермский национальный исследовательский
политехнический университет

В работе рассматриваются декомпиляция .Net приложений и различные способы защиты, такие как обфускация, упаковщики, контроль запуска в отладке и применение электронных ключей для защиты контроля запуска. Также оценивается степень надежности и устойчивости ко взлому рассмотренных средств защиты.

Ключевые слова: .Net-приложения, обфускация, способы защиты.

METHODS FOR .NET APPLICATIONS PROTECTION AND DISASSEMBLING

Student E.D. Salimzebarov,
Ph.D. in Technical Sciences, Associate Professor A.N. Kokoulin
Perm National Research Polytechnic University

The paper considers .Net applications decompilation process and different methods of protection against decompilation such as obfuscation, packers, launch control and electronic keys. Also we estimate the level of reliability of these protection approaches.

Keywords: .Net application, obfuscation, methods of protection.

Введение. На сегодняшний день популярность и усовершенствование .Net платформы обуславливают ее широкое распространение в настольных и серверных Windows-системах. В связи с активной политикой Microsoft по развитию и внедрению платформы .Net все больше разработчиков ведут разработку приложений для этой платформы, и в новых версиях Windows runtime-модули .Net Framework актуальных версий уже предустановлены в операционной системе. В связи с этим проблема защиты приложений от воздействия злоумышленников актуальна.

Декомпиляция и дизассемблирование. Рассмотрим независимо друг от друга задачу дизассемблирования и задачу декомпиляции программ. Под декомпиляцией понимается построение программы на языке высокого уровня, эквивалентной исходной программе на языке

низкого уровня (языке Ассемблера). Под дизассемблированием понимается построение программы на языке Ассемблера, эквивалентной исходной программе в машинном коде. Программа в машинном коде представляется либо в виде исполняемого модуля в стандартном для целевой операционной системы, либо в виде дампа содержимого памяти, либо в виде трассы исполнения программы.

При дизассемблировании выполняется трансляция исполняемого файла, представляемого в виде набора машинных команд, в программу на языке Ассемблера. При декомпиляции программа с представления низкого уровня транслируется в представление высокого уровня. Дальнейшим этапом повышения уровня абстракции программы может быть рефакторинг.

Чаще всего дизассемблер используют для анализа программы (или ее части), исходный текст которой неизвестен, – с целью модификации, копирования или взлома. Реже – для поиска ошибок (багов) в программах и компиляторах, а также для анализа и оптимизации создаваемого компилятором машинного кода.

Стоит отметить, что .Net-приложения компилируются на язык СIL, который является настоящим «родным» языком для платформы .NET. При создании .NET-сборки с помощью того или иного языка (C#, VB, COBOL.NET и т.д.) соответствующий компилятор всегда преобразует исходный код на этом языке в код на СIL. Так же и декомпиляторы уже работают с языком СIL, что позволяет восстанавливать исходный код программы чрезвычайно точно.

Рассмотрим рис. 1, где представлены исходный код и код, открытый декомпилятором .Net. По сравнению с реверсом и отладкой Win32 приложений задача декомпиляции .Net доступна даже мало-квалифицированному специалисту.

Защита от модификаций. Пожалуй, самый распространенный метод защиты – это применение обфускаторов [2]. Обфускатор – приведение исходного текста или исполняемого кода программы к виду, сохраняющему ее функциональность, но затрудняющему анализ, понимание алгоритмов работы и модификацию при декомпиляции.

Некоторые обфускаторы предоставляют функциональность, которая способна защитить приложение от модификаций. Такой механизм часто называют *tamper protection* или *tamper defense*. В этом случае программа будет корректно выполняться только в том случае, если она не была изменена злоумышленником. Если какие-то

изменения были, программа просто перестанет работать либо будет выводить некоторое сообщение о недопустимости использования модифицированной программы.

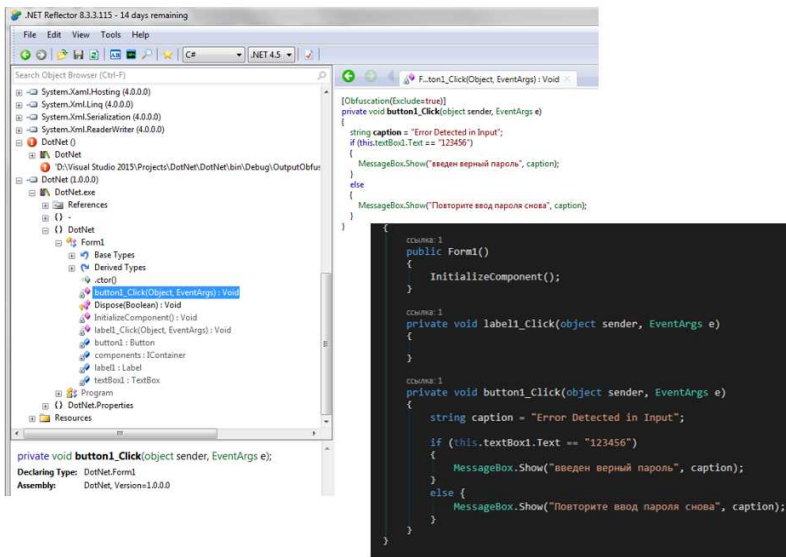


Рис. 1. Пример кода программы, полученного декомпилятором

Хотя подобный метод помогает создать иллюзию защиты, но это не значит, что вскрыть скомпилированное приложение будет сложно. На данный момент существует много различных так называемых «деобфускаторов» – пример такого «CodeWall», которые за пару кликов мыши восстановят исходный код программы. Выходом может служить только ручная настройка со своими методами, которых хакерские программы воспринимать не будут.

Защита от дизассемблирования. Несмотря на то, что в процессе обфускации структура исполняемого кода программы претерпела серьезные изменения, полученный код в результате работы дизассемблирования [1] будет чрезвычайно сложен и запутан, что потребует долгих часов разбора.

Поскольку злоумышленник ищет легкие пути обхода, он наверняка не думает сразу запускать дизассемблер и пытаться что-то разобрать, а, воспользовавшись уже готовым ПО, таким как «.Net Reflector» [3], получит более читаемый код и иногда полностью восстановит

ленный, но перед этим обязательно придется деобфусцировать приложение. Пример работы обфускатора представлен на рис. 2.

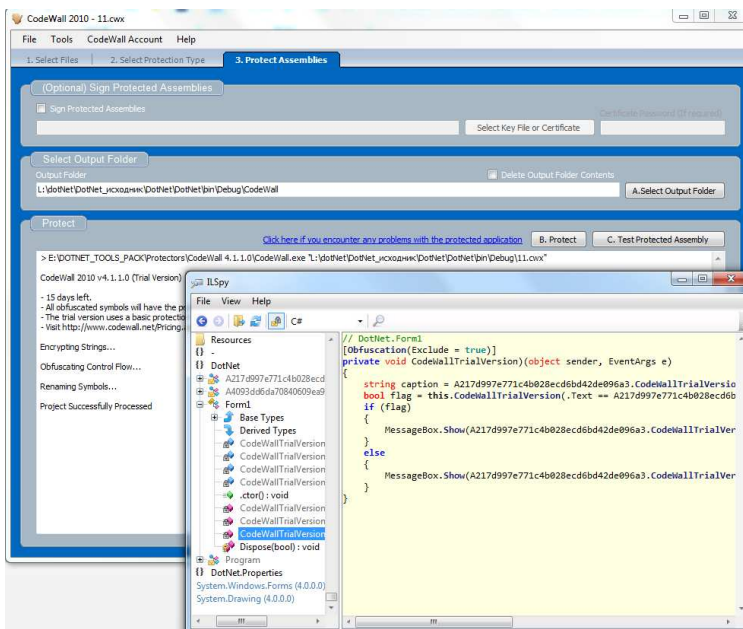


Рис. 2. Пример работы программы деобфускации

Исходя из такой ситуации, можно посоветовать применять различные обертки и упаковщики. Примером обертки для .Net-приложений может служить «win32». Она не позволит воспользоваться специальным ПО для взлома .Net-приложений, что заставит злоумышленника воспользоваться неудобным для него дизассемблером.

В дополнение к тому, что дизассемблер будет видеть код в машинных кодах, к тому же он еще способен запустить приложение под отладкой, а вот это уже очень серьезная угроза безопасности. Решением данной проблемы является установка так называемых «breakload». Это простой вариант вставки в тело каждого метода дополнительного фрагмента кода, который содержит некоторую некорректную или несуществующую инструкцию. При загрузке метода, которому добавлен такой фрагмент, дизассемблер не сможет распознать эту инструкцию и, соответственно, не сможет загрузить тело метода для анализа злоумышленником. А для того, чтобы данная инструкция не влияла на

процесс выполнения программы, ее следует оградить инструкциями безусловных переходов таким образом, чтобы она никогда не смогла получить управление. Такой метод, безусловно, осложнит жизнь злоумышленнику и заставит его копаться в компилированных кодах

Более продвинутый способ может выглядеть следующим образом. Код метода шифруется с помощью некоторого алгоритма шифрования, и результат шифрации помещается либо в ресурсы, либо вовсе в отдельный файл. Исходное тело метода замещается кодом некоторого загрузчика, который знает, где находится зашифрованный фрагмент кода, и знает способ, как его расшифровать. При вызове такого метода загрузчик получает управление, он дешифрует реальный код и выполняет его. Таким образом, при дизассемблировании исследовать этот код будет абсолютно невозможно.

Также стоит отметить, что защитить программный продукт можно с помощью HASP-ключей, которые будут намного надежнее всяких программных методов. Но это не означает, что их также невозможно обойти. Для обхода HASP-ключей используются различные эмуляторы подключения ключа. Это намного трудозатратнее и требует высокой квалификации. Примером программы для защиты с помощью HASP может служить HASP SRM Vendor Suite (рис. 3).

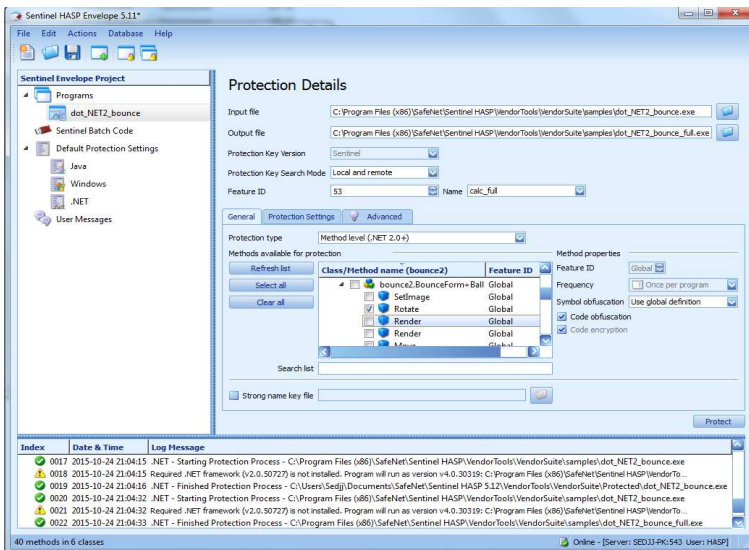


Рис. 3. Пример программы для защиты с HASP

Заключение. В данной статье были рассмотрены различные методы взлома и некоторые методы для защиты .Net -приложений. Подводя итоги, можно сказать, что на данный момент вариантов взлома довольно много, что существенно усложняет жизнь разработчиков .Net-приложений. Им приходится придумывать различные методы защиты своего ПО, иначе затраченные деньги уходят в никуда. Но стоит также отметить, что существуют довольно серьезные методы защиты ПО. Это не дает надежды разработчикам, и, заплатив за лицензию небольшие деньги или дописав самому модуль защиты из исходников, которые выложены в Open source, можно не переживать, что любой желающий сможет взломать программу. Подводя итоги, можно сказать, что разработка под .Net-платформу приносит много проблем при попытке заработать денег на своем ПО, и поэтому рекомендуется несколько раз подумать, прежде чем приступить к работе.

Библиографический список

1. SecurityLab [Электронный ресурс] // Реверс-инжиниринг NET-приложений. Ч. 1: Введение. – URL: <http://www.securitylab.ru/analytics/439107.php> (дата обращения: 24.11.15).
2. SecurityLab [Электронный ресурс] // Обратная инженерия с помощью Reflector. Ч. 1. – URL: <http://www.securitylab.ru/analytics/451094.php> (дата обращения: 24.11.15).
3. Kaimi. блог [Электронный ресурс] // Патчинг софта, накрытого .NET Reactor'ом, на примере ActualSpamPro. – URL: <http://kaimi.ru/2010/11/net-reactor-actualspampro/> (дата обращения: 24.11.15).
4. Scientific World [Электронный ресурс] // Безопасность баз данных oracle в многозвенных информационных системах. – URL: <http://www.sworld.com.ua/index.php/ru/technical-sciences-m115/informatics-computer-science-and-automation-m115/25323-m115-237> (дата обращения: 25.11.15).
5. Даденков С.А., Кон Е.Л. Оценка степени влияния некоторых факторов на производительность LonWorks сети // Образовательные ресурсы и технологии. – 2014. – № 2(5). – С. 72–76.
6. СибАК [Электронный ресурс] // Обзор сетевого протокола teredo на основе стандарта rfc 4380. – URL: <http://sibac.info/19640> (дата обращения: 24.11.15).

АНАЛИЗ ФУНКЦИОНАЛЬНЫХ ВОЗМОЖНОСТЕЙ СВОБОДНО РАСПРОСТРАНЯЕМЫХ SNMP-МЕНЕДЖЕРОВ

Студент С.Ю. Тиунов, старший преподаватель А.В. Гаврилов
Пермский национальный исследовательский
политехнический университет

В данной статье анализируются функции бесплатных SNMP-менеджеров и возможность их применения для обучения студентов основам управления и мониторинга компьютерных сетей по протоколу SNMP.

Ключевые слова: SNMP-менеджер, MIB, SNMP, TRAP.

THE ANALYSIS OF THE FUNCTIONAL CAPABILITIES OF FREE SNMP MANAGERS

Student S.Yu. Tiunov, Senior Lecturer A.V. Gavrilov
Perm National Research Polytechnic University

In this article functions of free SNMP of managers and possibility of their application for training of students in bases of control and monitoring of computer networks according to the SNMP protocol are analyzed.

Keywords: SNMP-manager, MIB, SNMP, TRAP.

Введение. Современные инфокоммуникационные системы представляют собой совокупность большого числа сложного, территориально разнесенного телекоммуникационного оборудования. Мониторинг и управление таким оборудованием реализуются системами управления с использованием специализированных протоколов управления. Наиболее распространенным протоколом управления является простой протокол сетевого управления SNMP. Студенты, обучающиеся по направлению подготовки 11.04.02 (210700) «Инфокоммуникационные технологии и системы связи», знакомятся с вопросами управления инфокоммуникационными сетями при изучении дисциплины «Современные принципы и технологии управления инфокоммуникационными сетями». В ходе выполнения лабораторных работ по указанной дисциплине студенты подробно изучают процесс взаимодействия агент-менеджера по протоколу SNMP. Поскольку на рынке в данный момент присутствует большое число менеджеров SNMP, то возникла необходимость в анализе современных программных продуктов, реализующих функции менеджера SNMP. Наиболее богатым функциона-

лом обладает программный продукт фирмы Castle Rock Computing – сетевой менеджер SNMPc Enterprise [1]. Однако весьма существенная стоимость данного продукта не позволяет его использовать в учебном процессе. Таким образом, задача поиска аналогичного свободного распространяемого менеджера SNMP, который можно использовать в образовательных целях, является актуальной и важной.

В данной работе был произведен анализ функциональных возможностей свободно распространяемых версий SNMP-менеджеров, таких как LorientPro Free Edition, iReasonig Free Personal Edition, OiDVieW SNMP MIB Browser, а также выполнено сравнение их с менеджером SNMPc Enterprise. Дополнительным ограничением при поиске SNMP-менеджеров являлось требование работы менеджера под операционной системой MS Windows, так как на компьютерах в лаборатории установлено лицензионное ПО Windows XP. По этой причине в обзор не вошли такие распространенные средства мониторинга, как Zabbix и net-snmp.

Анализ функциональных возможностей SNMP-менеджеров.

Анализ функциональных возможностей программного обеспечения SNMPc Enterprise позволил сформулировать следующие наиболее важные требования к SNMP-менеджерам:

1. Число узлов в дереве каталогов. Эта функция отображает какое количество устройств обнаруживает менеджер, а также показывает, какого масштаба сеть может обслужить менеджер.

2. Максимальное количество контролируемых узлов (PING, SNMP-опрос). Эта функция показывает количество узлов, которое менеджер может обслужить посредством ping-, snmp-запросов, и какого масштаба сеть может обслужить менеджер.

3. Число MIB-объектов в базе данных SNMP. Эта функция показывает количество переменных MIB, с которым менеджер может работать одновременно. Отображает количество переменных устройств, которые может контролировать менеджер.

4. MIB-компилятор. Эта функция позволяет прописывать модули MIB для частного устройства и добавлять его в MIB-базу менеджера без дополнительного ПО (использование какого-либо другого компилятора).

5. Просмотр модулей MIB и MIB-дерево. Эта функция позволяет просматривать MIB и отображает MIB в стандартном общепринятом виде. Влияет на удобство работы с менеджером и выбор нужных переменных для работы с устройствами.

6. Карта топологии сети. Эта функция позволяет наглядно видеть устройства и связи между ними, т.е. просматривать топологию сети, количество сетей и подсетей.

7. Прием и история SNMP-TRAP. Функция сообщает об аварийных ситуациях в устройстве и хранит их какой-то промежуток времени. Это могут быть сигналы, сообщающие о событиях: переназначение портов в коммутаторе или его перезапуск, отправка сигнала аварии от источника бесперебойного питания о переходе в режим работы от батареи, сигналы от различных датчиков, например вскрытия и т.д.

8. Поддержка IPv6. Эта функция показывает, может ли менеджер работать с устройствами IP, адрес которых задан посредством IPv6.

9. Версия протокола SNMP, поддерживаемая менеджером. Эта функция отображает, сколько версий из трех поддерживает менеджер. Таким образом, в зависимости от потребностей в различных версиях SNMP будет выбираться менеджер для мониторинга сети.

10. Модуль обнаружения SNMP. Функция позволяет посредством SNMP-запроса выяснить, есть ли SNMP-служба на устройстве и включена ли она.

11. Модуль обнаружения ping. Функция позволяет проверить, есть ли связь с необходимым нам устройством, без применения дополнительных средств.

12. График изменений переменных. Функция позволяет просматривать изменение значений MIB-переменных в виде графика в режиме реального времени. Таким образом, наглядно видна динамика изменения значений.

13. Фильтр принимаемых уведомлений. Функция позволяет просматривать только необходимые администратору события, выбирать их из множества произошедших событий, что упрощает мониторинг сети и в случае аварийной ситуации быстрее принять меры по ее устранению.

14. Получатель уведомлений. Прослушивает сообщения, уведомления и формирует сообщения-отклики, когда приходит сообщение с PDU Inform.

15. Фильтр ловушки SNMP. Используется для отсеивания малоинформативных уведомлений и получения только важных сообщений.

16. Команды GET, GetNext, GetBulk, SET. Команды для просмотра переменных по MIB-дереву. Команда Set позволяет менеджеру изменять значения какого-либо объекта. С помощью команды Set и происходит собственно управление устройством. Агент должен «понимать» смысл значений объекта, который используется для

управления устройством, и на основании этих значений выполнять реальное управляющее воздействие – отключить порт, присписать порт определенной линии VLAN и т.п.

С учетом указанных требований были найдены сопоставимые по функционалу менеджеры: LorientPro Lite Edition [2], iReasoning MIB browser [3], OiDViEW SNMP MIB Browser [4].

Результаты анализа функциональных возможностей менеджеров приведены в таблице.

Сравнение параметров SNMP менеджеров

Функции	SNMPC Enterprise	LorientPro Lite Edition	iReasoning MIB browser	OiDViEW SNMP MIB Browser
Число узлов в дереве каталогов	Не ограничено	Не ограничено	Не ограничено	Не ограничено
Максимальное количество контролируемых узлов (PING, SNMP-опрос)	Не ограничено	10	Не ограничено	Не ограничено
Число MIB-объектов в базе данных SNMP	Не ограничено	Не ограничено	10	Не ограничено
MIB-компилятор	Есть	Есть	Есть	Есть
Просмотр MIB и MIB-дерева	Есть	Есть	Есть	Есть
Карта топологии сети	Есть	Есть	Нет	Нет
Прием и история SNMP-TRAP	Есть	Есть	Есть	Есть
Поддержка IPv6	Есть	Нет	Есть	+модуль
Версия протокола	SNMPv1,2,3	SNMPv1,2,3	SNMPv1,2	SNMPv1,2,3
Модуль обнаружения SNMP	Есть	Первые 10 обнаруженных узлов	Есть	Есть
Модуль обнаружения ping	Есть	Первые 10 обнаруженных узлов	Нет	Есть
График изменений переменных	Есть	Нет	Нет	Есть
Фильтр принимаемых уведомлений	Есть	Нет	trap-filter	Есть
Получатель уведомлений	+	+	+	+
Фильтр ловушки SNMP	+	-	-	Пробный период
Команды GET, GetNext, GetBulk	+	+	+	+
Команда SET	+	+	+	-

На основе обзора SNMP-менеджеров можно подвести итог и выделить менеджеры, которые наиболее подходят для ознакомления и обучения, мониторинга сети с помощью SNMP.

Из таблицы видно, что наибольший функционал необходимый для целей обучения, имеет SNMPc Enterprise, но так как данная программа является платной и время ее использования ограничено, ее использование не всегда удобно.

Следующий менеджер по количеству функций – OidVIEW SNMP MIB Browser, он отличается от SNMPc Enterprise отсутствием карты топологии сети и команды «SET», а также пробным временем использования фильтра ловушки SNMP.

Остальные SNMP-менеджеры схожи по своему функционалу, какой-то уступает первым двум описанным менеджерам в большей степени, какой-то в меньшей.

Заключение. В работе был выполнен анализ функциональных возможностей свободно распространяемых SNMP-менеджеров. Определены два наиболее развитых менеджера. Однако ни один из них не реализует все функции, определенные при постановке задачи. Таким образом, при проведении лабораторных работ по курсу «Современные принципы и технологии управления инфокоммуникационными сетями» следует использовать оба выбранных менеджера, тем самым комбинируя их функционал и знакомя студентов со всеми функциональными возможностями мониторинга сети с помощью SNMP.

Библиографический список

1. SNMPc Enterprise – Products – Castle Rock Computing [Электронный ресурс]. – URL: <https://www.castlerock.com/products/snmpc/> (дата обращения: 30.04.2015).

2. LorientPro software editions [Электронный ресурс]. – URL: http://www.lorientpro.com/Products/Features/Table_Editions_EN.php (дата обращения: 30.04.2015).

3. MIB Browser [Электронный ресурс]. – URL: <http://ireasoning.com/mibbrowser.shtml> (дата обращения: 30.04.2015).

4. SNMP MIB Browser Overview [Электронный ресурс]. – URL: <http://www.oidview.com/mibbrowser.html> (дата обращения: 30.04.2015).

ОБЗОР СИСТЕМ ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ СЕТЕЙ СВЯЗИ

Студент Д.В. Федотов

Пермский национальный исследовательский
политехнический университет

Актуальность данной статьи в том, что она должна помочь начинающему пользователю в выборе системы имитационного моделирования сетей связи. Объектом изучения в статье являются системы имитационного моделирования сетей связи. Целью работы является выявление основных преимуществ и недостатков наиболее распространенных систем имитационного моделирования сетей связи. В результате проведенного обзора выявлены основные недостатки и преимущества наиболее распространенных систем моделирования сетей связи, определены специфика и особенности данных систем.

Ключевые слова: сеть связи, режим симуляции, моделирование сетей связи, графический интерфейс, оборудование сетей связи.

REVIEW OF SIMULATION NETWORK CONNECTION

Student D.V. Fedotov

Perm National Research Polytechnic University

The relevance of this article is that it should help the novice user in selecting the system of simulation networks, which is most suitable for the model network. The object of study of this article are of simulation networks. The aim is to identify the main advantages and disadvantages of the most common systems simulation networks. As a result of this review revealed the main advantages and disadvantages of the most common modeling systems networks, what is the specific nature of these systems.

Keywords: telecommunication network, simulation mode, modeling communications networks, graphical user interface, equipment networks.

При построении имитационных моделей сети связи перед пользователем стоит вопрос, какой системой имитационного моделирования воспользоваться для построения своей модели. Актуальность данной статьи в том, что она должна помочь начинающему пользователю в выборе той системы имитационного моделирования сетей связи, которая наиболее подходит для его модели сети связи.

Существует достаточно большое количество симуляторов и эмуляторов для построения моделей сетей связи. В данной статье показаны наиболее популярные инструменты, которые решают эту задачу. Информация будет полезна тем, кто изучает сетевые технологии или просто увлекается имитационным моделированием сетей связи.

Вначале немного терминологии.

Симуляторы имитируют некий набор команд, он шит, и стоит только выйти за рамки, сразу получим сообщение об ошибке. Классический пример – Cisco Packet Tracer.

Эмуляторы же, напротив, позволяют проигрывать (выполняя байт трансляцию) образы (прошивки) реальных устройств, зачастую без видимых ограничений. В качестве примера – GNS3/Dynamips.

Первым в данной статье рассмотрим Cisco Packet Tracer.

Cisco Packet Tracer. Внешний вид симулятора Cisco Packet Tracer представлен на рис. 1.

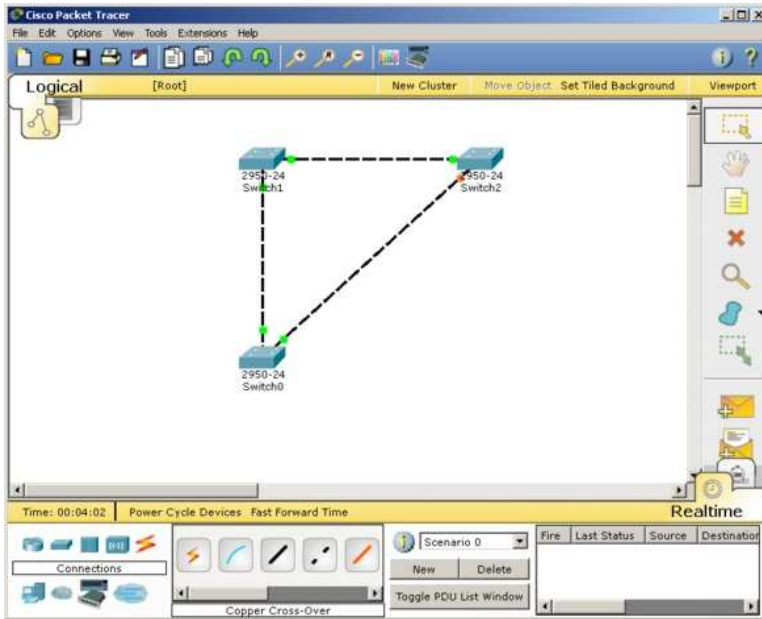


Рис. 1. Cisco Packet Tracer

Cisco Packet Tracer – это симулятор сети, созданный компанией Cisco. Данное приложение позволяет строить сети на разнообразном оборудовании в произвольных топологиях с поддержкой разных протоколов. Программное решение Cisco Packet Tracer позволяет имитировать работу различных сетевых устройств: маршрутизаторов, коммутаторов, точек беспроводного доступа, персональных компьютеров, сетевых принтеров, IP-телефонов и т.д. Работа с интерактивным

симулятором дает ощущение настройки реальной сети, состоящей из десятков или даже сотен устройств. Настройки, в свою очередь, зависят от характера устройств: одни можно настроить с помощью команд операционной системы Cisco IOS, другие – за счет графического веб-интерфейса, третьи – через командную строку операционной системы или графические меню. Благодаря такому свойству Cisco Packet Tracer, как режим визуализации, пользователь может отследить перемещение данных по сети, появление и изменение параметров IP-пакетов при прохождении данных через сетевые устройства, скорость и пути перемещения IP-пакетов. Анализ событий, происходящих в сети, позволяет понять механизм ее работы и обнаружить неисправности. Этот симулятор доступен как для Windows, так и для Linux [1].

Его плюсы – дружелюбность и логичность интерфейса. Кроме этого в нем удобно проверять работу разных сетевых сервисов, вроде DHCP/DNS/HTTP/SMTP/POP3 и NTP. И один из самых интересных плюсов – это возможность перейти в режим симуляции и увидеть перемещения пакетов с замедлением времени.

Минус в том, что иногда могут проявляться разнообразные зависания программы, которые исправляются только перезапуском программы.

Итак, данный симулятор имеет массу плюсов и один небольшой недостаток.

GNS3. Следующим в данном обзоре будет GNS3, внешний вид которого представлен на рис. 2.

Он представляет собой графический интерфейс (на Qt) для эмулятора dynamips. Свободный проект, доступен под Linux, Windows и Mac OS X. Но большинство его функций, призванных улучшить производительность, работают только для Linux, 64-битная версия также работает только для Linux. Это эмулятор, который работает с настоящими прошивками IOS. Для того чтобы им пользоваться, у вас должны быть прошивки. Например, если вы купили маршрутизатор Cisco, то с него можно их и вытащить. К нему можно подключать виртуальные машины VirtualBox или VMware Workstation и создавать достаточно сложные схемы, при желании можно пойти дальше и выпустить его в реальную сеть. Кроме того, Dynamips умеет эмулировать как старые Cisco PIX, так и небезызвестную Cisco ASA [3].

- предоставляет два различных стиля просмотра: режим Telnet'а или режим подключения по консоли;
- поддерживает до 200 устройств на одной топологии;
- включает в себя лаборатории, которые поддерживают симуляцию SDM;
- включает в себя не Cisco-устройства, такие как TFTP Server, TACACS генератор пакетов [2].

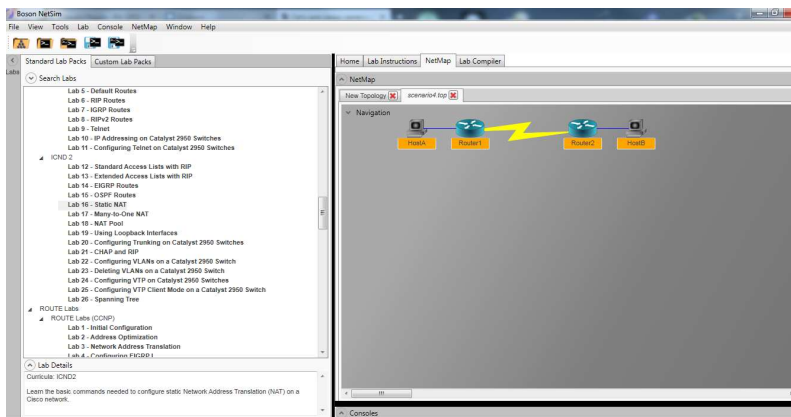


Рис. 3. Boson NetSim

Итог: тем, кому не жалко определенной суммы и при этом не хочется разбираться и создавать свои топологии, а хочется просто попрактиковаться, будет очень кстати.

Хотелось бы немного рассказать о симуляторе оборудования Huawei eNSP.

eNSP. Enterprise Network Simulation Platform симулирует маршрутизаторы уровня Enterprise, коммутаторы и конечное оборудование. По сути, он ближе к Cisco Packet Tracer, имеет понятный графический интерфейс, является именно симулятором. Распространяется совершенно бесплатно – достаточно зарегистрироваться на сайте. Реализует огромное количество функций реального оборудования, по сути, только довольно специфические вещи нельзя реализовать. Доступны MSTP, RRRP, SEP, BFD, VRRP, различные IGP, GRE, BGP, MPLS, L3VPN [4].

Для того чтобы сложилось окончательное мнение о рассмотренных системах имитационного моделирования сетей связи, составим сравнительную таблицу достоинств и недостатков (таблица).

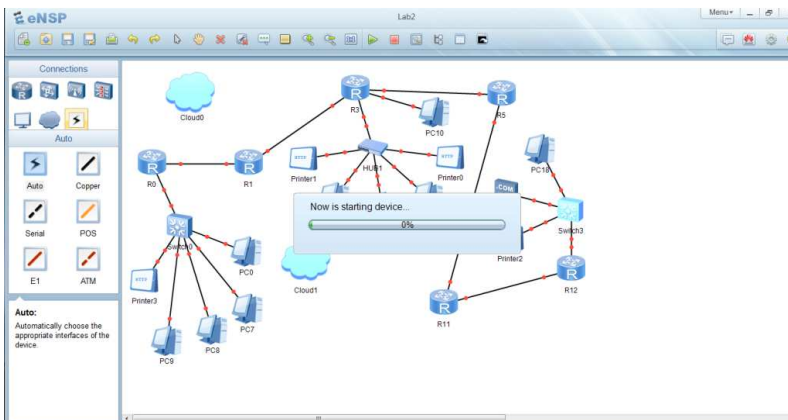


Рис. 4. Симулятор eNSP

Достоинства и недостатки систем имитационного моделирования сетей связи

Симулятор сети	Достоинства	Недостатки
Cisco packet tracer	<ul style="list-style-type: none"> • Дружественный интерфейс (GUI) • Моделирование в режиме реального времени • Большой выбор оборудования • Режим симуляции • Удобство работы с сетевыми сервисами 	<ul style="list-style-type: none"> • Редкие зависания программы
GNS3	<ul style="list-style-type: none"> • Возможность создавать виртуальные машины VirtualBox или VMware Workstation • Выход в реальную сеть • Возможность эмулировать как старые Cisco PIX, так и Cisco ASA 	<ul style="list-style-type: none"> • Количество платформ строго ограничено • Невозможно полноценно использовать коммутаторы Catalyst
Boson NetSim	<ul style="list-style-type: none"> • Симулирует сетевой трафик с помощью технологии виртуальных пакетов • Два стиля просмотра: режим Telnet'а или режим подключения по консоли, • Поддерживает до 200 устройств на одной топологии • Включает в себя не - Cisco устройства, такие как TFTP Server, TACACS + и генератор пакетов 	<ul style="list-style-type: none"> • Редкие зависания программы
eNSP	<ul style="list-style-type: none"> • Имеет понятный графический интерфейс • Распространяется совершенно бесплатно • Реализует огромное количество функций реального оборудования 	<ul style="list-style-type: none"> • Частые зависания программы

Исходя из данных таблицы, делаем вывод, что если вы хотите просто попрактиковаться, то для вас наиболее подойдет система моделирования Boson NetSim, которая представляет собой сборник лабораторных работ. Если вы хотите построить относительно небольшую модель сети, то вам совсем не нужно тратить деньги на приобретение дорогостоящих систем для моделирования, и вы можете выбрать ту систему из доступных для всех пользователей, которая наиболее подходит вам, исходя из задач, поставленных перед вами. Ну а если вы продвинутый пользователь и вам недостаточно того функционала, который представлен в доступных системах моделирования, то вам стоит задуматься о приобретении отнюдь не дешевой системы имитационного моделирования.

Библиографический список

1. Одома Уэнделла. Официальное руководство Cisco по подготовке к сертификационным экзаменам CCENT/CCNA ICND1 640-822. Сер. Cisco Press. – 3-е изд. – М.: Вильямс, 2013.
2. Boson NetSim Technology: руководство пользователя. – М., 2013. – 65 с.
3. GNS3. The software that empowers network professionals [Электронный ресурс]. – URL: <http://www.gns3.com>.
4. Huawei Россия [Электронный ресурс]. – URL: <http://huawei-russia.ru>.

РЕШЕНИЕ ПРОБЛЕМЫ МАСШТАБИРУЕМОСТИ ПРОГРАММНОЙ МОДЕЛИ ПРИЛОЖЕНИЯ УЗЛА РАСПРЕДЕЛЁННОЙ ИНФОРМАЦИОННО- УПРАВЛЯЮЩЕЙ СИСТЕМЫ

Магистранты М.М. Кон, В.В. Чмыков
Пермский национальный исследовательский
политехнический университет

В работе решается проблема масштабируемости программной имитационной модели приложения узла распределённой информационно-управляющей системы. Предложенный подход обеспечивает возможность применения модели с минимальными трудовременными затратами на её перестроение для различных исходных данных.

Ключевые слова: вероятностно-временные характеристики, гибкость, масштабируемость, приложение узла, программная имитационная модель, промышленная сеть, распределённая информационно-управляющая система, LonWorks.

SOLUTION PROBLEMS SCALABILITY SIMULATION MODEL APPLICATION-LEVEL NODE FIELDBUS SYSTEM

Master Students M.M. Kon, V.V. Chmikov
Perm National Research Polytechnic University

In this paper solved the problem of scalability of the software simulation model application fieldbus node. The proposed approach enables the use of models with minimal time spent on rebuild it for a variety of input data.

Keywords: probability-time characteristics, flexibility, scalability, application node, software simulation model, industrial network (fieldbus), LonWorks.

Введение. Современные распределённые информационно-управляющие системы являются эффективным инструментом автоматизации и представляют собой сложную гетерогенную систему, оперирующую большими объемами различной информации в реальном времени. Одной из важных задач при построении распределённой системы является обеспечение её функционирования с заданными вероятностно-временными характеристиками (ВВХ). Решение данной задачи определяет необходимость использования адекватных средств (моделей и методов) количественной оценки характеристик системы.

Анализ известных работ [1–6] определяет недостаточность проработанности вопросов адекватной оценки характеристик распределённых систем на основе fieldbus-технологий (промышленных сетей), что затрудняет их применение в промышленности. Наряду с задержкой, вносимой сетевой обработкой, особенно на фоне растущих скоростей передачи, необходимыми являются оценка и учёт задержки, вносимой приложением узла промышленной сети в процесс обработки и передачи информации [4]. Ввиду известной высокой размерности и вычислительной сложности решаемой задачи оценки ВВХ промышленной сети целесообразным является применение программного имитационного инструментария. Целью настоящей работы является проработка вопросов, связанных с разработкой программной модели функционирования приложения узла, и интеграция её в предложенную ранее инфраструктуру имитационной модели промышленной сети [7]. В частности, решается задача обеспечения масштабируемости разрабатываемой модели.

В работе выполняются анализ и моделирование приложения узла, характерного для широкого класса промышленных сетей со стеком транспортных протоколов на основе методов случайного множественного доступа CSMA: LonWorks, Modbus TCP, EtherNet/IP (DeviceNet), ProfiNet (Profibus поверх Ethernet), HSE (Foundation Fieldbus), Interbus-TCP/IP, EIB/KNX и др. Приложения промышленных сетевых узлов программируются с использованием стандартов IEC 61850, 61499 и др. Дисциплина обслуживания информационных событий приложением производится по циклической схеме с приоритетами, подробное описание и анализ функционирования которой представлены в [8], а в настоящей работе приводится лишь краткое описание в объёме, достаточном для понимания представляемых моделей. Моделирование приложения выполняется в распространённой системе AnyLogic, характеризующейся наиболее оптимальным сочетанием преимуществ языка программирования (Java) и графических средств разработки и визуализации, позволяющих выполнять адекватную оценку широкого спектра вероятностно-временных характеристик [9].

Краткое описание принципов функционирования приложения узла. Приложением узла решаются задачи циклического контроля (when) технологической информации (измерительной от сенсоров и управляющей от контроллеров) и решения соответствующих задач (task) по обработке информации (рис. 1).

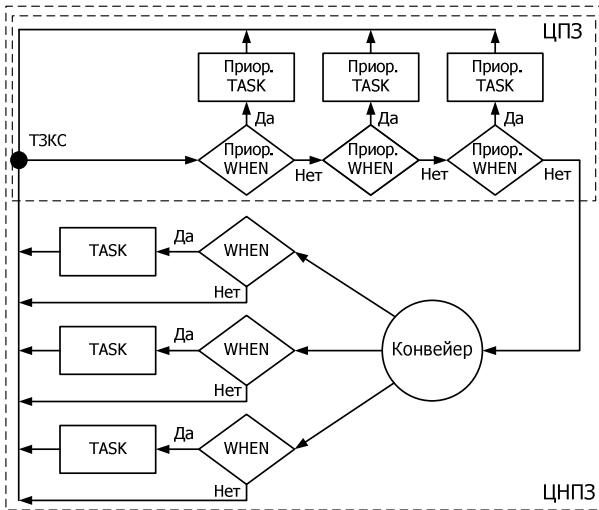


Рис. 1. Циклический алгоритм обслуживания задач уровня приложения

Последовательность обработки прикладных модулей (when-task) приложения осуществляется по круговой дисциплине обслуживания (Round-robin) с двумя уровнями приоритетов. Приложение содержит цикл приоритетных задач (ЦПЗ), в рамках которого происходит последовательное выполнение приоритетных проверок (priority when), при успешном выполнении которых производится решение соответствующих задачи модуля (task), а после – возврат к началу цикла в точку завершения критической секции (ТЗКС, точку входа). Переход к циклу неприоритетных задач (ЦНПЗ) происходит в случае завершения ЦПЗ без выполнения успешных проверок. Цикл обслуживания неприоритетных задач (ЦНПЗ) включает в состав ряд неприоритетных задач узла. За один этап неприоритетного цикла выполняется лишь один модуль (в порядке следования), после чего управление передаётся в точку входа, а после – начинается новый приоритетный цикл.

Разработка модели функционирования приложения узла.

Одним из основных требований, определяющих сложность разработки программной модели промышленной сети в целом и приложения узла в частности, является обеспечение высокой гибкости моделирования. Уровень гибкости должен обеспечить возможность применения модели для анализа вероятностно-временных характеристик сети

при исследовании широкого диапазона варьируемых исходных данных при минимальных трудовременных затратах этапов подготовки и редактирования модели в процессе её эксплуатации. В данной постановке задачи одним из основных показателей гибкости является масштабируемость модели, определяющая сложность изменения (перестроения) модели под необходимые исходные данные.

Проблема обеспечения масштабируемости модели тесно связана со сложностью её программной реализации. Современные системы имитационного моделирования (СИМ) активно внедряют механизмы уменьшения сложности разработки, в общем случае использованием визуального блочного инструментария построения модели. Модель строится с использованием запрограммированных разработчиками СИМ блоков-состояний, переходов между ними, оснащённых параметрами и методами работы с ними. В распоряжение исследователя предоставляется, как правило, определённый спектр программных директив, необходимых для дополнения модели требуемым логическим функционалом. Известны практические недостатки визуального блочного моделирования. Так, масштабирование модели может производиться только путём ручного изменения исследователем структуры модели и связей между её элементами. Указанный недостаток получен в работе при разработке первоначальной имитационной модели приложения узла (рис. 2).

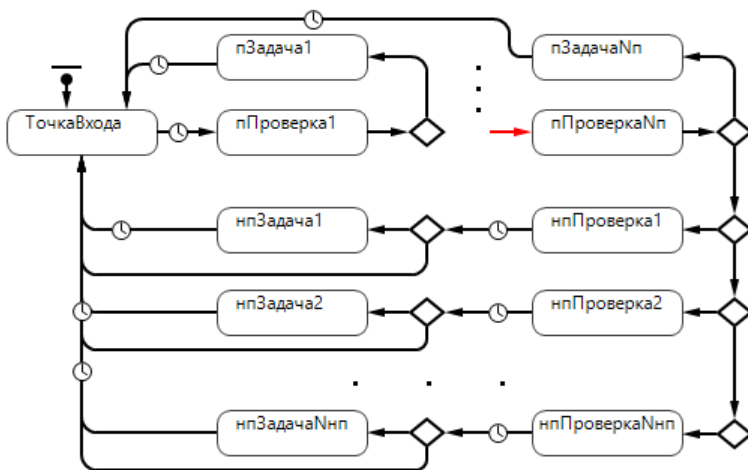


Рис. 2. Имитационная модель № 1 приложения узла

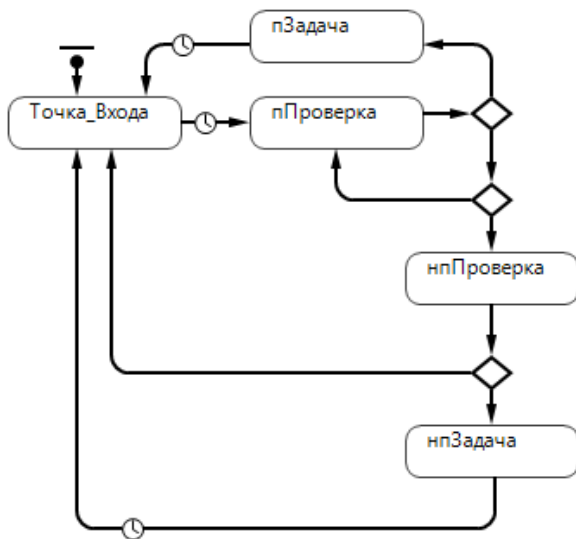
Изменение числа приоритетных и не приоритетных задач приложения узла определяет необходимость ручного удаления/добавления блоков и изменения связей между ними.

Таким образом, данный подход не может быть использован для решаемой авторами задачи оценки и исследования ВВХ, поскольку не может обеспечить гибкость моделирования для различных исходных данных и приведёт к высоким трудовременным затратам перестройки модели приложений каждого узла сети.

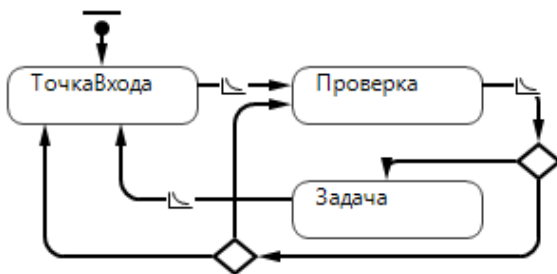
Известным подходом к обеспечению высокого уровня масштабируемости является подход на основе полной программной реализации модели без использования инструментария блочной разработки. Однако такой подход характеризуется крайне высокой сложностью и трудоёмкостью реализации, поскольку помимо реализуемого логического функционала модели, объектов различного класса, методов их взаимодействия необходимо обеспечение системной части моделирования (продвижение модельного времени, синхронизация и организация последовательностей цепей событий и др.). Таким образом, по мнению авторов, эффективным и целесообразным является поиск оптимального, с точки зрения простоты реализации, необходимого уровня масштабируемости модели, подхода к разработке модели. Поиск заключается в определении оптимального сочетания преимуществ средств систем имитационного моделирования, в том числе визуального блочного моделирования, с программными средствами разработки. Процесс поиска в работе выполняется путём последовательного обобщения модели, сокращения однотипных блоков и связей между ними, переводом их представления из визуального в программный формат.

Для повышения масштабируемости модели произведём замену однотипных блоков задач и проверок с одинаковым уровнем приоритета на общие блоки (рис. 1, модель № 2). Предложенная структура не нуждается в блочном масштабировании, однако необходима разработка программной процедуры определения текущих номеров обрабатываемых модулей приложения. Недостатками модели являются её не универсальность по отношению к возможному числу уровней приоритетов (разработана для 2 уровней), а также необходимость раздельного получения, хранения и использования исходных данных об индивидуальных параметрах модулей приложения узла.

Для устранения указанных недостатков предлагается выполнить обобщение блочной структуры модели. Путём замены блоков проверки и задач с разными уровнями приоритетов получена обобщённая блочная модель № 3 (рис. 3) приложения узла промышленной сети. Предложенная модель, по мнению авторов, представлена оптимальным сочетанием преимуществ визуального блочного моделирования СИМ и программной реализации логики функционирования (смены номеров обслуживаемых модулей).



a



б

Рис. 3. Модели №2 (а) и №3 (б) приложения узла сети

Блочная структура обеспечивает каркас модели, последовательность прохождения элементов приложения, моделирование с течением времени. Программная реализация служит для определения текущего номера обслуживаемого модуля (в зависимости от предыстории обслуживания и реализуемой логики функционирования), что позволяет в автоматическом, а не в ручном, как в модели №1, выбрать необходимые индивидуальные исходные данные по текущему обслуживаемому модулю приложения (проверке и задаче). Таким образом, решена задача обеспечения требуемого уровня масштабируемости и соответственно гибкости применения модели, без необходимости её ручного перестроения на этапе эксплуатации.

В работе решена проблема обеспечения требуемого уровня масштабирования модели с целью обеспечения возможности её эксплуатации с минимальными трудовременными затратами, связанными с перестроением модели под исследуемые исходные данные. Установлено, что оптимальный уровень масштабирования обеспечивается определённым сочетанием преимуществ средств визуального блочного программирования и программной реализации.

Библиографический список

1. Даденков С.А., Кон Е.Л. Подход к построению аналитической модели информационно-управляющей сети LonWorks на основе нейрочипов // Нейрокомпьютеры: разработка, применение. – 2013. – № 11. – С. 64–69.

2. Даденков С.А., Кон Е.Л. Исследование производительности алгоритма доступа к среде predictive p-persistent CSMA протокола // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. – 2012. – № 6. – С. 217–230.

3. Miśkiewicz M. Latency Characteristics of Event-Driven Task Scheduler Embedded in Neuron Chip // IJCSNS International Journal of Computer Science and Network Security. – 2007. – Vol. 7, № 12. – P. 132–149.

4. Даденков С.А., Кон Е.Л. Метод расчета числа узлов-соперников за канал в сети со случайным множественным доступом, построенной на основе нейрочипов // Нейрокомпьютеры: разработка, применение. – 2015. – № 10. – С. 26–32.

5. Даденков С.А., Кон Е.Л., Южаков А.А. Алгоритм проектирования промышленной информационно-управляющей сети требуемой производительности (на примере технологии LonWorks) // Вестник Поволж. гос. технолог. ун-та. Сер. Радиотехнические и инфокоммуникационные системы. – 2014. – № 5(24). – С. 35–43.

6. Даденков С.А., Кон Е.Л. Оценка степени влияния некоторых факторов на производительность LonWorks-сети // Образовательные ресурсы и технологии. – 2014. – № 2(5). – С. 72–76.

7. Даденков С.А., Кон Е.Л., Чмыков В.В. Имитационная модель промышленной сети (на примере технологии LonWorks) // Проблемы техники и технологии телекоммуникаций ПТиТТ-2014: материалы XV Междунар. науч.-техн. конф. – Казань: Изд-во Казан. гос. техн. ун-та, 2014. – Т. 1. – С. 82–84.

8. Дитрих Д., Лой Д., Швайнциер Г.Ю. LON-технология, построение распределенных приложений: пер. с нем. / под ред. О.Б. Низамудинова. – Пермь: Звезда, 1999. – 242 с.

9. Даденков С.А., Кон Е.Л. Анализ моделей и методов агентного и дискретно-событийного имитационного моделирования // Известия СПбГЭТУ «ЛЭТИ». – 2015. – № 5. – С. 35–41.

О МЕТОДЕ ВСТРАИВАНИЯ ЦИФРОВЫХ ВОДЯНЫХ ЗНАКОВ В ВИДЕОПОТОКИ ДЛЯ ЗАЩИТЫ ОТ ИНФОРМАЦИОННЫХ АТАК

Студент С.П. Шипицин, канд. техн. наук, доцент А.С. Шабуров
Пермский национальный исследовательский
политехнический университет

В работе рассматривается метод встраивания цифровых водяных знаков в подвергнутые сжатию видеопотоки с использованием алгоритмов, основанных на блочном кодировании. Предлагаются некоторые варианты его применения. Описываются детали реализации данного метода, в частности, способы повышения устойчивости встроенного сообщения к характерным информационным атакам.

Ключевые слова: цифровой водяной знак, видеопоток, стегосистема, блочное кодирование, робастность, суперблок, субблок.

ABOUT THE METHOD OF EMBEDDING A DIGITAL WATERMARK IN THE VIDEO STREAM FOR PROTECTION FROM INFORMATION ATTACKS

Student S.P. Shipitsin,
Ph.D. in Technical Sciences, Associate Professor A.S. Shaburov
Perm National Research Polytechnic University

The paper concentrates on a method of embedding a digital watermark in a video stream compressed using algorithms based on block coding. Offers some options for its use. Describes the implementation details of this method, in particular, ways to improve the stability of the built-in message to the characteristic information attacks.

Keywords: digital watermark, video stream, stegosystem, block coding, robustness, coding tree unit, coding unit.

Использование цифровых водяных знаков (ЦВЗ), например, как средства обеспечения авторских и смежных прав в сфере предоставления цифрового медиаконтента, на сегодняшний день является стандартом де-факто. Широкое распространение получили системы встраивания ЦВЗ в видеопотоки. Следует, однако, отметить, что самые современные из этих технологий разработаны к середине прошлого десятилетия и уже не только морально устарели, но и не обеспечивают должного уровня защиты от информационных атак. В данной работе рассматривается метод встраивания ЦВЗ, в перспективе

позволяющий установить личность злоумышленника, в случае утечки информации при интеграции с системами предоставления проприетарного видеоконтента. Кроме того, предлагаются меры по повышению его устойчивости к наиболее характерным для стегосистем информационным атакам.

В работе [1] был предложен метод встраивания ЦВЗ в видеопотоки, сжатые с использованием древовидных структур кодирования, на основе стандарта HEVC. Кадр сжимаемого таким образом видеопотока разделяется на блоки 64×64 пикселя, затем каждый блок, содержащий высокодетализированный участок изображения, разбивается на блоки меньшего размера, вплоть до 4×4 (размеры приведены для кодека VP9). Применение этой технологии стало отличительной особенностью стандартов сжатия нового поколения (рис. 1) [2].

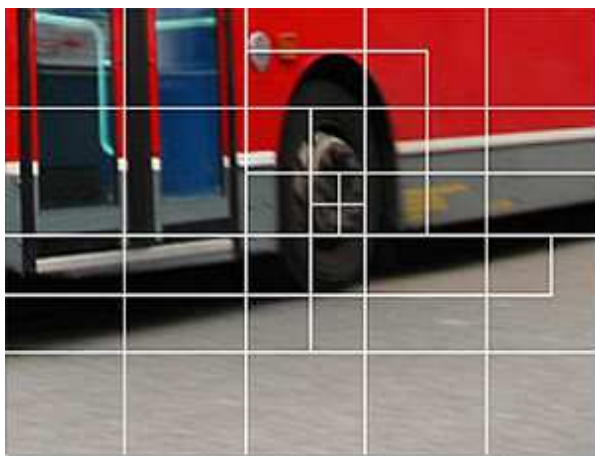


Рис. 1. Условное разбиение кадра на блоки в HEVC

Поскольку в русскоязычной литературе отсутствует общепринятый аналог терминам «coding tree unit» и «coding unit», без потери смысла будем называть самые крупные блоки суперблоками, блоки меньшего размера – субблоками. Суть метода состоит во встраивании монохромного ЦВЗ в компоненту яркости исходного изображения. Субблоки определённого размера играют роль своего рода пикселей и используются следующим образом: после наложения изображения – ЦВЗ в блоках, соответствующих контуру изображения, производится инкремент компоненты яркости, а в соседних блоках – её декремент

(рис. 2). В связи с наличием суперблоков полное формирование ЦВЗ в одном кадре маловероятно, поэтому предлагается применять сложение функций яркости нескольких кадров друг с другом (рис. 3).

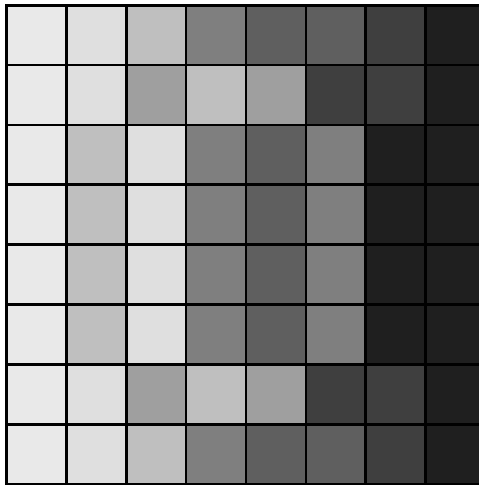


Рис. 2. Пример встраивания (контрастность увеличена)

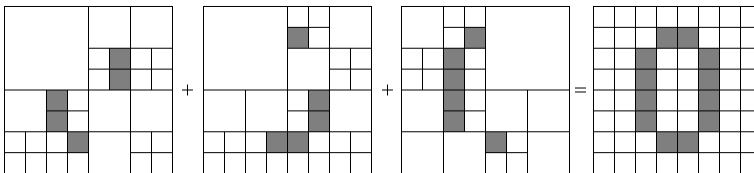


Рис. 3. Окончательное формирование ЦВЗ

Предпочтительный размер блоков для встраивания выбирается исходя из разрешения изображения-ЦВЗ и необходимости противодействия атакам на стегосистему. Так, в случае использования суперблоков в видеостандарте HD (1280×720) разрешение встраиваемого изображения должно составлять: $\frac{1280}{64} \times \frac{720}{64} \approx 20 \times 11$, т.е. всего 220 бит. С другой стороны, при использовании наименьших субблоков получаем большой информационный объём: $\frac{1280}{4} \times \frac{720}{4} \approx 320 \times 180 = 56,25$ Кбайт, однако дальнейшее сжатие видео приведёт к разрушению ЦВЗ.

Очевидно, что описанный выше метод применим к любым изображениям, сжатым с помощью алгоритмов блочного кодирования. В частности, стандарт сжатия предыдущего поколения AVC предполагает разбиение изображения на блоки одинакового размера – макроблоки (рис. 4), которые с некоторыми допущениями можно использовать аналогичным образом.



Рис. 4. Разбиение того же кадра на макроблоки в AVC

Вид встраиваемого сообщения может значительно изменяться в зависимости от конечной цели правообладателя. Если ставится задача простого обнаружения авторского контента для блокировки его воспроизведения или копирования на конечных устройствах, более эффективным является внедрение ЦВЗ в виде, например, логотипа компании-владельца на всю величину кадра, что серьёзно усложняет разрушение ЦВЗ без неприемлемой потери качества самого контента. Если же необходимо обнаружить и привлечь к ответственности злоумышленника постфактум, после утечки авторского контента, тогда, очевидно, требуется вносить во встраиваемое сообщение информацию однозначно идентифицирующего конкретного пользователя, т.е. «отпечатки пальцев» [3]. В данной работе предполагается использование второго варианта встраивания, поскольку это предполагает личную ответственность злоумышленника, что наиболее перспективно с точки зрения предотвращения неконтролируемого распространения авторского контента.

Характер и объём встраиваемой в ЦВЗ информации во многом определяют его робастность (устойчивость) к различным типам

информационных атак. Так, коалиционные атаки (осуществляемые посредством сговора нескольких злоумышленников) абсолютно бесполезны, если все распространяемые владельцем контента копии содержат одно-единственное сообщение, однако крайне эффективны при встраивании уникальных идентификаторов [3]. При этом злоумышленники могут как создавать ложный ЦВЗ, так и исказить уже существующий. Доказательство возможности организации защиты стегосистемы от подобной атаки приведено в [4]. Далее будем предполагать, что ЦВЗ представляет собой изображение-текст, содержащее уникальный идентификатор-ссылку на конкретную строку в базе данных пользователей и дату-время предоставления доступа к видеофайлу. Соответственно, в простейшем случае для противодействия коалиционным и некоторым другим атакам можем применять помехоустойчивое кодирование.

Важную роль играет глубина квантования, с которой закодировано видео, так как не все уровни компонентов изображения используются по прямому назначению. Так, при квантовании с разрядностью 8 бит из 256 доступных уровней только 220 используются для передачи сигнала яркости (диапазон 16–235), а остальные – для сигналов синхронизации. При 10-битном кодировании используются 877 уровней [5]. В связи с тем, что зрительный аппарат человека наиболее чувствителен к изменениям компонента яркости [6], куда и производится встраивание, следует искать компромисс между незаметностью и устойчивостью ЦВЗ. В качестве дополнительной меры снижения заметности можно встраивать ЦВЗ только в высокодетализированных участках изображения (высокая детализация обуславливает применение субблоков меньшего размера и высокую контрастность), однако в этом случае для встраивания потребуется более глубокое декодирование исходного видео, что снизит быстродействие системы.

Возвращаясь к выбору размера субблоков, применяемых для встраивания ЦВЗ, прежде всего необходимо обеспечить устойчивость последнего к атакам перекодирования с низким разрешением и аффинным (геометрическим) преобразованием. Эмпирически примем разрешение, при котором качество видео достигает критически низкого уровня, равного 144 p. Кроме того, после перекодирования блоки должны выродиться не в единичные пиксели, а в достаточно устойчивые структуры, например 2×2 . Допустим, имеем исходное ви-

део с разрешением 720 p, тогда получаем $\frac{720 \times 2}{144} = 10$. Округляя до ближайшей большей степени двойки, получаем 16×16 – эффективный для встраивания размер субблока.

Таким образом, развивающаяся законодательная база в сфере защиты авторских и смежных прав сталкивается с проблемой недостаточно эффективных методов сбора доказательной базы при совершении правонарушений со стороны злоумышленников. Особенно сильно это сказывается на владельцах цифрового медиаконтента, в том числе видео. Рассмотренный в данной работе метод встраивания цифровых водяных знаков позволяет однозначно установить личность злоумышленника, что обуславливает перспективность метода для дальнейших исследований.

Библиографический список

1. Шипицин С.П. Встраивание цифровых водяных знаков в видеопотоки, сжатые с использованием древовидных структур кодирования // Научное сообщество студентов XXI столетия. Технические науки: материалы XXXII студ. междунар. науч.-практ. конф.; Новосибирск, 26 мая 2015 г. / АНС «СибАК». – Новосибирск: СибАК, 2015. – № 5(31).
2. Richardson I.E. An introduction to High Efficiency Video Coding (FULL) [Electronic resource]. – Electronic data. – Aberdeen: Vcodex Ltd, 2013. – URL: <https://www.vcodex.com/press.asp>. – Title screen.
3. Грибунин В.Г., Оков И.Н., Туринцев И.В. Цифровая стеганография. – М.: СОЛОН-ПРЕСС, 2009. – 272 с.
4. Соловьёв Т.М., Черняк Р.И. Множества идентификационных номеров, устойчивые к атаке сговором // Прикладная дискретная математика. – 2012. – С. 60–68.
5. ITU-T. Recommendation H.265. – Geneva, 2014. – 518 p.
6. Marc R.E., H.G. Sperling. Chromatic Organization of Primate Cones // Science. New Series. – Apr. 22, 1977. – Vol. 196, № 4288. – P. 454–456.

ПРЕИМУЩЕСТВА И НЕДОСТАТКИ СИСТЕМЫ ЗАЩИТЫ АВТОРСКОГО ПРАВА YOUTUBE «CONTENT ID» ПО ВИДЕОПОТОКУ

Студент М.И. Ямаев

Пермский национальный исследовательский
политехнический университет

В статье затронута часть принципов работы системы защиты авторского права «Content ID» видеохостинга YouTube. Также рассмотрены положительные и отрицательные стороны этой системы. Раскрыты основные её уязвимости.

Ключевые слова: авторское право, контент, уязвимость, видео, защита информации.

ADVANTAGES AND DISADVANTAGES OF COPYRIGHT PROTECTION SYSTEM YOUTUBE CONTENT ID FOR VIDEO STREAM

Student M.I. Yamaev

Perm National Research Polytechnic University

The article touched upon principles of the system of the Content ID copyright protection of video hosting YouTube. Discussed the positive and negative sides of this system. Declared the main vulnerabilities of system.

Keywords: copyright, content, vulnerability, video, information protection.

Защита прав интеллектуальной собственности является особенно актуальной в современном информационном обществе. Некоторым людям результаты их интеллектуальной деятельности приносят денежные доходы, а также иные выгоды. Злоумышленники могут попытаться скопировать эти результаты и выдать за свои для получения таких же или других выгод. Для этого и существует институт защиты интеллектуальной собственности, т.е. авторское право. Авторское право распространяется на объекты в письменной или устной форме, на изображения, звукозаписи или видеозаписи, а также объёмно-пространственные объекты.

Непрерывное развитие видеоиндустрии вызвано мощными коммерческими и технологическими силами. YouTube – лидер среди видеохостингов, который принадлежит компании Google и предоставляет пользователям услуги хранения, доставки и показа видео. Сейчас дан-

ный видеохостинг очень популярен среди зрителей. Аудитория YouTube – больше одного миллиарда человек. Это почти треть всех пользователей Интернета. Миллиарды просмотров и сотни миллионов часов воспроизведения. Правообладатели загруженных на видеохостинг видеозаписей становятся партнёрами YouTube или иных внешних компаний, встраивая в своё видео их рекламу. За счёт просмотров самой видеозаписи правообладатели получают доходы. Злоумышленники могут скопировать видеозапись и перезагрузить на видеохостинг, но уже для своей выгоды. В итоге правообладатель теряет часть зрительской аудитории и соответственно часть дохода. Поэтому была разработана система защиты авторского права «Content ID» для авторов, которые обладают исключительными правами на свой контент и хотят следить за тем, как он используется на YouTube.

Работу «Content ID» можно выразить в четыре этапа:

1. Настройка аккаунта владельца контента, т.е. открытие возможности использования «Content ID». Перед подачей заявки на подключение системы заявители должны предоставить свидетельства своих исключительных авторских прав на контент.

2. Загрузка контента в систему. Для каждого загруженного элемента создаётся объект в системе управления контентом и цифровой отпечаток в системе «Content ID». Если был размещён визуальный контент, то для него будет создано видео. *Цифровой отпечаток* – это копия аудиофайла и видеофайла объекта, которая используется системой для поиска совпадений.

3. Сканирование видеофайла и поиск совпадений. Когда объекты и цифровые отпечатки будут созданы, «Content ID» начнет сравнивать с ними все ролики, которые были загружены пользователями, но ещё не опубликованы на видеохостинге. Если в каком-то из них будет обнаружен контент, защищенный авторским правом, система заявит на ролик права от имени владельца и применит выбранную им политику к размещённому видео в случае обнаружения совпадений с цифровым отпечатком. Всего существуют 3 политики: автор может «монетизировать» видео, прикрепив к нему рекламу, заблокировать его к просмотру или же просто оставить для наблюдения за его статистикой. Кроме того, система ищет совпадения не только в новых, ещё не загруженных файлах, но и в уже размещённых. Но данная процедура может длиться несколько месяцев.

4. Управление контентом и просмотр статистики. После того как на видео другого пользователя будут заявлены права собственника,

ему, возможно, потребуется рассмотреть заявку, разрешить конфликт прав или выполнить какое-то другое действие.

Существующая на данный момент система «Content ID» уже обеспечивает на хорошем уровне защиту от копирования и дублирования контента на видеохостинге. Правообладателям теперь намного реже или вовсе не приходится напрямую создавать спор о принадлежности контента. Более того, доходы некоторых пользователей только возросли от того, что к видеофайлу или аудиофайлу другого пользователя применена политика «монетизации». Но данная система имеет ряд уязвимостей, не предусмотренных решений, которыми пользуются злоумышленники. С помощью некоторых операций можно обойти систему сравнения нового видеофайла с цифровым отпечатком. К сожалению, на YouTube таких пользователей очень много. Эти пользователи забирают себе часть зрительской аудитории, а значит, и часть доходов или иных выгод.

Попытаемся найти или протестировать ряд из этих проблем. С помощью бинарного перебора попыток загрузить файл, защищённый авторским правом, определяем наименьшую длину скопированного видеофайла для размещения на видеохостинге. Было использовано несколько видеофайлов для более точных результатов, так как в некоторых случаях система «Content ID» не обнаруживала видео длиной до 38 с, а в иных обнаруживала видео длиной от 35 с. Отсюда первая проблема: абсолютно любой пользователь может загрузить видеофайл, наделённый авторскими правами, укоротив его до 30 с и избежать политик. При добавлении поверх видео (рис. 1) статичной текстуры или картинки со степенью прозрачности 30 % и более система так же не обнаруживала видеофайл (рис. 2), причём любой длины, вплоть до оригинальной.



Рис. 1. Исходный видеофайл



Рис. 2. Видеофайл после наложения текстуры

Следовательно, система уязвима к сильным искажениям видео. Можно также наложить рамку на видео, уменьшив видимую область видео (рис. 3, *а*), или же изменить его разрешение (рис. 3, *б*), после чего система так же не найдёт совпадений с цифровым отпечатком.



а



б

Рис. 3. Добавление рамки к видео

На данный момент система далека от своего идеала. В видеохостинге очень много дублирующего контента, некоторый может быть даже популярнее исходного. Другие пользователи могут почти беспрепятственно «отбирать» часть зрительской аудитории на себя за счёт перезакачивания уже созданного, но незначительно отредактированного контента и получать доходы выставляя на них «монетизацию».

Но в итоге система «Content ID» все же обеспечивает защиту авторского права на приемлемом уровне, хотя и не достаточном. Она будет развиваться со временем, сможет более точно определять скопированный контент, и большая часть уязвимостей будет устранена.

Библиографический список

1. Тропченко А.Ю., Тропченко А.А. Методы сжатия изображений, аудиосигналов и видео: учеб. пособие. – СПб: Изд-во СПбГУ ИТМО, 2009. – 108 с.
2. Статистика YouTube [Электронный ресурс]. – URL: <http://www.youtube.com/yt/press/ru/statistics.html> (дата обращения: 26.11.2015).
3. Справочный центр YouTube [Электронный ресурс]. – URL: <https://support.google.com/youtube> (дата обращения: 27.11.2015).

АЛТЕРНАТИВНЫЙ АЛГОРИТМ ПРОТОКОЛА «ПОДБРАСЫВАНИЯ МОНЕТЫ ПО ТЕЛЕФОНУ»

Студент М.И. Ямаев, канд. физ.-мат. наук, доцент Е.Л. Кротова
Пермский национальный исследовательский
политехнический университет

В статье разбираются существующие алгоритмы «подбрасывания монеты по телефону» и предлагаются альтернативные версии не использующие в качестве основы вычислительную сложность дискретного логарифма и факторизацию большого простого числа.

Ключевые слова: протокол, шифрование, жеребьёвка, случайная последовательность, защита информации.

ALTERNATIVE ALGORITHM OF PROTOCOL COIN FLIPPING BY TELEPHONE

Student M.I. Yamaev, Ph.D. in Physics and Mathematical Sciences,
Associate Professor E.L. Krotova
Perm National Research Polytechnic University

The paper explains the existing algorithms "coin flip phone" and offered an alternative versions that is not use as a basis the computational complexity of the discrete logarithm and factoring large prime numbers.

Keywords: protocol, encryption, sortition, random sequence, information protection.

Анна и Барт хотят сделать случайный выбор. К примеру, подбросить монету, но при этом они находятся друг от друга удалённо, в разных городах. Здесь у того, кто подбрасывает монету, есть возможность после броска монеты солгать другому, а другой может ему не поверить. Поэтому нужно было создать алгоритм, выдающий независимый от пользователей случайный результат. Это возможно сделать, прибегая к криптографическим методам. В 1981 г. Мануэль Блум опубликовал статью о протоколе «подбрасывания монеты по телефону» (Coin Flipping By Telephone), причём в заголовке своей работы он назвал это методом решения «нерешаемых задач». Проблема была решена с помощью добавления в процесс третьего лица, на которое Анна и Барт возлагали доверие. Протокол позволял сторонам генерировать случайное число, состоящее из m бит, и включал он 7 шагов:

1. Барт генерирует большое целое число Блюма $N = PQ$, где P и Q – это два больших простых числа, удовлетворяющих условию $P \equiv Q \equiv 3 \pmod{4}$ (такие числа обладают полезными криптографическими свойствами). После отправляет N Анне.

2. Анна генерирует m случайных больших целых чисел: x_1, x_2, \dots, x_m . В данном случае результатами жеребьёвки считаются числа $\left(\frac{xi}{N}\right)$, $i = 1, 2, \dots, m$. Далее вычисляет числа $y_i = x_i^2 \pmod{N}$ и отправляет их Барту.

3. Барт генерирует случайные знаки b_1, b_2, \dots, b_m , пытаясь угадать знаки чисел $\left(\frac{xi}{N}\right)$. Далее вычисляет числа $y_i = x_i^2$, после чего отправляет их Барту.

4. Анна сообщает Барту результаты угадывания, отправляя ему числа x_1, x_2, \dots, x_m .

5. Барт проверяет выполнение условия $y_i = x_i^2 \pmod{N}$ и открывает Анне числа P и Q .

6. Анна проверяет условия $P \equiv Q \equiv 3 \pmod{4}$ и проверяет, являются ли числа P и Q простыми.

7. Анна и Барт формируют случайную последовательность битов r_i где $i = 1, 2, \dots, m$, так, что $r_i = 1$, если $\left(\frac{xi}{N}\right) = b_i$ (Барт угадал знак), и $r_i = 0$, если $\left(\frac{xi}{N}\right) \neq b_i$ (Барт не угадал знак).

Получается алгоритм, при котором Анна и Барт могут создать неслучайные последовательности, но при этом они не знают последовательности друг друга, пока не сформируют общую конечную, которую можно считать случайной.

В конечном счёте третьим лицом можно назвать функцию $y_i = x_i^2 \pmod{N}$. На сегодняшний день данная функция является односторонней и основана на сложности задачи восстановления дискретного логарифма, которую пока нельзя решить за разумное время.

Точно так же, пока не решена задача факторизации числа, $N = PQ$, при соответствующих больших простых числах P и Q . Модифицированная версия протокола уже в 5 шагов:

1. Анна выбирает случайное большое целое число x , вычисляет $y = g^x \pmod{p}$ и посылает y Барту.

2. Барт генерирует случайный бит b , случайное большое целое число k , вычисляет $r = y^b g^k \pmod{p}$ и посылает r Анне.

3. Анна генерирует случайный бит c и посылает его Барту.

4. Барт посылает Анне b и k .

5. Анна проверяет, выполняется ли сравнение $r = y^b g^k \bmod p$. Если да, то результатом выполнения протокола будет бит $d = b \oplus c$.

Здесь уже более наглядно отображено то, что для работы протокола достаточно лишь временного сокрытия жеребьевок и открытия их уже после получения результата.

Но над задачами дискретного логарифмирования и факторизации числа, состоящего из двух перемноженных простых чисел, работают огромные математические институты по всему миру, так как на данной задаче основан алгоритм RSA, который на данный момент считается настолько надёжным, что ему доверяют большинство существующих вопросов информационной безопасности. Не исключено, что при существующем росте возможностей вычислительной техники такие алгоритмы станут уязвимыми. Рано или поздно будет найдено простое решение.

Поэтому в этой статье предлагается альтернативный алгоритм, не использующий вычислительную сложность дискретного логарифмирования и факторизации больших целых чисел. Вместо них можно использовать существующие и зарекомендовавшие себя другие криптографические протоколы. К примеру, блочные шифры, работа которых основана на Сети Фейстеля, такие как DES, TripleDES, Blowfish и др. Blowfish был разработан в 1993 г. Брюсом Шнайером, он запатентован и свободно распространяется. Ключ в Blowfish может варьироваться по длине от 32 до 448 бит, а блоки шифрования имеют длину в 64 бит. Если данные, которые необходимо зашифровать, меньше или не кратны 64 битам, то данные автоматически расширяются до нужной длины, а при расшифровке сжимаются. Поэтому в качестве функции, которая будет временно скрывать жеребьевки, будем использовать алгоритм Blowfish. Алгоритм, генерирующий случайную последовательность из m бит, так же будет состоять из 5 шагов:

1. Анна формирует последовательность битов a_1, a_2, \dots, a_m , осуществляет выбор общего кодового слова S , после чего они шифруются алгоритмом Blowfish с ключом k_a . Далее Анна передаёт зашифрованную последовательность, кодовое слово в зашифрованном и открытом виде. (Кодовое слово необходимо для предотвращения подмены ключа для получения иного результата одной из сторон).

2. Барт также формирует уже свою последовательность битов b_1, b_2, \dots, b_m , шифрует её алгоритмом Blowfish с ключом k_b и отправляет её в зашифрованном виде Анне.

3. Анна после получения последовательности от Барта отправляет свой ключ k_a Барту.

4. Барт осуществляет проверку на соответствие ключа и последовательности, а также на соответствие ключа к кодовому слову (шифрует кодовое слово полученным ключом и сравнивает его с полученным ранее зашифрованным). После чего отправляет свой ключ k_b Анне.

5. Анна также проверяет соответствие ключа последовательности и кодовому слову. Если всё удовлетворяет условиям, то формируется конечная случайная последовательность c_1, c_2, \dots, c_m , где $c_1 = 1$, если $a_1 = b_1$ и $c_1 = 0$ в ином случае.

В качестве функции сокрытия жеребьёвки можно использовать любой другой алгоритм, который будет обладать достаточной криптостойкостью, при этом не меняя общего принципа. Blowfish на сегодня является актуальным алгоритмом. Мануэль Блум в своей работе указал, что протокол «подбрасывания монеты по телефону» уже применяется для заказной почты, покера, обмена секретами и в других задачах, требующих создания случайного выбора.

Библиографический список

1. Тригуб С.Н. Современная криптография: теория и практика. – М.: Вильямс, 2005. – 768 с.

2. Введение в криптографию / В.В. Яценко [и др.]. – М.: Изд-во МЦНМО, 2000. – 288 с.

3. Coin Flipping By Telephone / published by Manuel Blum. – University of California at Berkeley, 1981.

О ТРЕБОВАНИЯХ К РАЗРАБОТКЕ МОДЕЛИ КОМПЬЮТЕРНЫХ АТАК НА ТЕЛЕКОММУНИКАЦИОННЫЕ СИСТЕМЫ

Студентка А.А. Миронова, канд. техн. наук, доцент А.С. Шабуров
Пермский национальный исследовательский
политехнический университет

В статье рассмотрена проблема роста компьютерных атак на автоматизированные и телекоммуникационные системы. Приведена статистика роста кибератак. Предлагается минимальный набор требований к модели компьютерной атаки в целях их эффективного распознавания. Приводится схема реализации этапов жизненного цикла типовой информационной атаки на телекоммуникационную систему. Перечисляются основные параметры информационной атаки, необходимые для ее моделирования

Ключевые слова: телекоммуникационная система, кибератака, защита информации, теория распознавания образов, распределенный отказ в обслуживании.

ABOUT REQUIREMENTS FOR THE DEVELOPMENT MODEL OF COMPUTER ATTACKS ON TELECOMMUNICATION SYSTEMS

Student A.A. Mironova,
Ph.D. in Technical Sciences, Associate Professor A.S. Shaburov
Perm National Research Polytechnic University

This article deals with the problem of the increase of computer attacks on the automatized and telecommunicational systems. The increasing cyber-attack statistics has been stated. The minimal set of demands for the model of computer attack to define them effectively has been suggested. The realisation plan of life cycle stages of a typical information attack on a telecommunicational system has been given. The basic characteristics of an information attack necessary for its modelling have been specified.

Keywords: telecommunicational system, cyber-attack, information defence, the theory of pattern recognition, distributed Denial of Service.

Развитие и повсеместное внедрение автоматизированных и телекоммуникационных систем, совершенствование технологии информационного взаимодействия на различных уровнях, в свою очередь, обуславливает ряд современных проблем. Одной из таких проблем, безусловно, является проблема обеспечения информационной безопасности.

Информационная безопасность (ИБ) телекоммуникационных систем (ТС) подвержена угрозам широкого спектра: от вирусного

заражения, с которым можно справиться локально, до нормативно-правовых противоречий, требующих работы, как законодательных органов власти, так и вмешательства правоохранительных структур. Влияние на основные критерии безопасности, которые выражены конфиденциальностью, целостностью и доступностью информационных ресурсов, может происходить как по причине действий злоумышленников, так и являться следствием ошибочных действий пользователей информационных систем. Множество угроз для ТС можно условно классифицировать по различным основаниям: на преднамеренные, или случайные, с использованием аппаратно-программных средств (закладочных устройств), по причине физических нарушений, радиоэлектронные помехи, помехи организационного характера и т.д.

На сегодняшний день наиболее актуальной угрозой для безопасности ТС являются информационные, или кибератаки, на системы обработки информации или управления. Подобное воздействие может прекратить работу ТС или прервать обеспечение ее деятельности, вызвать события, которые могут привести к блокированию жизненно важных операций в различных системах. ИБ ТС нередко можно приравнять к компьютерной безопасности. Наиболее критичными могут быть атаки, которым подвергаются данные, хранящиеся и передающиеся с помощью глобальных сетей и систем. Например, взлом электронных почтовых ящиков стал одним из наиболее распространенных видов нежелательного доступа к конфиденциальной информации. При этом использование злоумышленниками вредоносных программ не только способствует хищению данных, но и серьезно ограничивает скорость их передачи, а также может спровоцировать блокирование работы с информацией во всей ТС.

На сегодняшний день заинтересованность в решении проблем кибербезопасности ТС проявляют как государственные, так и коммерческие структуры. По сведениям специальных служб, ФСБ России в 2014 г. пресекла 74 млн кибератак на официальные сайты российских структур и государственных органов. В то же время подобные проблемы все чаще становятся предметом обсуждения совета директоров коммерческих компаний. По данным глобального исследования по вопросам обеспечения ИБ за 2015 г. в 45 % компаний совет директоров принимает участие в формировании стратегии безопасности. По сравнению с прошлым годом компании увеличили бюджеты, выделяемые

на кибербезопасность, на 35 %. Инвестиции компаний для защиты информационных активов в основном направлены на разработку стратегии ИБ, повышение осведомленности сотрудников об угрозах ИБ, определение требований к безопасности поставщиков услуг. Кроме того, значительную часть бюджетных расходов составляет организация службы защиты информации, а также регулярная оценка угроз ИБ от кибератак. Однако при увеличении внимания к кибербезопасности в 2015 г. наблюдается рост количества инцидентов в 2,5 раза по сравнению показателем предыдущего года. Нанесенный ущерб от кибератак на ТС увеличился примерно на 47 % [1].

Анализ публикаций подтверждает значительную актуальность тематики в области разработки моделей и методов распознавания и противодействия компьютерным атакам на информационные и телекоммуникационные системы [2].

Атакой на информационную систему, как правило, называется действие или последовательность связанных между собой действий нарушителя, которые приводят к реализации угрозы за счет использования уязвимостей системы. Под уязвимостью принято понимать слабое место информационной системы, на основе которого возможна успешная реализация угрозы безопасности информации [3].

Распознавание компьютерных атак в процессе функционирования информационной системы может быть осуществлено на основе анализа параметров работы системы по установленным правилам и выявления тех из них, которые характеризуют действие атаки. Системное описание способов и средств защиты информации, направленных на противодействие компьютерным атакам, наиболее рационально осуществить на основе теории распознавания образов, в соответствии с которой объекты компьютерных атак могут быть интерпретированы распознаваемыми образами пространства их признаков [4].

В соответствии с данной теорией компьютерная атака является образом, необходимо распознаваемым в ходе процесса сбора, хранения, обработки и передачи информации в ИС, при попытках нарушителя воздействовать на ИС с целью вывода ее из строя или снижения эффективности ее применения.

Словарь признаков компьютерных атак может содержать количественные и качественные признаки. Данные признаки декомпозируются на детерминированные признаки атак, распознаваемые сигнатурными методами обнаружения атак, вероятностные признаки атак,

распознаваемые методами анализа аномальных отклонений в ИС, логические признаки атак, распознаваемые методами функционального анализа.

Пространство признаков атак формируется, декомпозируется и систематизируется на основе предшествующего опыта и знаний об эксплуатации самой ИС, а впоследствии уточняется по апостериорной информации. При этом ключевыми средствами и источниками информации для распознавания объектов компьютерных атак и диагностики потенциальных угроз их воздействия на ИС могут быть интеллектуальные датчики средств противодействия. Средства противодействия компьютерным атакам совместно со средствами мониторинга состояния информационной безопасности осуществляют сбор данных от датчиков. На основе данных формируется пространство признаков атак и возможна последующая оценка ожидаемого нарушения устойчивости функционирования ИС.

В целях эффективного распознавания вероятных информационных атак нарушителя по отношению к ТС и ее последующей оценки по уровню последствий разрабатываемая модель кибератаки должна удовлетворять минимальным требованиям [5]:

- универсальностью, позволяющей использовать модель для представления различных типов атак, т.е. модель должна быть построена таким образом, чтобы она могла применяться в процессе моделирования как сетевых, так и локальных атак;

- формализуемостью, указывающей на возможность использования математического аппарата при описании параметров модели;

- расширяемостью, т.е. возможностью добавления в модель новых характеристик атаки, базирующихся на параметрах модели. Данное свойство позволило бы изменять характеристики моделируемой атаки в зависимости от среды ТС, в которой она рассматривается;

- многофакторностью, дающей возможность учитывать три основных параметра моделируемой атаки: уязвимость, активизируемую атакой, способ реализации атаки и её возможные последствия;

- простотой, позволяющей эксперту легко воспринимать структуру и способы реализации моделируемой атаки, что напрямую позволяет эффективно использовать построенную модель.

На рисунке представлена схема реализации этапов жизненного цикла типовой информационной атаки на ТС.

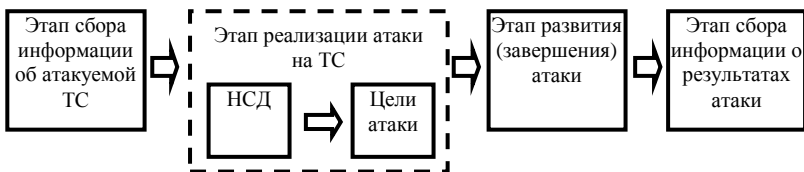


Рис. Схема реализации этапов информационной атаки на ТС

1. Сбор информации – основной этап. На данном этапе выбирается цель нападения, собирается информация о ТС (конфигурация, сервисы), идентифицируются наиболее уязвимые места атакуемой системы, воздействие на которые приводит к нужному результату, выбирается тип реализуемой атаки.

2. Этап реализации атаки. На этом этапе нарушитель осуществляет несанкционированный доступ (НСД) к ресурсам тех узлов ТС, по отношению к которым осуществляется атака. Если по характеру воздействия атака является активной [6], то данный этап сопровождается также реализацией тех целей, ради которых предпринималась атака. Результатом таких действий может являться нарушение конфиденциальности, целостности и доступности информации. Кроме того, на данном этапе может происходить скрытие источника и факта атаки с целью ухода от ответственности за совершаемое действие.

3. На этапе дальнейшего развития атаки выполняются действия, которые направлены на продолжение атаки на ресурсы других узлов ТС. Если атака носит пассивный характер, то данный этап является завершающим.

При моделировании информационной атаки необходимо определить ее параметры и характеристики. Основными параметрами атаки являются время действия атаки, ее кратность, а также список уязвимостей, используемых для реализации атаки.

Моделирование некоторых типов атак, например DDoS («распределенный отказ в обслуживании»), требует определения дополнительных параметров атаки по таким признакам [7]:

- по темпу атаки, который может быть постоянным или переменным. В последнем случае интенсивность атаки меняется во времени. Изменение темпа может быть нарастающим или колеблющимся;
- по типу атаки: семантическая атака (TCP SYN, Incorrect packets, Hard request и др.), или атака типа «грубая сила», такие как UDP/ICMP flood, smurf/fraggle и др.;

– по постоянству набора атакующих узлов (набор может быть постоянным (атакуют одни и те же узлы) или переменным);

– по степени влияния на цель атаки. При этом можно выбрать «подрывную» атаку, когда распределенная атака будет осуществляться сразу со всех атакующих узлов, или ухудшающую, когда число атакующих узлов включается в атаку постепенно. Первый вариант атаки легче обнаружить;

– по степени автоматизации (автоматизированная или автоматическая).

Таким образом, рассмотренные требования к моделированию информационных атак на ТС позволяют с разной степенью детализации описать процесс информационной атаки. В свою очередь, разработка моделей может осуществляться на различной математической базе, что является задачей для дальнейших научных исследований.

Библиографический список

1. Спецпроект: Ответ бизнеса на актуальные вызовы и угрозы // Information Security / Информационная безопасность. – 2015.– № 5. – URL: <http://www.itsec.ru/articles2/focus/otvet-biznesa-na-aktualnye-vyzovy-i-ugrozy> (дата обращения: 12.11.2015).

2. Мазин А.В., Ключко О.С. Анализ методов противодействия угрозам и атакам на вычислительные системы // Научные технологии в приборостроении и развитии инновационной деятельности в вузе: материалы всерос. науч.-техн. конф. Т. 3. – 2014. – С. 71–75.

3. Лукацкий А.В. Обнаружение. – СПб.: БВХ-Петербург, 2001. – 624 с.

4. Климов С.М., Сычёв М.П., Астрахов А.В. Противодействие компьютерным атакам. Методические основы: электронное учеб. издание. – М.: МГТУ им. Н.Э. Баумана, 2013. – 108 с.

5. Казимир В.В., Серая А.А. Метод построения моделей информационных атак // Математические машины и системы. – 2010. – Вып. 4, Т. 1. – С. 52–61.

6. Натров В.В. Классификация сетевых атак // Информационные технологии в управлении и моделировании: сб. докладов. – Белгород, 2005. – С. 128–132.

7. Котенко И.В., Уланов А.В. Моделирование противоборства программных агентов в Интернете: общий подход, среда моделирования и эксперименты // Защита информации. INSIDE.– 2006. – № 5. – С. 2–10.

СОДЕРЖАНИЕ

Секция 1. НОВЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И СИСТЕМЫ	3
Андреев Р.А., Феофилова П.А., Бадртдинов А.С., Кротова Е.Л. Основные методы реализации концепции BYOD	4
Ахметзянов Д.И., Посохин И.А., Файзрахманов Р.А. Разработка онтологической модели технологического графа задачи календарного планирования	10
Бурьлов А.В., Шилов В.С. Архитектура подсистемы формирования маршрута режущего инструмента программного комплекса раскроя «Itas Nesting»	19
Ветчанинов А.А., Мурзакаев Р.Т. Имитация действия силы тяжести на плоские контуры	26
Голохвастова М.В., Проданов Д.Е., Шаякбаров Н.Ф. Анализ сетевого и ленточного графика в задачах планирования производства	32
Иванов А.С. Мини-компьютеры	39
Конюхов М.И. Модели реальных систем при обучении технических специалистов	44
Макагонов Н.Г. Разработка системы детектирования улыбающихся лиц на изображениях	46
Михайлова И.Т., Морозов М.А., Голохвастова М.В. Проектирование и реализация информационной базы данных для планирования машиностроительного производства	51
Поляков А.Н., Сорока Д.П. Реализация алгоритма обратного преобразования управляющих программ для станков с ЧПУ	56
Рутин В.А., Шакирзянов А.Р., Шилов В.С. Поиск участков совмещенного реза в редакторе карт раскроя	62
Собакина Ю.В., Набатов А.В. Использование активных методов обучения при разработке интерактивных обучающих систем	68
Соловьева Н.В., Синицын Н.А., Шилов В.С. Проверка валидности управляющей программы	72

Феофилова П.А., Андреев Р.А., Кокоулин А.Н. Вопросы биометрической аутентификации в операционных системах Windows и Linux	76
Хабибуллин А.Ф., Шкляев Ф.И. Моделирование колебаний троса для тренажера портального крана	82
Шестаков Д.А., Братчиков И.А. Разработка технологии создания мобильных приложений на основе инфраструктуры Apache Cordova	89
Шилов В.С. Группировка деталей при решении задач фигурного раскроя листового материала	94
Секция 2. АВТОМАТИЗАЦИЯ И УПРАВЛЕНИЕ ТЕХНОЛОГИЧЕСКИМИ ПРОЦЕССАМИ И ПРОИЗВОДСТВАМИ	99
Архипов К.Б., Радостев П.Е. Исследование методов анализа пространственных данных на клиентской части веб-приложений	100
Глухов Д.А., Катаев Я.А., Туктамышев В.Р., Гришарин А.О. Современные подходы к управлению показателями технологичности	106
Глухов Р.И., Елисеев А.С. Разработка веб-редактора UML-моделей	112
Каверин А.А., Баянов Ф.Н. Построение математической модели технологического участка учебно-экспериментальной установки целлюлозно-бумажного производства	119
Кацко Е.В. Микропроцессорная система управления температурой картофелехранилища	126
Клещевникова А.А., Петроченков А.Б. Внедрение современного информационного решения в области подготовки производства на крупных промышленных предприятиях	134
Лялин Д.А. Алгоритм преобразования плоских заготовок в объемные для трехмерной гравитационной модели	139
Макагонов Н.Г. Детектирование объекта управления с помощью компьютерного зрения в системе «Шарик на балансире»	143
Софин Н.А. Модернизация фреймворка на языке C# для разработки программного обеспечения верхнего уровня	150
Софин Н.А. Модернизация аппаратно-программного обеспечения системы управления перевёрнутым маятником на тележке с применением беспроводных технологий	158

Чудинов М.А. Особенности применения OpenSCADA для реализации программного обеспечения систем управления	166
Чудинов М.А. Реализация взаимодействия OpenSCADA с микроконтроллером Arduino	173
Шачков А.М., Солодкий Е.М. Разработка системы векторного управления асинхронным двигателем с настройкой регуляторов по параметрам схемы замещения	181
Секция 3. ИННОВАЦИОННЫЕ НАПРАВЛЕНИЯ В ЭНЕРГЕТИКЕ. ЭНЕРГОРЕСУРСОСБЕРЕЖЕНИЕ	189
Бояршинова В.В., Елтышев Д.К. Использование нейронечетких моделей для комплексной оценки состояния электротехнического оборудования	190
Косташ Д.И., Трефилов В.А. Расчет рабочих характеристик синхронного двигателя в двухмашинном агрегате нагрузочного устройства синхронных машин	200
Кулик В.Д., Елтышев Д.К. Анализ состояния объекта по результатам мониторинга его энергетических параметров	205
Микрюков Г.П., Кычкин А.В. Разработка структуры системы энергомониторинга активно-адаптивной сети на основе WAMS-технологии	214
Мошкин С.А., Елтышев Д.К. Аналитический обзор подходов к оценке приоритетов при эксплуатации электроэнергетических объектов	221
Носков В.В., Кычкин А.В., Яблоков А.А. Построение многошинной сетевой архитектуры системы сбора энергетических данных	227
Прохоров А.С. Энергосберегающая логика	233
Чудинов А.В., Кычкин А.В. CIM-модель локального сегмента активно-адаптивной сети HIL MicroGrid	235
Секция 4. ТЕЛЕКОММУНИКАЦИОННЫЕ СИСТЕМЫ И СЕТИ. ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ	243
Кокоулин А.Н., Бадртдинов А.С., Тихоненко М.А. Разработка простейшей системы обнаружения вторжений	244
Борисов В.И., Шабуров А.С. О применении SIEM-систем для обеспечения безопасности корпоративных сетей	249

Васиньков В.А. Построение модели приложения узла промышленной распределённой информационно-управляющей системы	255
Гайнанова Д.Н. Анализ протокола доступа и передачи технологии децентрализованной автоматизации EIB/KNX	262
Горожанкин А.И. Модель функционирования транспортной подсистемы распределённой информационно-управляющей системы	268
Елтышев А.В. Аналитическое и имитационное моделирование самомаршрутизирующегося нейросетевого аналого-цифрового преобразователя	274
Журилова Е.Е., Шабуров А.С. О внедрении доверенной информационной среды городской инфраструктуры на основе программно-аппаратного комплекса «Средство обеспечения доверенного сеанса “МАРШ!”»	281
Кокоулин А.Н., Зее С.Д., Тур А.И. Вопросы интеграции смарт-карт в ОС Windows	288
Лихачева Ю.В. Постановка задачи исследования эффективности протоколов множественного доступа	295
Лужнов В.С., Соколов А.Н. Математическая модель атак на информационные ресурсы корпоративных автоматизированных систем	299
Маро Е.А. Анализ стойкости симметричных шифров ГОСТ Р 34.12-2015	305
Олейников А.А., Тарутин А.В. К проблеме надежности вычислительной сети	310
Пукса А.О. Проектирование систем связи стандарта GSM-1800 в районах с малой плотностью населения	316
Пукса А.О. Снижение пик-фактора сигналов с OFDM с помощью блочного кодирования	322
Рангулов А.В. О разработке учебно-лабораторного комплекса для построения систем защиты информации от несанкционированного доступа на основе «Аккорд-АМДЗ 5.5»	326
Рогожин В.П., Фрейман В.И. Декодирование в каналах связи с пакетирующимися ошибками методом посимвольного перемежения и с использованием порождающего полинома $G(X^i)$	331

Салимзебаров Э.Д., Кокоулин А.Н. Методы защиты от изменений и дизассемблирования .NET-приложений	338
Тиунов С.Ю., Гаврилов А.В. Анализ функциональных возможностей свободно распространяемых SNMP-менеджеров	344
Федотов Д.В. Обзор систем имитационного моделирования сетей связи	349
Кон М.М., Чмыков В.В. Решение проблемы масштабируемости программной модели приложения узла распределённой информационно-управляющей системы	356
Шипицин С.П., Шабуров А.С. О методе встраивания цифровых водяных знаков в видеопотоки для защиты от информационных атак	364
Ямаев М.И. Преимущества и недостатки системы защиты авторского права Youtube «Content ID» по видеопотоку	370
Ямаев М.И., Кротова Е.Л. Альтернативный алгоритм протокола «Подбрасывания монеты по телефону»	375
Миронова А.А., Шабуров А.С. О требованиях к разработке модели компьютерных атак на телекоммуникационные системы	379

Научное издание

**ИННОВАЦИОННЫЕ ТЕХНОЛОГИИ: ТЕОРИЯ,
ИНСТРУМЕНТЫ, ПРАКТИКА**

Материалы VII Международной
интернет-конференции молодых ученых,
аспирантов, студентов
(01 ноября 2015 г. – 30 ноября 2015 г.)

Редактор и корректор *И.Н. Жеганина*

Подписано в печать 13.04.2016.
Формат 60×90/16. Усл. печ. л. 20,5.
Тираж 100 экз. Заказ 47/2016.

Издательство
Пермского национального исследовательского
политехнического университета.
Адрес: 614990, г. Пермь, Комсомольский пр., 29, к. 113.
Тел. (342) 219-80-33.