

Momenty wychowawcze

ISSN 2072-0297

МОЛОДОЙ

учёный

международный научный журнал

Koszałki-Opałki

Król

Maciuś Pierwszy

Dzieci ulicy

Moski, Joški i Srule

Jak kochać dziecko

Sam na sam z Bogiem:
modlitwy tych, którzy się nie modlą



16+

12
2016
Часть II



ISSN 2072-0297

Молодой учёный

Международный научный журнал

Выходит два раза в месяц

№ 12 (116) / 2016

РЕДАКЦИОННАЯ КОЛЛЕГИЯ:

Главный редактор: Ахметов Ильдар Геннадьевич, кандидат технических наук

Члены редакционной коллегии:

Ахметова Мария Николаевна, доктор педагогических наук

Иванова Юлия Валентиновна, доктор философских наук

Каленский Александр Васильевич, доктор физико-математических наук

Куташов Вячеслав Анатольевич, доктор медицинских наук

Лактионов Константин Станиславович, доктор биологических наук

Сараева Надежда Михайловна, доктор психологических наук

Абдрасилов Турганбай Курманбаевич, доктор философии (PhD) по философским наукам

Авдюк Оксана Алексеевна, кандидат технических наук

Айдаров Оразхан Турсункожаевич, кандидат географических наук

Алиева Тарана Ибрагим кызы, кандидат химических наук

Ахметова Валерия Валерьевна, кандидат медицинских наук

Брезгин Вячеслав Сергеевич, кандидат экономических наук

Данилов Олег Евгеньевич, кандидат педагогических наук

Дёмин Александр Викторович, кандидат биологических наук

Дядюн Кристина Владимировна, кандидат юридических наук

Желнова Кристина Владимировна, кандидат экономических наук

Жуйкова Тамара Павловна, кандидат педагогических наук

Жураев Хусниддин Олтинбоевич, кандидат педагогических наук

Игнатова Мария Александровна, кандидат искусствоведения

Калдыбай Кайнар Калдыбайулы, доктор философии (PhD) по философским наукам

Кенесов Асхат Алмасович, кандидат политических наук

Коварда Владимир Васильевич, кандидат физико-математических наук

Комогорцев Максим Геннадьевич, кандидат технических наук

Котляров Алексей Васильевич, кандидат геолого-минералогических наук

Кузьмина Виолетта Михайловна, кандидат исторических наук, кандидат психологических наук

Кучерявенко Светлана Алексеевна, кандидат экономических наук

Лескова Екатерина Викторовна, кандидат физико-математических наук

Макеева Ирина Александровна, кандидат педагогических наук

Матвиенко Евгений Владимирович, кандидат биологических наук

Матроскина Татьяна Викторовна, кандидат экономических наук

Матусевич Марина Степановна, кандидат педагогических наук

Мусаева Ума Алиевна, кандидат технических наук

Насимов Мурат Орленбаевич, кандидат политических наук

Паридинова Ботагоз Жаппаровна, магистр философии

Прончев Геннадий Борисович, кандидат физико-математических наук

Семахин Андрей Михайлович, кандидат технических наук

Сенцов Аркадий Эдуардович, кандидат политических наук

Сенюшкин Николай Сергеевич, кандидат технических наук

Титова Елена Ивановна, кандидат педагогических наук

Ткаченко Ирина Георгиевна, кандидат филологических наук

Фозилов Садриддин Файзуллаевич, кандидат химических наук

Яхина Асия Сергеевна, кандидат технических наук

Ячинова Светлана Николаевна, кандидат педагогических наук

На обложке изображен Януш Корчак (1878–1942) — выдающийся польский педагог, писатель, врач и общественный деятель.

Журнал зарегистрирован Федеральной службой по надзору в сфере связи, информационных технологий и массовых коммуникаций.

Свидетельство о регистрации средства массовой информации ПИ № ФС77-38059 от 11 ноября 2009 г.

Журнал входит в систему РИНЦ (Российский индекс научного цитирования) на платформе elibrary.ru.

Журнал включен в международный каталог периодических изданий «Ulrich's Periodicals Directory».

Статьи, поступающие в редакцию, рецензируются. За достоверность сведений, изложенных в статьях, ответственность несут авторы. Мнение редакции может не совпадать с мнением авторов материалов. При перепечатке ссылка на журнал обязательна.

Международный редакционный совет:

Айрян Заруи Геворковна, *кандидат филологических наук, доцент (Армения)*

Арошидзе Паата Леонидович, *доктор экономических наук, ассоциированный профессор (Грузия)*

Атаев Загир Вагитович, *кандидат географических наук, профессор (Россия)*

Ахмеденов Кажмурат Максutowич, *кандидат географических наук, ассоциированный профессор (Казахстан)*

Бидова Бэла Бертовна, *доктор юридических наук, доцент (Россия)*

Борисов Вячеслав Викторович, *доктор педагогических наук, профессор (Украина)*

Велковска Гена Цветкова, *доктор экономических наук, доцент (Болгария)*

Гайич Тамара, *доктор экономических наук (Сербия)*

Данатаров Агахан, *кандидат технических наук (Туркменистан)*

Данилов Александр Максимович, *доктор технических наук, профессор (Россия)*

Демидов Алексей Александрович, *доктор медицинских наук, профессор (Россия)*

Досманбетова Зейнегуль Рамазановна, *доктор философии (PhD) по филологическим наукам (Казахстан)*

Ешиев Абдыракман Молдоалиевич, *доктор медицинских наук, доцент, зав. отделением (Кыргызстан)*

Жолдошев Сапарбай Тезекбаевич, *доктор медицинских наук, профессор (Кыргызстан)*

Игисинов Нурбек Сагинбекович, *доктор медицинских наук, профессор (Казахстан)*

Кадыров Кутлуг-Бек Бекмурадович, *кандидат педагогических наук, заместитель директора (Узбекистан)*

Кайгородов Иван Борисович, *кандидат физико-математических наук (Бразилия)*

Каленский Александр Васильевич, *доктор физико-математических наук, профессор (Россия)*

Козырева Ольга Анатольевна, *кандидат педагогических наук, доцент (Россия)*

Колпак Евгений Петрович, *доктор физико-математических наук, профессор (Россия)*

Куташов Вячеслав Анатольевич, *доктор медицинских наук, профессор (Россия)*

Лю Цзюань, *доктор филологических наук, профессор (Китай)*

Малес Людмила Владимировна, *доктор социологических наук, доцент (Украина)*

Нагервадзе Марина Алиевна, *доктор биологических наук, профессор (Грузия)*

Нурмамедли Фазиль Алигусейн оглы, *кандидат геолого-минералогических наук (Азербайджан)*

Прокопьев Николай Яковлевич, *доктор медицинских наук, профессор (Россия)*

Прокофьева Марина Анатольевна, *кандидат педагогических наук, доцент (Казахстан)*

Рахматуллин Рафаэль Юсупович, *доктор философских наук, профессор (Россия)*

Ребезов Максим Борисович, *доктор сельскохозяйственных наук, профессор (Россия)*

Сорока Юлия Георгиевна, *доктор социологических наук, доцент (Украина)*

Узаков Гулом Норбоевич, *доктор технических наук, доцент (Узбекистан)*

Хоналиев Назарали Хоналиевич, *доктор экономических наук, старший научный сотрудник (Таджикистан)*

Хоссейни Амир, *доктор филологических наук (Иран)*

Шарипов Аскар Калиевич, *доктор экономических наук, доцент (Казахстан)*

Руководитель редакционного отдела: Кайнова Галина Анатольевна

Ответственные редакторы: Осянина Екатерина Игоревна, Вейса Людмила Николаевна

Художник: Шишков Евгений Анатольевич

Верстка: Бурьянов Павел Яковлевич, Голубцов Максим Владимирович, Майер Ольга Вячеславовна

Почтовый адрес редакции: 420126, г. Казань, ул. Амирхана, 10а, а/я 231.

Фактический адрес редакции: 420029, г. Казань, ул. Академика Кирпичникова, д. 25.

E-mail: info@moluch.ru; <http://www.moluch.ru/>.

Учредитель и издатель: ООО «Издательство Молодой ученый».

Тираж 500 экз. Дата выхода в свет: 15.07.2016. Цена свободная.

Материалы публикуются в авторской редакции. Все права защищены.

Отпечатано в типографии издательства «Молодой ученый», 420029, г. Казань, ул. Академика Кирпичникова, д. 25.

СОДЕРЖАНИЕ

ИНФОРМАТИКА

Алёшин В. В., Сумкин К. С. Проверка корректности программного обеспечения.....	139
Бурмистров В. Д., Заковряшин Е. М. Создание хранилища данных для распределённой системы.....	143
Глебов А. П. Разработка диспетчера параллельного исполнения задач для формирования изображения в авиатренажёре.....	147
Григорьев Ю. А., Качалкин К. И. MapReduce и метод доступа к хранилищу MRIJ on RCFile.....	151
Григорьев Ю. А., Эрдынеев М. Ц. Nadoop и метод доступа к хранилищу MRIJ....	154
Журавлева Л. В., Стригулин К. А. Анализ тональности отзывов пользователей в метаобласти фильмов.....	157
Коробов Д. В., Патин М. В. Метод распознавания шрифта текста с изображения.....	161
Лещёв В. А. Эффективность применения CRM-системы.....	165
Лещёв В. А. Технология тестирования программных модулей.....	168
Романова В. О. Обзор методов и средств автоматизированного сбора информации с новостных лент.....	170
Рудников А. В., Шувалов А. А. Анализ систем управления содержимым для создания информационного портала.....	173
Салтецкая Т. В., Конарёва Е. А. Автоматизированное заполнение плана проведения медицинских осмотров для сотрудников любого предприятия (на примере медицинского центра ООО «Олеся»).....	175
Степаненко А. А. Разработка программных модулей обработки многомерных данных различной природы в среде EXCEL.....	180
Стригулин К. А., Журавлёва Л. В. Анализ тональности высказываний в Twitter ...	185

Сумкин К. С., Тараненко Л. О. Анализ страницы пользователя социальной сети «ВКонтакте».....	189
Усачёв В. В., Рагулин П. Г. Концептуальная модель масштабируемого сервиса социальной сети.....	194

ТЕХНИЧЕСКИЕ НАУКИ

Александров А. А. Роль сетевой модели OSI в построении промышленных сетей.....	199
Аронов Л. В. Метод расчета активного сопротивления цилиндрического провода с учетом поверхностного эффекта.....	202
Бабченко Л. Ю., Багирян М. А., Патиева А. М. Колбасные изделия из мяса индейки, обогащенные йодказеином.....	205
Бобков Н. В. Разработка баз данных для электроэрозионных станков.....	209
Бруданов А. М. Выбор геометрических параметров ограничителя переворота гусеничного движителя устанавливаемого на автомобиль УАЗ-Патриот.....	213
Бруданов А. М. Обзор существующих конструкций для повышения проходимости автомобиля категории М1.....	216
Буш И. В. Результаты определения параметров сопротивления грунтов сдвигу в приборе трехосного сжатия.....	220
Вардияшвили А. А., Мурадов М. О., Вардияшвили А. А., Сайдалиев А. А., Холов К. Н. Комбинированное использование солнечной энергии и тепловых отходов в гелиотеплице.....	223
Великанова В. А., Степанец В. Г. Применение щебеночно-мастичной асфальтобетонной смеси при строительстве покрытий в Алтайском крае.....	226

Виноградов О. В., Кирсанов Е. А. Обеспечение и подтверждение надежности модульных транспортных средств.....	230	Ганиева З. Г. Изотерма и дифференциальные теплоты адсорбции бензола на цеолитах LiZSM-5, CsZSM-5 и NaZSM-5	239
Виноградов О. В., Кирсанов Е. А. Применение методов исследований и испытаний традиционных транспортных средств для испытаний модульных транспортных средств.....	232	Грязнова М. К. Модифицированный динамический инженерный метод расчета главных напряжений в дискретных материалах дорожных конструкций	241
Гаевская В. А. Обзор традиционных и современных расчетов дорожных конструкций.....	234	Джанаев С. И. Обзорное описание метода Multi-Fragment- ReplicationJoin (MFRJ) доступа к многомерному хранилищу данных по технологии MapReduce	244

ИНФОРМАТИКА

Проверка корректности программного обеспечения

Алёшин Владислав Владимирович, студент;
Сумкин Константин Сергеевич, кандидат технических наук, доцент
Московский технологический университет

Перед современной IT индустрией стоит непростая задача. Увеличивается количество программных комплексов, призванных решать различные бизнес-задачи, а также увеличивается их сложность. В связи с этим остро встает вопрос о контроле бизнес-требований, предъявляемых к продукту.

Для решения этой задачи в данной статье осуществляется обзор различных методов контроля качества программного обеспечения и предлагается новый подход. Подход объединяет преимущества контрактного программирования для контроля корректности мощь темпоральных логик для описания бизнес-требований.

В данной статье рассмотрены:

- Актуальность проблемы. Необходимость контроля корректности ПО.
- Подходы к автоматизированному контролю корректности ПО.
- Плюсы и минусы каждого подхода.

Далее проведен анализ всех предлагаемых подходов.

На основании проведенного анализа и изложенных результатов сформулированы основные проблемы, мешающие массовому использованию рассмотренных подходов в промышленных масштабах, поставлены задачи и цели данной работы.

При разработке информационных систем очень важно контролировать качество программного обеспечения. Понятие качества программного обеспечения сформулировано стандарте ISO 9126 [1]. На сегодняшний день, действует редакция, принятая в 2001 году. По этому стандарту качество можно рассматривать как совокупность различных аспектов. Совокупность этих аспектов представлена на рисунке 1.

Одним из самых важных аспектов качества, представленных на диаграмме, является функциональность. Функциональность определяет способность ПО решать поставленные задачи. При этом очевидно, что важнейшей характеристикой функциональности является корректность — соответствие создаваемого ПО заданной спецификации. Если ПО не соответствует своей специфи-

кации, то таким продуктом нельзя пользоваться, даже если он надежен, производителен и удобен в использовании.

Контроль корректности ПО может осуществляться двумя способами — вручную и автоматизировано. При ручном контроле корректности огромную роль при получении результата играет человеческий фактор. Чтобы избежать ошибок, возникающих в процессе ручного контроля корректности, используют полную или частичную автоматизацию процесса.

Поэтому важнейшей задачей в IT индустрии является обеспечение автоматизированного контроля корректности ПО.

Все подходы к контролю корректности программного обеспечения можно разделить на две большие группы:

1. контроль качества программного обеспечения на этапе создания;
2. контроль качества программного обеспечения на этапе выполнения.

Подходы из первой группы позволяют проводить анализ корректности работы программного обеспечения еще до его запуска и использования. Такие подходы делятся на три большие группы:

- статический анализ;
- тестирование;
- верификация.

Подходы из второй группы проявляют себя уже после компиляции продукта и старта его использования. Их основной задачей является завершение или приостановка работы программы в случае обнаружения какой-либо критической ошибки. Считается, что гораздо лучше прекратить (или приостановить) выполнение программы, нежели продолжать некорректное выполнение. Также динамический контроль позволяет быстрее находить и устранять ошибки, которые не были выявлены на более ранних этапах.

Критическими ошибками в таких случаях могут служить:

- Нулевое значение знаменателя при выполнении операции целочисленного деления.



Рис. 1

— Ошибка при попытке коммуникации с недоступным внешним устройством.

— Истощение доступной памяти или переполнение стека.

— Появление сигнала аварийного отключения электропитания системы.

— Нарушение бизнес-требований, заложенных при создании программы. Например, выдача чека с отрицательной суммой заказа. В рамках представленной работы данный пункт является наиболее интересным и будет исследован более подробно.

Такие подходы можно также разделить на три группы, в порядке появления и развития подходов:

- коды ошибок;
- исключения;
- контракты.

Далее перечисленные выше подходы рассмотрены более подробно, а также произведено их сравнение.

Статический анализ используется для поиска часто встречающихся дефектов по некоторым шаблонам. Такой анализ хорошо автоматизируется. Однако он способен обнаружить лишь малое число из всех возможных ошибок.

Большинство техник статической проверки корректности программ, доказавших свою эффективность, рано или поздно становятся частью средств разработки, компиляторов, а иногда преобразуются в семантические правила языков программирования.

Тестирование является динамическим методом проверки программ. Для его проведения необходима работающая система, какой-то ее компонент, либо прототип.

Тестирование позволяет обнаруживать в системе ошибки, проявляющиеся во время работы. Однако, для проведения тестирования необходимо подготовить набор тестов, а иногда разработать саму тестовую систему. И, если тестовую систему еще можно повторно использовать при разработке очередного программного продукта, то набор тестов разработчикам приходится писать каждый раз заново, учитывая особенности и требования конкретной системы. На практике существует множество различных средств, позволяющих частично автоматизировать тестирование. Например, достаточно часто используется вероятностное тестирование, позволяющее быстро проверить корректность работы системы, при подаче ей на вход каких-либо случайных данных.

Существенным недостатком тестирования является то, что оно, как правило, помогает выявить значительное число ошибок, но гарантировать, что система будет работать, корректно не может.

Верификация является статическим методом проверки программы. Верификация относится к формальным методам и применяется только к тем свойствам системы, которые можно выразить с помощью формальной математической модели.

Проблему при данном подходе представляет собой необходимость построения математической модели, а также формальное описание требований, которым эта математическая модель, а, следовательно, и проектируемая система, должны удовлетворять.

Достоинством данного метода может служить тот факт, что верификация гарантирует корректность работы си-

стемы по всем проверяемым свойствам, если они успешно прошли проверку на модели.

На рисунке 2 сведены воедино ключевые характеристики всех трех подходов. Каждый подход имеет свои достоинства и недостатки. Однако если рассматривать все эти подходы с точки зрения улучшения корректности ПО,

наиболее интересным является подход на основе верификации.

Только верификация позволяет *гарантировать* корректность работы программы. Именно поэтому верификация программ представляет наибольший интерес на сегодняшний день.



Рис. 2

Код ошибки — это номер (или сочетание буквы и номера), который соответствует конкретной проблеме в работе программы. Коды ошибок используются для идентификации неправильной работы аппаратного и программного обеспечения, неверного ввода данных пользователем без обработки, возникающей при этом исключительной ситуации в коде программы.

Хотя иногда коды ошибок используются в сочетании с обработкой исключений. Также коды ошибок часто используются вместе с кодами возврата.

Во время выполнения программы могут возникать ситуации, когда состояние данных, устройств ввода-вывода или компьютерной системы в целом делает дальнейшие вычисления в соответствии с базовым алгоритмом невозможными или бессмысленными. В этом случае системе генерирует исключение, уведомляя пользователей о возникновении такой ситуации.

В отсутствие собственного механизма обработки исключений для прикладных программ наиболее общей реакцией на любую исключительную ситуацию является немедленное прекращение выполнения с выдачей пользователю сообщения о характере исключения. Можно сказать, что в подобных случаях единственным и универ-

сальным обработчиком исключений становится операционная система.

Возможно игнорирование исключительной ситуации и продолжение работы, но такая тактика опасна, так как приводит к ошибочным результатам работы программ или возникновению ошибок впоследствии. Например, проигнорировав ошибку чтения из файла блока данных, программа получит в своё распоряжение не те данные, которые она должна была считать, а какие-то другие. Результаты их использования предугадать невозможно.

Обработка исключительных ситуаций самой программой заключается в том, что при возникновении исключительной ситуации управление передаётся некоторому заранее определённом обработчику — блоку кода, процедуре, функции, которые выполняют необходимые действия.

Исключения позволяют повысить безопасность и качество ПО, предотвращая выполнение программы в случае возникновения каких-либо ошибок. В смысле контроля качества, исключения гораздо надежнее кодов ошибок, так как неверный код ошибки гораздо легче пропустить или намеренно игнорировать.

Основная идея контрактного программирования — это модель взаимодействия элементов программной системы, основывающаяся на идее взаимных обязательств и преимуществ. Контракт некоторого метода или функции может включать в себя:

1. конкретные обязательства, которые любой клиентский модуль должен выполнить перед вызовом метода — предусловия;
2. конкретные свойства, которые должны присутствовать после выполнения метода — постусловия, которые входят в обязательства поставщика;
3. обязательства по выполнению конкретных свойств — инвариантов, которые должны выполняться при получении поставщиком сообщения, а также при выходе из метода.

В объектно-ориентированном программировании контракт метода обычно включает следующую информацию:

- возможные типы входных данных и их значение;
- типы возвращаемых данных и их значение;
- условия возникновения исключений, их типы и значения.

При использовании контрактов сам код не обязан проверять их выполнение. Обычно при нарушении контракта выбрасывается исключение, сообщающее о возникшем несоответствии.

Таким образом, контракты представляю собой модель контроля логических ошибок, или, другими словами, модель проверки бизнес-правил.

На рисунке 3 сведены воедино ключевые характеристики всех трех подходов. Каждый подход имеет свои досто-

инства и недостатки. Однако если рассматривать все эти подходы с точки зрения улучшения корректности ПО, наиболее интересным является подход на основе контрактов.

Только контрактное программирование позволяет описывать и контролировать выполнение различных бизнес-требований не затрачивая при этом значительных усилий.

Все перечисленные подходы можно показать на диаграмме времени жизни программного продукта (рисунок 4).

Основной целью данной статьи является теоретическое обоснование и разработка инструмента, который бы позволял встраивать в программный продукт требования, записанные в формальном виде и на этапе выполнения проверять их выполнение.

Более конкретно, можно сформулировать следующие подцели данной статьи:

1. Теоретическое обоснование — более подробный обзор контрактного программирования и формализма темпоральных логик. Описание предлагаемого комбинированного подхода.
2. Создание спецификация инструмента на основе теоретических выводов.
3. Практическое сравнение подходов к динамическому контролю качества на примере разработки одного и того же блока кода.
4. Проектирование архитектуры инструмента.
5. Создание инструмента.

Решение обозначенной выше проблемы, позволит увеличить качество программного обеспечения и автоматизировать проверку различных бизнес-требований на этапе выполнения программ, написанных на языках семейства.NET.



Рис. 3

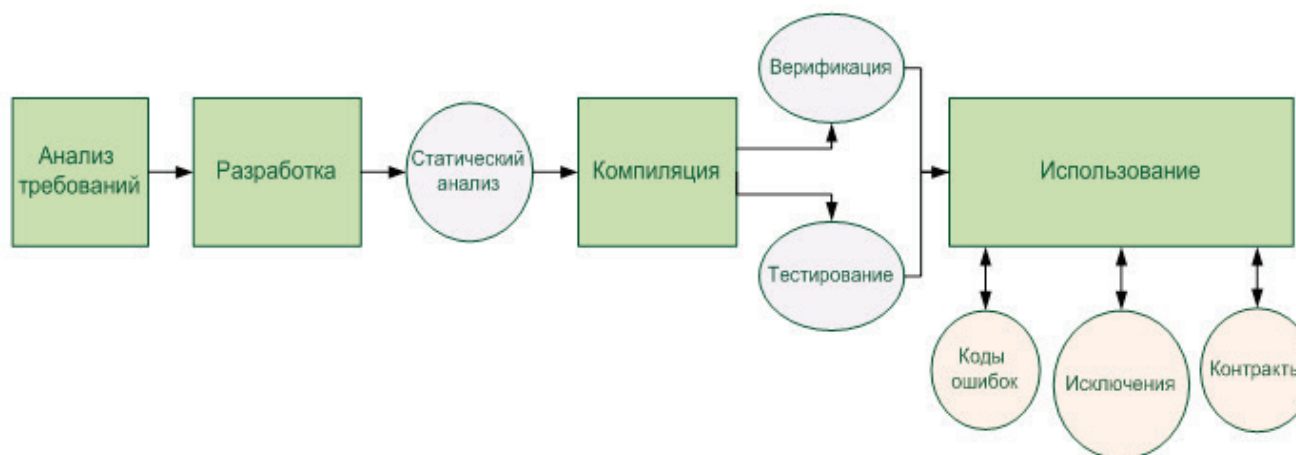


Рис. 4

Также очень важно проводить теоретические исследования в данной области, чтобы на основании полученных

знаний, спроектировать мощный и пригодный к использованию в различных условиях инструмент.

Литература:

1. Введение в программные системы и их разработку Национальный Открытый Университет «ИНТУИТ». 2016 год, 650 страниц
2. Приемы объектно-ориентированного проектирования. Паттерны проектирования. Автор: Э. Гамма, Р. Хелм, Р. Джонсон, Дж. Влиссиде Издательство: Питер Год: 2010, язык: Русский ISBN: 978–5–496–00389–6 Страниц: 366 Источник: <http://forcoder.ru/developing/>
3. Release it! Проектирование и дизайн ПО для тех, кому не все равно Автор: Майкл Нейгард Издательство: СПб.: Питер Год: 2016 Язык: Русский Страниц: 320 Источник: <http://forcoder.ru/developing/>
4. Принципы работы с требованиями к программному обеспечению. Унифицированный подход Автор: Дин Леффингуэлл, Дон Уидриг Издательство: Вильямс Год: 2002 ISBN: 5–8459–0275–4, 0–2016–1593–2 Источник: <http://progbook.ru/proektirovanie-i-razrabotka-po/>

Создание хранилища данных для распределённой системы

Бурмистров Валерий Дмитриевич, студент;
Заковряшин Егор Михайлович, студент
Санкт-Петербургский государственный университет

В данной работе рассматривается реализация модуля хранения данных в распределённой системе, решающей задачи хранения и анализа файлов, содержащих данные в виде текста. Задачами модуля являются размещение файлов, а также управление загруженными файлами. Модуль реализован на языке программирования Java с использованием технологии Openstack Swift.

Ключевые слова: хранилище данных, объектное хранилище, распределённые системы, Openstack Swift, документация

В настоящее время происходит переход от традиционного способа хранения и обработки данных к цифровому. В конце прошлого столетия были предложены концепции «облачных» проектов, реализующих модель приложений, при которых обработка информации осуществлялась бы не на компьютере пользователя, а на удалённых серверах. В то время реализация такого рода

проектов не представлялась возможной из-за отсутствия глобальной сети Интернет.

Сейчас «облачная» модель обработки данных является перспективной и быстроразвивающейся. В такой модели данные хранятся в распределённой системе на основе множества серверов, объединённых общей сетью. В пользу такой модели говорят следующие преимущества:

1. Доступ может предоставляться любому устройству, имеющему подключение к сети Интернет;
2. Сохранность данных в случае сбоя, отказоустойчивость;
3. Неограниченная масштабируемость.

Зачастую, процесс обучения и другие области человеческой деятельности тесно связаны с работой с текстовыми материалами. В настоящее время большая часть этих материалов хранится в электронном виде и их число постоянно растёт. Потому актуальна проблема упорядоченного хранения большого объёма информационных ресурсов, содержащие текстовые данные.

Решение данной проблемы — это создание «облачного» программного продукта, реализующего функции систематизированного хранения файлов, содержащих текстовую информацию.

В данной работе рассматривается задача реализации одного из важнейших модулей распределённой системы — модуля размещения и управления размещёнными в системе файлами. То есть модуля, реализующего такие функции работы с файлами, как создание директорий, перемещение файла, переименование и прочие функции, аналогичные функциям файловых систем.

Инструмент разработки

В качестве инструмента разработки файлового хранилища был выбран проект Openstack Swift. OpenStack — это open source проект по разработке платформы, позволяющей строить частные и публичные «облака». Технология включает в себя серию взаимосвязанных проектов, обеспечивающих разработку многочисленных составляющих инфраструктурного решения для «облака» [1].

Swift (OpenStack Object Storage) — распределённое объектное хранилище, главными особенностями которого являются высокий уровень надёжности и отказоустойчивости, а также лёгкая масштабируемость системы. Доступ к объектам в Swift осуществляется по интерфейсу REST с помощью базовых HTTP запросов.

Swift организован на базе четырёх основных компонент:

- Proxy Server (сервер, объединяющий остальные компоненты вместе). Отвечает за приём пользовательских запросов HTTP, производит поиск местоположение запрошенного объекта с помощью кольца;
- Object Server (непосредственно хранящий данные);
- Container Server (отдаёт список объектов в контейнере);
- Account Server (листинги контейнеров для конкретного аккаунта) Отслеживает имена контейнеров, принадлежащих определённому аккаунту.

При этом, путь доступа к каждому из объектов хранения в Swift состоит из трёх компонент: аккаунт, контейнер, объект и выглядит следующим образом: As-

countName/ContainerName/ObjectName. Под объектом хранения подразумевается хранящийся файл.

Как правило, хранилище данных, созданное с помощью Swift, имеет вид масштабного кластера, одна из машин которого выполняет роль прокси-сервера; некоторое количество машин — контейнер-серверы и аккаунтинг-серверы; остальные же — объектные серверы и их количество может исчисляться сотнями и более машин, в зависимости от нужд потребителя [4].

Отдельно стоит упомянуть структуру данных, используемую Swift для поиска положения данных в системе — «кольцо» (ring). По сути, кольцо является базой данных, описывающей местоположение объектов. При любом действии, связанном с записью или удалением данных в хранилище, а также выходе из строя узлов хранилища, кольцо изменяется.

Важнейшим элементом абсолютно любого хранилища являются объектные серверы, выполняющие функции размещения и отдачи файлов. Объектный сервер — это хранилище BLOB-объектов (большой двоичный объект). Каждый объект хранилища оказывается на жёстком диске объектного сервера и хранится на нём в двоичных файлах, сопровождаемых метаданными, которые, в свою очередь, хранятся в расширенных атрибутах файлов (xattr). Swift обеспечивает надёжное хранение данных, благодаря их репликации сразу в несколько серверов. Таким образом, в случае утери данных с одного из серверов при его отказе, система восстановит данные с другого сервера и снова создаст копию [2].

Реализация поставленной задачи

Для реализации требуемого модуля программного продукта, необходимо реализовать элементы MVC-модели приложения на языке программирования Java. Для этого используется фреймворк Spring, предоставляющий архитектурный каркас для реализации следующих компонентов системы:

- Классы модели представления, получаемые при получении запроса от пользователя и формировании ответа на него;
- Контроллеры, обрабатывающие объекты, представляющие пользовательский запрос, передавая их в соответствующие сервисы;
- Сервисы, реализующие типовые функции, используемые контроллером или другими сервисами.

Кроме того, используется библиотека-провайдер Apache jclouds, обеспечивающая доступ к функциям, предоставляемым облачными средствами, в том числе Swift API [3], позволяющий управлять аккаунтами, контейнерами и объектами в системе объектного хранения.

Таким образом, для решения поставленных задач, связанных с хранением файлов, нужно создать классы, описывающие сущности необходимые для выполнения заданных функций, таких как местоположение файла, его атрибуты, а также, на основе этих классов, сервисы

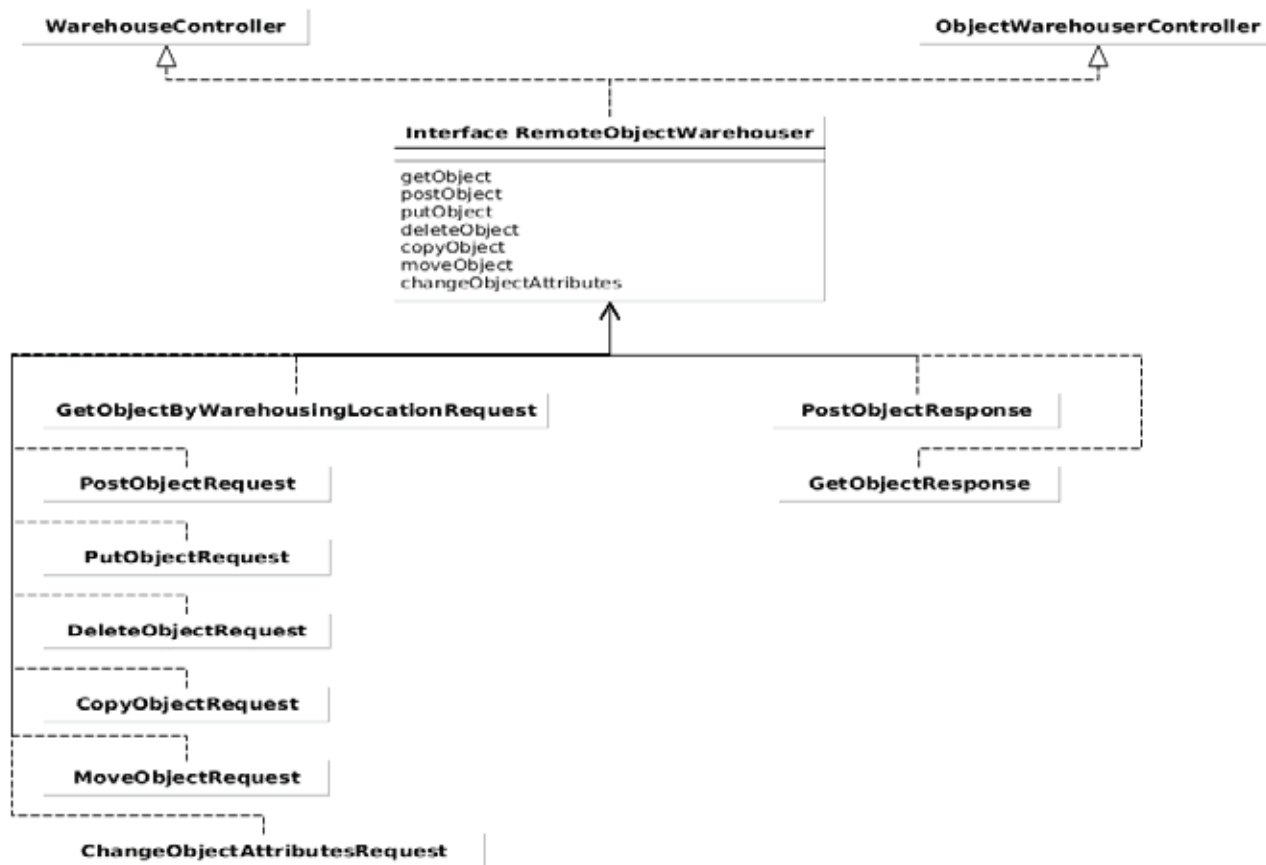


Рис. 1. Взаимодействие элементов модуля хранения

и контроллеры, которые в свою очередь реализуют непосредственно логику осуществления функции и их выполнение посредством обращения к API.

На рисунке 1 продемонстрирована схема взаимодействия данных элементов системы.

Как видно из диаграммы, контроллер WarehouseController и сервис ObjectWarehouseService реализуют методы интерфейса RemoteObjectWarehouseer, используя классы модели сервера хранилища данных. Причём контроллер генерирует соответствующие запросу JSON-запросы, а сервис реализует логику исполнения этих запросов. Методами интерфейса RemoteObjectWarehouseer описываются запросы для взаимодействия с объектами хранения и ответы на эти запросы.

На рисунке 2 представлена диаграмма классов для класса GetObjectByWarehousingLocationRequest, описывающего запрос обращения к объекту хранения. Для выборки хранящегося объекта необходимо установить его местоположение, кроме того, это объект может быть предназначен для публичного доступа, либо для личного, потому необходимо также установить является ли этот объект публично доступным. Эти два требования ясно видны из рисунка 2.

Ниже приведены краткие описания классов, отвечающих за запросы к объектам хранения:

– GetObjectByWarehousingLocationRequest — запрашивает объект хранения, использует для этого, как было

написано ранее, местоположение объекта и данные о его публичности;

– PutObjectRequest — создаёт объект вместе с его метаданными в выбранном контейнере;

– PostObjectRequest — создаёт или обновляет метаданные объекта хранения;

– DeleteObjectRequest — удаляет выбранный объект;

– CopyObjectRequest — создаёт копию объекта в указанном местоположении;

– MoveObjectRequest — перемещает объект в указанное местоположение;

– ChangeObjectAttributesRequest — изменяет атрибуты выбранного объекта (имя, публичность доступа);

– PostObjectResponse — в ответ на произведённое действие возвращает результат в виде выбранного объекта и его обновлённых метаданных;

– GetObjectResponse — по аналогии с предыдущим, возвращает выбранный объект.

Механизм работы большинства из этих функций интуитивно понятен и почти продиктован названием. Swift API предоставляет почти аналогичные по названиям функции, которые реализуют «серверную» составляющую этих запросов, так, например, операция DELETE, с указанием местоположения и названия объекта, удаляет этот объект из контейнера.

Но в Swift не предусмотрены операции перемещения и переименовывания объекта, в привычном понимании.

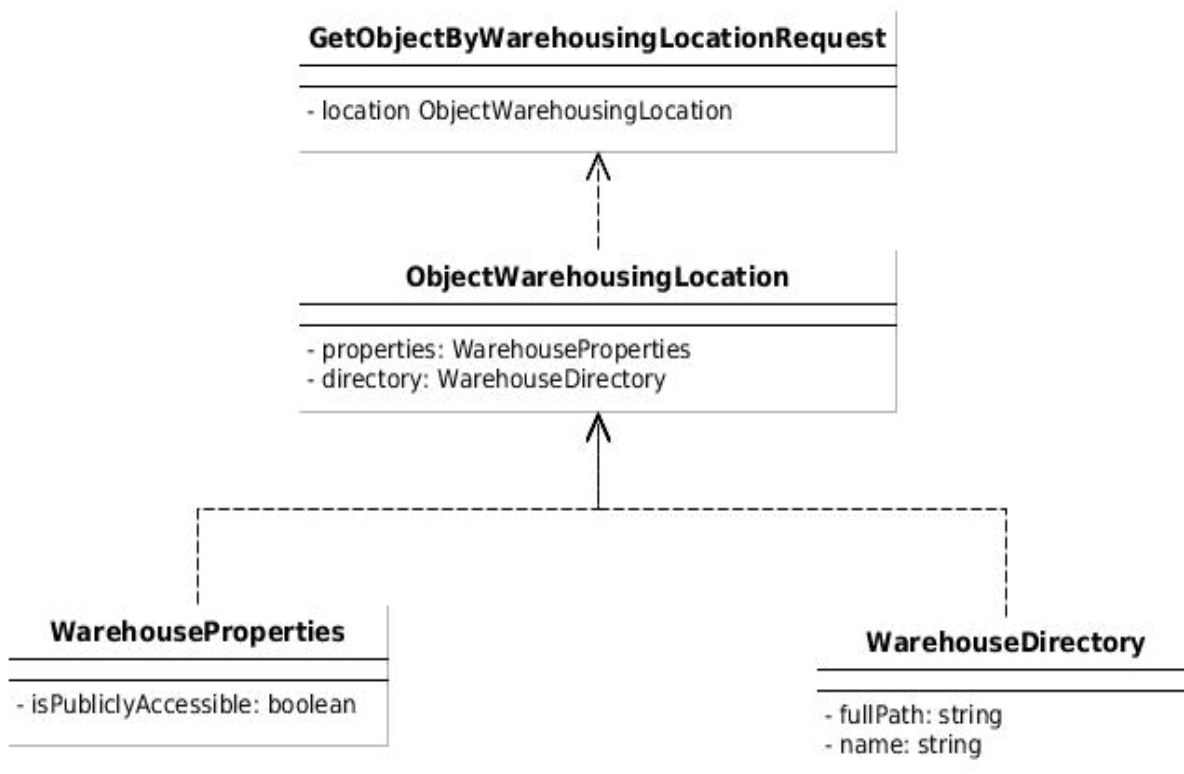


Рис. 2. Диаграмма для класса GetObjectByWarehousingLocationRequest

Такого функционала Swift API не предоставляет. Проблема реализации данных функций решается элементарно. Для перемещения файла необходимо совместить операции копирования объекта и его удаления, таким образом операция перемещения объекта легко выполняется с помощью последовательности тривиальных функций: копирование существующего объекта, размещение его в новом, указанном местоположении, удаление исходного объекта.

Аналогичным образом решается проблема переименования объекта. Для назначения объекту нового имени, необходимо сделать копию содержимого объекта, создать новый объект в этом же контейнере, но с новым именем, удалить исходный объект.

Следующая возникающая проблема связана со способом размещения объектов в Swift. Дело в том, что в отличие от привычных файловых систем, Swift не предусматривает создание директорий — контейнеры не могут быть вложены друг в друга, поэтому директорий, в привычном понимании, в Swift нет. Объектное хранилище Swift не может быть интерпретировано в смысле иерархической структуры, вместо этого оно может быть представлено псевдо-бесконечной картой «ключ-значение», где ключи — это строки, упорядоченные в алфавитном порядке, а значение — есть бинарные объекты.

Взамен реального создания директорий, Swift предлагает механизм имитации директорий, то есть позволяет создавать псевдо-директории [4].

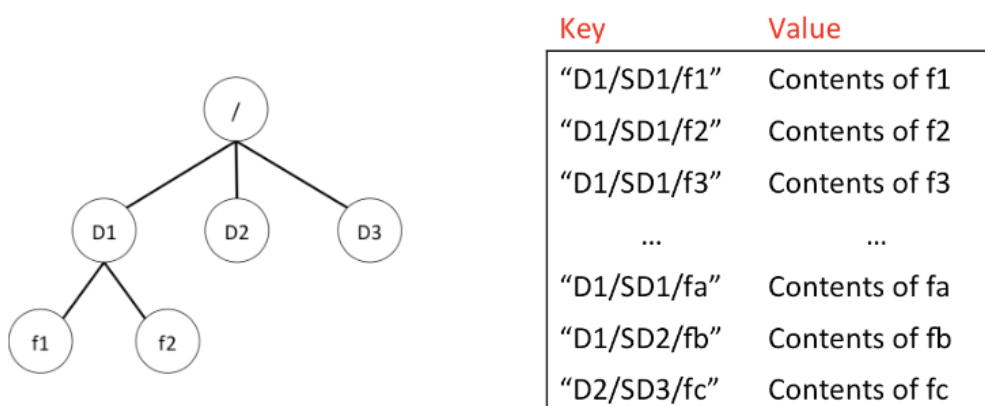


Рис. 3. Представление хранимых объектов в ОС и в Swift

Как было указано ранее, доступ к объекту хранения в Swift осуществляется с помощью трёх компонент: аккаунта, контейнера и объекта. Запрос к хранимому объекту выглядит следующим образом: `https://domain.name/{APIversion}/{accountname}/{containername}/{objectname}`

Для создания псевдо-директории необходимо добавить символ «/» в имя объекта, к примеру, файл `text.txt` можно поместить в каталог `texts` путём добавления соответствующей строки в имя объекта: `texts/text.txt`. Таким образом, доступ к файлу `text.txt` будет осуществляться с помощью запроса `https://domain.name/{API version}/{accountname}/{containername}/texts/text.txt`. Фактическое местоположение объекта при этом — контейнер. Можно создать множество псевдо-директорий и даже «вкладывать» друг в друга, но фактическое место хранения при этом не изменится — все объекты располагаются внутри контейнера.

Для обращения к объектам, располагающимся «внутри» какой-либо директории при формировании запроса используются вспомогательные параметры «delimiter» и «prefix». Для примера, предположим, что в контейнере с названием `container` находятся объекты `folder1/text1.txt` и `folder2/text2.txt`, тогда для обращения к объекту `text1` путь к объекту в запросе должен быть сформирован следующим образом:

```
{API version} / {accountname} /container?prefix=-folder1/&delimiter=/. Соответственно, для получения данных о всех хранящихся в контейнере файлах следует формировать GET-запрос без вспомогательных параметров.
```

Псевдо-директории при этом не являются реальными объектами, Swift ссылается на реальный объект, название которого не заканчивается символом «/». Псевдо-директории не имеют типа содержимого, но каждая псевдо-директория имеет свою SUBDIR запись в JSON-ответе [4].

Литература:

1. Кападиа Амар, Варма Средхар, Раджана Крис Реализация облачного хранилища с Openstack Swift. Packt Publishing, 2014. 140 с.
2. Ротон Дж. Знакомство с OpenStack: Storage-компоненты Swift и Cinder — <https://www.ibm.com/developerworks/ru/library/cl-openstack-swift-cinder/>
3. The Apache Software Foundation. Apache jclouds@:: User Guides — <https://jclouds.apache.org/guides/>
4. The OpenStack Foundation. OpenStack Docs: Current — <http://docs.openstack.org/index.html>

Таким образом, проблемы создания директории и перемещения файлов в директориях, а также изменения атрибутов директории, решаются изменением имени группы объектов. Решение задачи переименовывания было описано ранее. Перемещение и переименовывание целой директории осуществляется путём одновременного применения соответствующих операций к каждому объекту в директории.

Заключение

В ходе работы был реализован программный модуль, осуществляющий задачи хранения файлов в распределённой системе. Данный модуль предоставляет следующий функционал:

1. Создание директории размещения файла;
2. Изменение атрибутов созданной директории;
3. Удаление директории и её содержимого с сервера;
4. Копирование директории вместе с содержимым на сервере;
5. Перемещение директории вместе с содержимым на сервере;
6. Загрузка файла на сервер в выбранную директорию;
7. Скачивание файла с сервера из выбранной директории;
8. Удаление файлов;
9. Изменение атрибутов файла (имя файла, флаг файла — публичный/частный);
10. Перемещение файлов на сервере;
11. Копирование файлов на сервере.

Реализация данного модуля предоставляет минимально необходимый для работы приложения, в данной проблемной области, функционал.

Разработка диспетчера параллельного исполнения задач для формирования изображения в авиатренажёре

Глебов Андрей Павлович, студент

Национальный исследовательский университет «Московский институт электронной техники»

Разработан программный модуль диспетчера, распределяющего задачи, связанные с формированием изображения при помощи технологии OpenGL, по нескольким потокам выполнения. Разработанный диспетчер использует новый алгоритм распределения задач по потокам, основанный на сочетании двух существующих

методов. Для описания зависимостей между задачами по данным используется граф выполнения. Разработанный модуль предназначен для использования в системе визуализации авиатренажёра.

Ключевые слова: диспетчер задач, параллельное программирование, технология OpenGL, система визуализации, авиатренажёр, граф выполнения

В настоящее время активно развивается как военная, так и гражданская авиационная промышленность. В связи с этим, возникает всё большая потребность в высокотехнологичных тренажёрах для обучения пилотов. Авиатренажёры должны позволять обучать пилотов в условиях, приближенных к реальным. Одним из основных критериев реалистичности является качество симуляции внекабинного пространства. Для достижения реалистичного изображения применяются современные средства трёхмерной компьютерной графики. В авиатренажёрах необходимо обеспечивать высокую плавность восприятия движения, для чего требуется отображать изменения обстановки в реальном времени, формируя при этом реалистичное изображение не менее 60-ти раз в секунду.

Каждый кадр формируется в результате выполнения порядка ста различных задач. Некоторые из этих задач можно выполнять одновременно, а другие — нельзя, так как между ними существуют зависимости по данным. Время выполнения задач варьируется от порядка ста микросекунд до нескольких миллисекунд, при этом, в зависимости от условий симулируемого виртуального мира, выполнение одной и той же задачи может занимать разное время.

OpenGL (англ.) — спецификация, определяющая независимый от языка программирования, платформо-независимый программный интерфейс для написания приложений, использующих двумерную и трёхмерную компьютерную графику. Программный интерфейс обеспечивает взаимодействие с графическим процессором для достижения аппаратного ускорения визуализации [1].

Использование в системе визуализации технологии OpenGL накладывает на выполнение некоторых задач дополнительное ограничение: все задачи, использующие функции OpenGL, необходимо запускать в одном потоке выполнения, так как контекст OpenGL «привязан» к программному потоку [1].

Система визуализации авиатренажёра использует высокопроизводительные ПК, укомплектованные многоядерными процессорами. В связи с этим, возможно параллельное выполнение нескольких задач на нескольких программных потоках выполнения. В результате параллелизации исполнения задач может быть достигнуто существенное увеличение производительности системы визуализации. Ввиду выше описанных ограничений на параллельное исполнение некоторых задач, необходимо использование метода распределения задач по потокам выполнения, учитывающего все зависимости по данным между задачами, а также выполняющего задачи форми-

рования изображения (с использованием OpenGL) на одном потоке.

В результате исследования было выявлено, что существует два основных метода распределения задач по потокам: метод функциональной декомпозиции и граф выполнения [2, 3].

В основе метода функциональной декомпозиции лежит деление множества всех задач на несколько видов по сходности их функционала [4]. Такое разделение проводится, так как между смежными по функционалу задачами, зачастую, существует большое количество зависимостей по данным, и их параллельное выполнение может привести к большому числу потенциальных конфликтов. Для каждого вида выделяется поток выполнения, на котором выполняются задачи этого вида.

Необходимо отметить, что между задачами разных видов также существуют зависимости, например, обрабатывать источники освещения и геометрические объекты, принадлежащие городу, можно только после завершения их загрузки, а задачи, использующий OpenGL для формирования изображения, должны ожидать готовности данных, подготавливаемых задачами других видов. Таким образом, для удовлетворения зависимостей некоторых задач необходимо производить блокирующее ожидание (то есть простой) в текущем потоке.

Преимуществом данного метода является то, что разработчику всегда известно, в каком порядке и на каком потоке будут выполняться задачи, что может упростить отладку. К основным недостаткам метода относится снижение КПД использования вычислительных ресурсов при изменении условий виртуального мира и высокая сложность ручного распределения задач по потокам на этапе разработки.

Граф выполнения — специальная структура данных, связывающая задачи и зависимости между ними. Узлы графа представляют задачи и промежуточные данные, а направленные рёбра — зависимости между задачами и данными [5].

Граф выполнения строится разработчиком системы и подаётся на вход диспетчеру задач, который на основе него обеспечивает выполнение всех задач на рабочих потоках. Задачи, готовые к выполнению помещаются в очередь, из которой они передаются на выполнение в свободные рабочие потоки. Особенностью этого метода является требование к равноправности всех задач: диспетчер может запустить любую задачу на любом потоке.

К главным достоинствам метода относится адаптивное распределение нагрузки по вычислительным потокам, отсутствие блокирующего ожидания зависимостей задач и простота добавления задачи для разработчика. Недо-

статком является отсутствие возможности выполнения задач, использующих OpenGL на одном потоке.

Сравнение существующих, а также гибридного (разработанного), методов представлено в таблице 1.

Таблица 1. Сравнение методов распределения задач

Критерий оценки	Функциональная декомпозиция	Граф выполнения	Гибридный метод
Масштабируемость	Нет	Есть	Есть
Равномерное использование ресурсов	Нет	Да	Да
Разработчику известен поток выполнения каждой задачи	Да	Нет	Только для задач формирования изображения
Адаптивный порядок выполнения задач	Нет	Да	Да
Отдельный поток для OpenGL	Есть	Нет	Есть
Блокирующее ожидания результатов других задач	Есть	Нет	Нет

В результате анализа существующих методов решения было установлено, что необходимо разработать новый, гибридный, метод распределения задач, объединяющий преимущества существующих методов и учитывающий особенности работы с технологией OpenGL.

За основу был взят граф выполнения, так как он обладает большим количеством достоинств и имеет ряд преимуществ по производительности и масштабируемости. Основным недостатком графа выполнения является однородность задач, из-за которой невозможно организовать выполнение задач, использующих функции OpenGL, в одном потоке. Для решения данной проблемы был использован метод функциональной декомпозиции.

Выделяется два вида задач: вычислительные задачи и задачи формирования изображения (с использованием графического процессора и технологии OpenGL). При помощи такого разделения, диспетчер может определить вид задачи и выполнить её в верном потоке. Пример разработанного гибридного графа выполнения представлен на рисунке 1.

Диспетчер задач создаёт заданное количество потоков выполнения, затем выделяет один из них для задач формирования изображения. При готовности к выполнению каждой задачи, диспетчер получает её вид и организует её выполнение либо на одном из вычислительных потоков, либо на потоке формирования изображения.

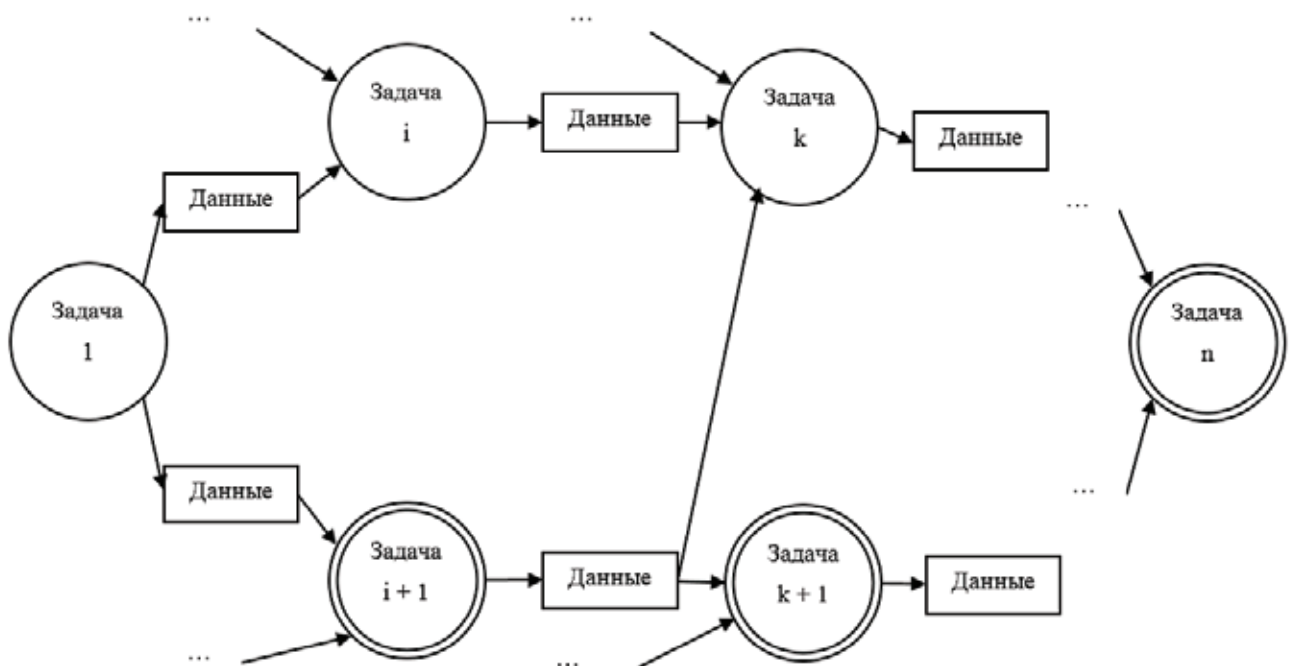


Рис. 1. Пример гибридного графа выполнения. Двойными окружностями обозначены задачи, формирующие изображение при помощи технологии OpenGL

Главной особенностью использования графа выполнения является возможность ожидания всех зависимостей задачи без использования блокирующей синхронизации. За счёт этого, избегается простой рабочих потоков. Для обеспечения адаптивной и равномерной загрузки рабочих потоков применяются очереди задач. Для вычислительных потоков используется общая очередь вычислительных задач, а для потока формирования изображения — отдельная очередь с задачами, использующими OpenGL.

Исполнение задач графа выполнения начинается с входной задачи (так называемой «точки входа»), не имеющей зависимостей. Результатом её выполнения является удовлетворение зависимостей некоторых других задач и их готовность к выполнению. Как только все зависимости задачи оказываются удовлетворёнными, она обрабатывается диспетчером, который получает её вид и, в зависимости от него, помещает её либо в очередь вычислительных задач, либо в очередь задач формирования изображения. Исполнение графа завершается после выполнения последней задачи, от которой не зависит ни одна другая задача («точка выхода»).

Непосредственное выполнение задач производится в функции рабочего потока. Каждый поток находится в цикле, из которого он выходит только после выставления флага конца работы диспетчера. В теле цикла происходит извлечение задачи из очереди, соответствующей типу потока, и её выполнение. Если в результате выполнения очередной задачи оказываются полностью удовлетворёнными все зависимости некоторой другой задачи, то готовая задача помещается в очередь, соответствующую её типу.

Отдельный интерес представляет выбор реализации очередей задач. К обоим используемым очередям производится одновременное обращение из нескольких по-

токов. В связи с этим, необходимо наличие механизма синхронизации доступа к ним [6]. Самый простой и распространённый вариант синхронизации — использование критических секций, например, мьютексов. Такая реализация имеет серьёзные недостатки по производительности, так как блокирует выполнение рабочего потока до получения доступа в критическую секцию. Для избежания блокировки можно использовать неблокирующий (lock-free) алгоритм синхронизации. Существует множество реализаций потокобезопасных очередей, рассчитанных на различное количество одновременно обращающихся потоков, использующих неблокирующую синхронизацию, чаще всего, на основе атомарных переменных.

В результате использования описанного метода распределения задач по потокам достигается:

- масштабируемость системы визуализации при изменении количества рабочих потоков;
- адаптивное распределение вычислительной нагрузки по потокам в зависимости от конкретных условий симулируемого мира;
- удовлетворение требований стандарта OpenGL;
- удобство добавления задачи для разработчика;
- отсутствие блокирующей синхронизации в рабочих потоках.

На основе гибридного метода была разработана реализация диспетчера задач системы визуализации. Результаты тестирования его производительности при помощи профилировочных инструментов, таких как Intel VTune Amplifier [7], подтвердили все преимущества данного метода. Так как система визуализации авиатренажёра во многом схожа с другими системами симуляции виртуального мира, разработанный метод может быть применён в любых системах, использующих технологию OpenGL, таких как компьютерные игры и системы дополненной реальности.

Литература:

1. OpenGL API Documentation [Электронный ресурс]. URL: <https://www.opengl.org/documentation/> (дата обращения: 18.03.2016).
2. Gregory, J. Game Engine Architecture [Текст] / Jason Gregory. — 2nd ed. — Boca Raton, FL: CRC Press, 2014. — 1052 p.
3. Andrews, J. Designing the Framework of a Parallel Game Engine [Электронный ресурс] / Jeffrey Andrews. Intel Developer Zone, 2009–2015. URL: <https://software.intel.com/en-us/articles/designing-the-framework-of-a-parallel-game-engine> (дата обращения: 15.02.2016).
4. Богачёв, К. Ю. Основы параллельного программирования [Текст] / К. Ю. Богачёв. — М.: БИНОМ. Лаборатория знаний, 2003. — 342 с.
5. Parallelizing Data Flow and Dependence Graphs [Электронный ресурс]. Intel Developer Zone. URL: <https://software.intel.com/en-us/node/517340> (дата обращения: 20.03.2016).
6. Herlihy, M., Shavit, N. The Art of Multiprocessor Programming [Текст] / Maurice Herlihy, Nir Shavit. — Revised reprint, 1st ed. — San Francisco, CA: Morgan Kaufmann, 2012. — 536 p.
7. Intel VTune Amplifier Performance Profiler [Электронный ресурс]. URL: <https://software.intel.com/en-us/intel-vtune-amplifier-xe> (дата обращения: 20.05.2016).

MapReduce и метод доступа к хранилищу MRIJ on RCFile

Григорьев Юрий Александрович, доктор технических наук, профессор;

Качалкин Кирилл Игоревич, студент

Московский государственный технический университет имени Н.Э. Баумана

Сегодня большая часть информации хранится в реляционных базах данных. Это связано с тем, что такая модель данных проста, понятна и удобна для физической реализации на ЭВМ, а также имеет оптимизаторы выполнения транзакций и запросов к базе данных, обеспечивающих их «дробление» на более мелкие задания и параллельную реализацию этих работ на многопроцессорных или многомашинных комплексах. Благодаря этим факторам пользователи до сих пор выбирают и используют именно реляционные базы данных.

Но такая модель хранения данных отчасти устарела и подходит не для всех проектов. С тех пор, как придумали строчные реляционные базы данных, технические характеристики устройств резко выросли: мощности процессоров увеличились в 5000–10000 раз, время передачи данных с диска в память уменьшилось в 100 раз, время поиска данных на диске — в 10 раз. Различие в приросте мощностей обрабатывающих ресурсов значительно повлияло на рабочие нагрузки СУБД. Наблюдался также дисбаланс между ростом емкости диска и скоростью передачи данных:

1. пропускная способность передачи информации по отношению к общему объему доступной информации уменьшилась на два порядка;

2. отношение скорости доступа к последовательно расположенным на диске данным и при случайной выборке выросло на порядок. Стало понятно, что в СУБД нужно устранить случайный доступ к данным, а также уменьшить нагрузку на канал их передачи.

Часть этих проблем была решена с приходом колоночных баз данных. Но основным минусом и строчных и колоночных СУБД является поддержка реляционной модели, которая предполагает хранение данных в виде та-

блиц и наличие схемы базы данных. Из этого вытекают трудно решаемые задачи:

1. Потеря соответствия.
2. Масштабируемость.
3. Реляционные базы данных не справляются с обработкой большого объема неструктурированных данных (текстов).

Для решения этих проблем создаются новые варианты хранения и обработки данных, получившие название «базы данных NoSQL». Исходя из потребностей системы, выбирают одну из четырех типов хранилищ: «ключ-значение» (key-value store), документно-ориентированные (document store), хранилища семейств колонок (column database), графовые базы данных (graph database) [1]. В данной статье рассматривается только тип «ключ-значение». Это простейшее хранилище, которое использует ключ для доступа к значению. Именно такой тип используется для создания специализированных файловых систем, а также систем, ориентированных на масштабируемость.

MapReduce

Базы данных, построенные на парадигме распределенных хранилищ «ключ-значение», обеспечивают доступ к неструктурированным данным посредством прямого чтения записей или выполнения заданий MapReduce [2].

MapReduce — это алгоритм, доступа к неструктурированным данным. Первым шагом этого алгоритма является выполнение функции Map ко всем элементам исходной коллекции. Результатом этого шага будет либо пустота, либо экземпляр «ключ-значение»:

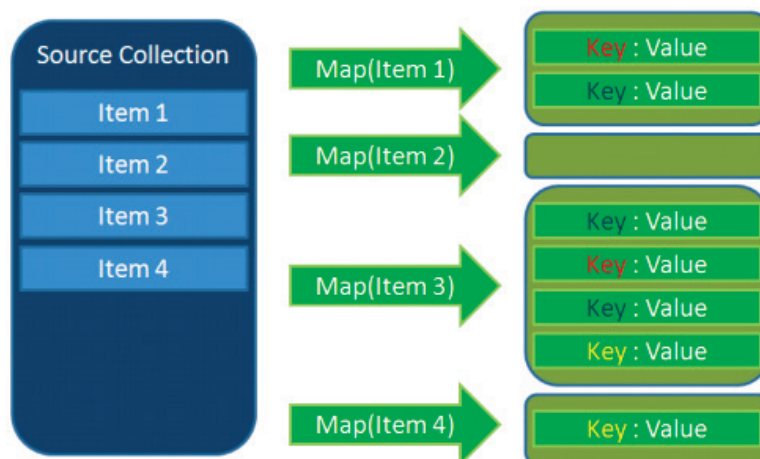


Рис. 1. Первый шаг алгоритма MapReduce

На втором шаге происходит сортировка всех экземпляров и создание новых, где все значения сгруппированы по ключу:

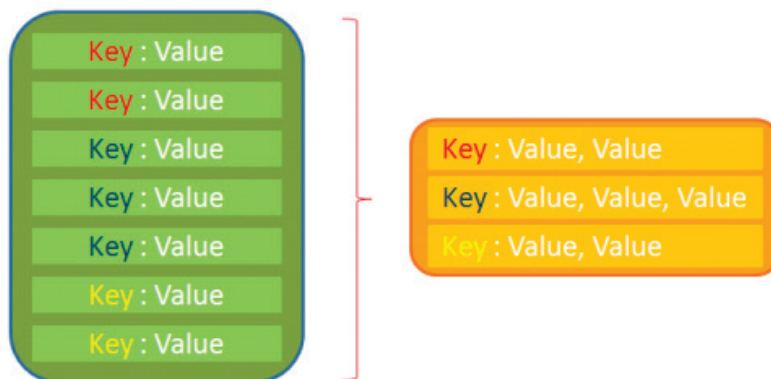


Рис. 2. Второй шаг алгоритма MapReduce

Заключительным шагом выполнится функция Reduce, которая вернет новый экземпляр объекта, который будет включен в результирующую коллекцию:

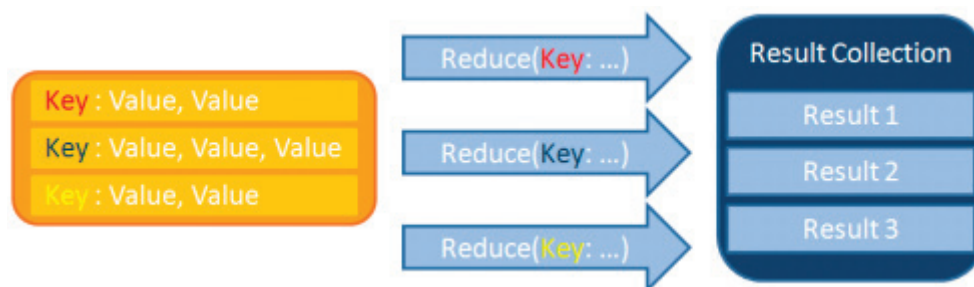


Рис. 3. Третий шаг алгоритма MapReduce

Метод MRIJ on RCFile

Существуют различные методы доступа к хранилищу данных по технологии MapReduce. В [3] рассматриваются способы реализации соединения таблиц измерений и фактов, которые связаны по схеме «звезда». Они позволяют избежать перемещений записей таблицы фактов. Запрос для схемы с двумя измерениями (D1 и D2) и одной таблицей фактов (F) можно представить так:

```
SELECT D1.d11, D2.d21, F.m1
FROM D1 JOIN F ON (D1.d10 = F.fk1) JOIN D2 ON
(D2.d20 = F.fk2)
WHERE CD1 AND CF1,
```

где CD1 и CF1 — некоторые дополнительные условия, накладываемые на измерения и факты.

Рассмотрим вариант соединения по методу MRIJ on RCFile.

Таблица (измерений или фактов) разбивается на несколько горизонтальных разделов (row group). В одном блоке HDFS таблицы может храниться несколько раз-

делов. Внутри каждого раздела данные хранятся по столбцам (в сжатом виде). Данные можно читать по отдельным столбцам (или группам столбцов) в блоке. При этом считанный столбец раздела разжимается только в том случае, если он действительно используется. Блоки таблицы распределены по узлам произвольным образом.

Первый шаг выполнения запроса.

Map1:

1. На узле читаются записи таблицы измерения; выделяются группы столбцов, удовлетворяющие условию поиска, в файловую систему MapReduce помещаются записи:

$dkv_i = \langle \text{key}, \text{value} \rangle,$

где key — первичный ключ таблицы измерения, value — список значений других колонок данного измерения, которые необходимо поместить в результирующую таблицу.

Далее следует повторить пункт 1 для всех измерений, которые участвуют в запросе и блоки которых хранятся на данном узле.

Reduce1: — отсутствует.

Второй шаг выполнения запроса.

Map2:

1. На узле читаются полученные записи измерения dkv_i и для них строится хеш-индекс в ОП по ключу *key*.

Пункт 1 следует повторить для всех поступивших измерений.

2. Читается столбец первого внешнего ключа таблицы фактов, участвующего в запросе, и проверяется наличие значений этого ключа в соответствующем хеш-индексе; при успешном сравнении сохраняются записи в ОП узла (т.е. локально):

$fkv_i = \langle \text{позиция}, \text{value} \rangle$,

где «позиция» — это номер строки таблицы фактов (нумерация сквозная по всем строкам таблицы фактов); *value* — список значений из соответствующей записи dkv_i .

3. Шаг 3 выполняется по числу измерений, участвующих в запросе (минус 1, так как для первого измерения выполнен шаг 2), т.е. $i=2, \dots, n$. По позициям промежуточной таблицы fkv_{i-1} читаются кортежи столбца *i*-го внешнего ключа таблицы фактов и проверяется наличие значений этого ключа в соответствующем хеш-индексе; при успешном сравнении сохраняются записи в ОП узла (т.е. локально):

$fkv_i = \langle \text{позиция}, \text{value} \rangle$,

где «позиция» — номер строки таблицы фактов;

$\text{value} = \text{value} \cup \text{value}_i$, *value* справа от знака присваивания — это значение из записи предыдущей промежуточной таблицы fkv_{i-1} , value_i — список значений из соответствующей записи dkv_i .

Предыдущая промежуточная таблица fkv_{i-1} удаляется. Пример, иллюстрирующий выполнение шага:

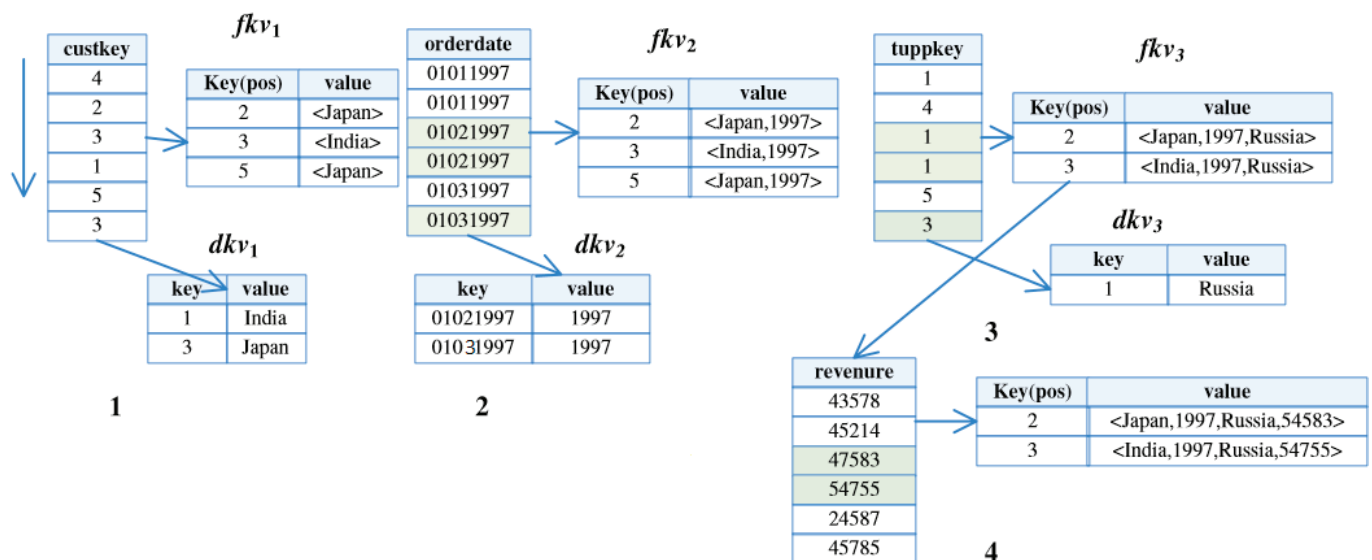


Рис. 4. Пример последовательного соединения таблиц измерений и таблицы фактов

4. По позициям последней промежуточной таблицы fkv_n читаются значения столбцов фактов (они хранятся в том же блоке, что столбцы внешних ключей). На диске сохраняются записи результирующей таблицы:

$\langle \text{позиция}, (\text{value}, vm_1, \dots, vm_k) \rangle$,

где *value* — значение в записи промежуточной таблицы fkv_n (список значений атрибутов из таблиц измерений); vm_i — значение *i*-го факта.

Reduce2: нет.

Литература:

1. NoSQL: [Электронный ресурс]. [http://ru.wikipedia.org/wiki/NoSQL] Проверено 30.05.2016.
2. Редмон, Э., Уилсон Д. Р. Семь баз данных за семь недель. Введение в современные базы данных и идеологию NoSQL. — М.: ДМК Пресс, 2013. — 384 с.
3. Zhou, G., Zhu Y., Wang, G. Cache Conscious Star-Join in MapReduce Environments. Cloud-I '13 Proceedings of the and International Workshop on Cloud Intelligence, August 26 2013.

Заключение

Таким образом мы получили общую картину о модели распределённых вычислений MapReduce и разобрались в методе доступа к хранилищу данных MRIJ on RCFile. Реализацию данного метода рассмотрим в следующей статье.

Hadoop и метод доступа к хранилищу MRIJ

Григорьев Юрий Александрович, доктор технических наук, профессор;

Эрдынеев Майдари Цыденович, студент

Московский государственный технический университет имени Н.Э. Баумана

ВigData, MapReduce, облачные вычисления, NoSQL. Все эти понятия стали популярными в последние годы. И все они связаны с распределенной обработкой больших объемов данных. Hadoop — одна из самых популярных open-source систем для обработки больших объемов данных. Необходимость в таких системах растет с каждым годом — все больше компаний сталкиваются с проблемой растущего объема данных.

Facebook, Twitter, Yahoo!, Bing, Mail.ru — это далеко не полный список компаний, которые используют Hadoop. Многие из них, при этом, активно участвуют в его развитии. И это неслучайно, т.к. именно большие интернет-компании первыми столкнулись с проблемой обработки больших объемов данных: как их надежно хранить, как обрабатывать, как получать быстрый доступ на их изменение. Сейчас Hadoop используется не только в интернет-компаниях, но и во многих других сферах, где возникает проблема с объемом данных (экономика, астрономия, биология, физика и т.д.). В своей статье я хочу рассказать про дистрибутивы, подход MapReduce, а также MRIJ — метод соединения таблиц в рамках парадигмы MapReduce.

Обзор дистрибутивов

Hadoop — это проект фонда Apache Software Foundation, свободно распространяемый набор утилит, библиотек и фреймворк для разработки и выполнения распределённых программ, работающих на кластерах из сотен и тысяч узлов. Т.к. Hadoop — это не монолитный продукт, а экосистема, состоящая из множества компонентов, то в данной сфере существуют организации, которые свободно разрабатывают поддерживают уже собранные дистрибутивы Hadoop. Вот обзор основных вендоров Hadoop:

1. **Cloudera**. Ключевой продукт — CDH (Cloudera Distribution including Apache Hadoop) — связка наиболее популярных инструментов из инфраструктуры Hadoop под управлением Cloudera Manager. Менеджер берёт на себя ответственность за развёртывание кластера, установку всех компонентов и их дальнейший мониторинг. Кроме CDH компания развивает и другие свои продукты, например, Impala (об этом ниже). Отличительной чертой Cloudera также является стремление первыми предоставлять на рынке новые фишки, пусть даже и в ущерб стабильности. Ну и да, создатель Hadoop — Doug Cutting — работает в Cloudera.

2. **Hortonworks**. Так же, как и Cloudera, они предоставляют единое решение в виде HDP (Hortonworks

Data Platform). Их отличительной чертой является то, что вместо разработки собственных продуктов они больше вкладывают в развитие продуктов Apache. Например, вместо Cloudera Manager они используют Apache Ambari, вместо Impala — дальше развивают Apache Hive. Мой личный опыт с этим дистрибутивом сводится к паре тестов на виртуальной машине, но по ощущениям HDP выглядит стабильней, чем CDH.

3. **MapR**. В отличие от двух предыдущих компаний, основным источником доходов для которых, судя по всему, является консалтинг и партнёрские программы, MapR занимается непосредственно продажей своих наработок. Из плюсов: много оптимизаций, партнёрская программа с Amazon. Из минусов: бесплатная версия (МЗ) имеет урезанный функционал.

Архитектура Hadoop. Обзор

По состоянию на 2016 год, проект Hadoop состоит из основных четырех модулей:

- Hadoop Common (набор инфраструктурных программных библиотек и утилит, используемых для других модулей и родственных проектов)
- HDFS (Распределенная файловая система)
- YARN (система для планирования заданий и управления кластером)
- Hadoop MapReduce (платформа программирования и выполнения распределённых MapReduce-вычислений).

Hadoop Common

В Hadoop Common входят библиотеки управления файловыми системами, поддерживаемыми Hadoop и сценарии создания необходимой инфраструктуры и управления распределённой обработкой, для удобства выполнения которых создан специализированный упрощённый интерпретатор командной строки (FS shell, filesystem shell), запускаемый из оболочки операционной системы командой вида: `hdfs dfs-command URI`, где `command` — команда интерпретатора, а `URI` — список ресурсов с префиксами, указывающими тип поддерживаемой файловой системы, например `hdfs://example.com/file1` или `file:///tmp/local/file2`.

Большая часть команд интерпретатора реализована по аналогии с соответствующими командами Unix (таковы, например, `cat`, `chmod`, `chown`, `chgrp`, `cp`, `du`, `ls`, `mkdir`, `mv`, `rm`, `tail`, притом, поддерживаются некоторые ключи аналогичных Unix-команд, например ключ рекурсивности `-R` для `chmod`, `chown`, `chgrp`).

Есть команды, специфические для Hadoop (например, `count` подсчитывает количество каталогов, файлов и байтов по заданному пути, `hadoop fs -ls` очищает корзину, а `setrep` модифицирует коэффициент репликации для заданного ресурса).

Архитектура Hadoop. HDFS

HDFS — основа Hadoop, это файловая система для хранения больших данных. HDFS основана на другой большой системе — GFS (Google file system), она работает на кластере из серверов, на каждом из которых может быть своя файловая система (Ext3, Ext4), при этом для пользователя она выглядит как «один большой диск», т.к. скрывает внутри себя физическое управление файлами. HDFS обладает следующими свойствами:

- Хорошо подходит для хранения больших файлов (от 100 мб).

- Fault Tolerance (данные не теряются, если диски выходят из строя).

Однако из-за особенностей архитектуры HDFS не подходит для:

- хранения большого количества маленьких файлов.
- постоянной записи на файловую систему, т.к. одновременно в файл может писать только 1 процесс.

Все файлы HDFS поделены на блоки определенного размера, и каждый блок может быть размещён на нескольких узлах, размер блока и коэффициент репликации (количество узлов, на которых должен быть размещён каждый блок) определяются в настройках на уровне файла). Благодаря репликации обеспечивается устойчивость распределённой системы к отказам отдельных узлов. Организация файлов в пространстве имён — традиционная иерархическая: есть корневой каталог, поддерживается вложение каталогов, в одном каталоге могут располагаться и файлы, и другие каталоги.

Демоны HDFS

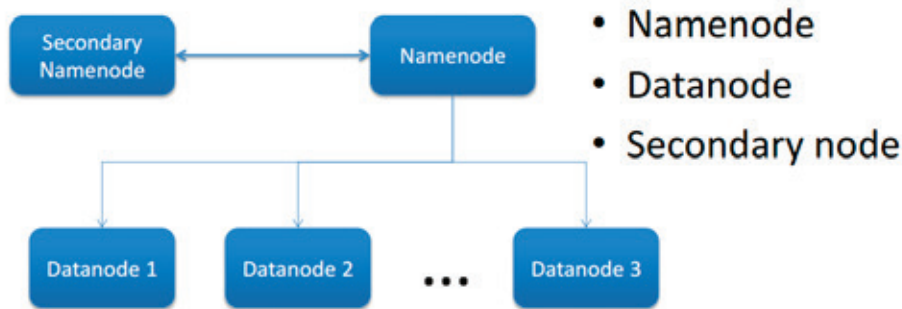


Рис. 1

Развёртывание экземпляра HDFS предусматривает наличие центрального узла имён (англ. name node), хранящего метаданные файловой системы и метainформацию о распределении блоков, и серии узлов данных (англ. data node), непосредственно хранящих блоки файлов. Узел имён отвечает за обработку операций уровня файлов и каталогов — открытие и закрытие файлов, манипуляция с каталогами, узлы данных непосредственно отрабатывают операции по записи и чтению данных. Узел имён и узлы данных снабжаются веб-серверами, отображающими текущий статус узлов и позволяющими просматривать содержимое файловой системы. Также для помощи в обработке существует Secondary Node, в его обязанности входит обновление fsimage (файл, который используется для восстановления в случае падения кластера).

Архитектура Hadoop. MapReduce

Hadoop MapReduce — фреймворк для программирования распределённых вычислений в рамках Hadoop. Основан на двух операциях: Map и Reduce.

На входе файл бьётся на «сплиты» определенного размера (который равен обычно размеру блока в hdfs). Далее каждому «воркеру» передается один «сплит». Затем происходит фаза Map. На каждом воркере стоит обработчик, который обеспечивает обработку входных данных, причем промежуточные результаты пишутся на локальный диск. На выходе «Маппера» получается пара Ключ — Значение, которая затем передается по сети «воркеру» с Reduce обработчиком. Стоит заметить, что данные с одним и тем же ключом попадают на один «воркер» и они начинают обрабатываться только тогда, когда отработают все «воркеры» с операцией Map. Далее начинается фаза Reduce, фаза свертки результата. Каждый «воркер» с Reduce обработчиком пишет в один файл, который сохраняется в HDFS.

Архитектура Hadoop. YARN

YARN (Yet Another Resource Negotiator) — модуль, который появился в 2013 году, он отвечает за управление кластерами и планирование заданий. В ранних версиях

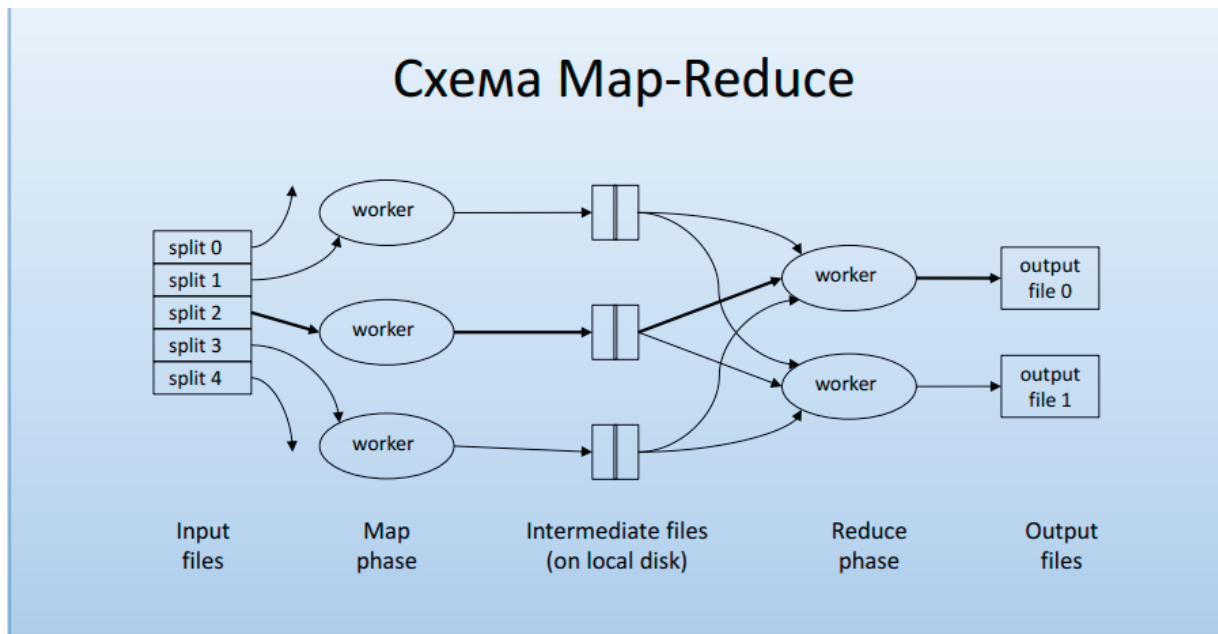


Рис. 2

эта функция была интегрирована в модуль MapReduce, где была реализована как компонент JobTracker. Сейчас же в YARN функционирует самостоятельный демон — ResourceManager, который абстрагирует ресурсы кластера

и управляет их предоставлением приложениям распределенной разработки. YARN обеспечивает возможность параллельного выполнения нескольких задач в рамках кластера и их изоляцию.

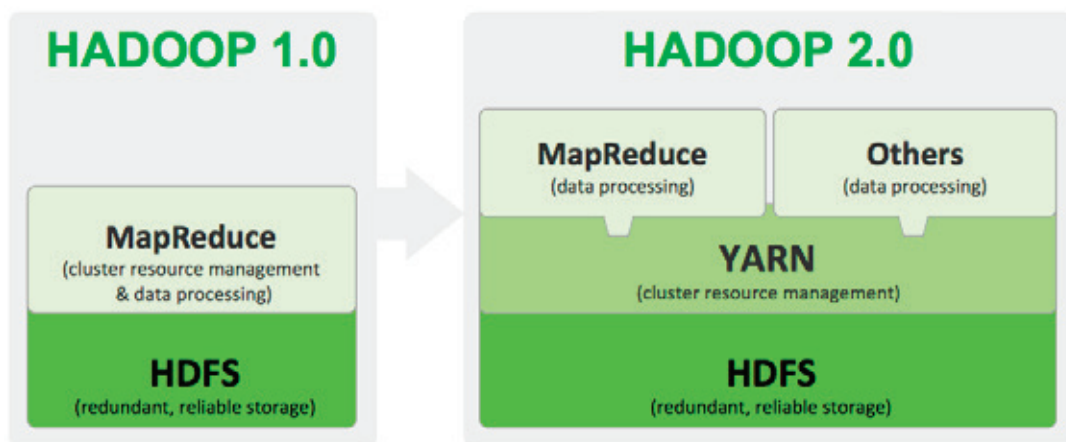


Рис. 3

Для этого разработчику необходимо реализовать класс управления приложением ApplicationMaster, который отвечает за координацию заданий в рамках ресурсов, предоставленных планировщиком. Планировщик же отвечает за создание экземпляров класса управления приложением и взаимодействия с ним через соответствующий сетевой протокол.

Метод MRJ

MRJ — метод доступа к данным, которые организованы в виде таблиц измерений и фактов, связанных по схеме «Звезда».

Типовой запрос Star Join можно представить следующим образом:

```
SELECT D1.d11, D2.d21, F.m1
FROM D1 JOIN F ON (D1.d10 = F.fk1) JOIN D2 ON
(D2.d20 = F.fk2)
WHERE CD1 AND CF1,
```

где Di — таблица измерений, а F — таблица фактов. Обычно при такой схеме запрос вида соединения вида F x D1 x D2 разделяется на два запроса F x D1 и F x D2. Это производится на фазе Map, на фазе Reduce эти два варианта соединяются. Однако вычисление промежуточных результатов требует больших мощностей и высокую пропускную способность.

Описание метода MRIJ

На первом шаге Первого Map-Reduce задания производится фильтрация таблиц, указанных в запросе Select (Для каждой таблицы). Результатом этого будут записи вида

$DKv_i = \langle \text{key}, \text{value} \rangle$, где key — первичный ключ таблицы измерения, value — список значений других колонок данного измерения, которые необходимо поместить в результирующую таблицу.

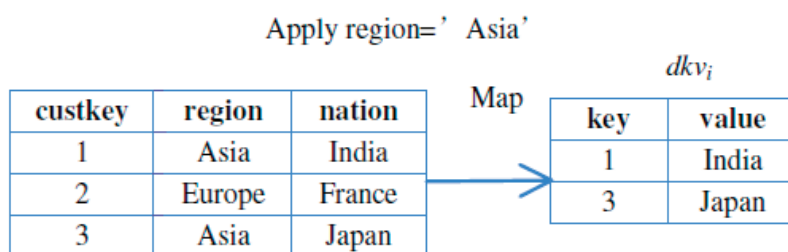


Рис. 4

На втором шаге записи переносятся на узлы с другим обработчиком. На втором узле происходит следующее:

1. Для всех входящих измерений строится хэш-индекс в памяти по их ключу
2. Далее из таблицы факто читается колонка внешнего ключа и ищется в индекса, при успехе на выход отдается запись следующего вида:
3. $fkv_i = \langle \text{позиция}, \text{value} \rangle$, где «позиция» — это номер строки таблицы фактов (нумерация сквозная по всем строкам таблицы фактов); value — список значений из соответствующей записи dkv_i

4. Далее читается очередная колонка из таблицы фактов и проверяется условие CF1 и при успехе на выход подается запись вида:

5. $vm_j = \langle \text{позиция}, \text{value} \rangle$, где «позиция» — это номер строки таблицы фактов (нумерация сквозная по всем строкам таблицы фактов), value — значение факта
6. Затем выполняется фаза Reduce: происходит группировка по позиции, затем проверяется число элементов, оно должно составлять $n+k$ (n и k — это соответственно число измерений и фактов, участвующих в запросе) и проверяется удовлетворение условию запроса. В случае успеха на выход помещается запись вида: $\langle \text{позиция}, (fkv_1.\text{value}, \dots, fkv_n.\text{value}, vm_1.\text{value}, \dots, vm_k.\text{value}) \rangle$

Литература:

1. Guoliang Zhou, Yongli Zhu, Guilan Wang Cache Conscious Star-Join in MapReduce Environments. — The Higher Education Scientific Research Project of Hebei Province — с. 7.
2. Том Уайт Hadoop: Подробное руководство. — СПб.: Питер, 2013. — 672 с.: ил. — (Серия «Бестселлеры O’Reilly»)
3. Donald Miner, Adam Shook MapReduce Design Patterns — O’Reilly Media, Inc. 2013 — с. 233.
4. Lars George HBase: The Definitive Guide — O’Reilly Media, Inc. 2011 — с. 524
5. Брюс Эккель. Философия Java = Thinking in Java. — 3-е изд. — СПб.: Питер, 2003. — 976 с. — ISBN 5-88782-105-1.

Анализ тональности отзывов пользователей в метаобласти фильмов

Журавлева Любовь Викторовна, студент;
 Стригулин Кирилл Анатольевич, студент
 Санкт-Петербургский государственный университет

В статье рассмотрена задача анализа тональности текста на примере отзывов о фильмах. Рассмотрена проблематика подхода, основанного на словарях, создан свой собственный словарь сентиментов и для него проведены исследования на корпусе данных.

Ключевые слова: словарь сентиментов, тональность

Анализ тональности текста — это сложный процесс, касающийся выделения полезной субъективной информации из текста. Огромное множество пользовательского контента в интернете появляется с каждым днем. Миллионы пользователей ежедневно высказывают свое мнение о продуктах и услугах в блогах, социальных сетях и других информационных ресурсах. Предоставление надежного извлечения мнения из неструктурированного текста имеет важное значение для коммерческих организаций. С помощью предоставленных данных компании смогут узнать важное для них мнение покупателей, найти невидимые для их глаза недостатки и повысить свой уровень продаж.

Анализ тональности применяется на множестве текстовых документов, содержащих в себе эмоции и оценки определенных объектов, к примеру людей, событий, тем (например, отзывы о фильмах, книгах, продуктах). Анализ тональности предполагает идентификацию сентимента в документе, и в последствии определения его положительной/отрицательной полярности.

Как правило, анализ тональности применяется на корпусах текстов, содержащих отзывы. Однако, анализ тональности может быть в том числе применен к новостным статьям [1] или блогам и социальным сетям. Сентимент анализ также применяют чтобы извлечь общественное мнение о различных темах в пределах от фондовых рынков [2] до политических споров [3].

Существуют два основных подхода к задаче автоматического извлечения тональности — подход, основанный на использовании словарей сентиментов и подход, основанный на машинном обучении. В данной статье будет рассматриваться подход, основанный на словарях.

Создание словарей сентиментов очень сложный процесс. Для того чтобы создать словарь сентиментов необходимо вручную обработать большое количество текста, чтобы понять, какая лексика используется в рассматриваемой области.

В качестве основы для создания словаря сентиментов были взяты 5000 ключевых слов из корпуса о фильмах, книгах и фотоаппаратах [4] без разделения по тональности. Каждое слово было вручную определено на положительную/отрицательную тональность. Те слова, которые невозможно было однозначно отнести к положительной или отрицательной тональности исключались. Таким образом, специально для мета-области фильмов было получено 1926 слов с разделением по тональности. Из них 1002 слова было с положительной тональностью и 924 слова с отрицательной тональностью.

Тестовая коллекция

Отбор данных для тестирования очень важен. Без тестовых данных невозможно четко и точно определить работоспособность полученной модели. Для тестирования полученного словаря был использован корпус отзывов

о фильмах, предоставленный НП РОМИП [5] в виде xml-кода.

Коллекция была представлена в виде xml-кода, каждый отзыв содержит оценку и текст мнения. Корпус данных состоит из 15718 таких отрывков. После предварительной обработки данных было определено, что данный корпус состоит из 1502030 слов.

В качестве положительных отзывов было принято считать те тексты, которые имели оценку выше пяти баллов, а отрицательными те тексты, которые имели оценку ниже шести баллов.

Так как исходные данные для тестирования были представлены в виде необработанного текста, была проведена предварительная обработка для приведения документов к нормализованному виду.

В своей программе для первоначальной обработки текста использовалась библиотека Beautiful Soup.

В рассматриваемом случае задача состояла в том, чтобы выделить из корпуса данных информацию для двух массивов — массива оценок и массива непосредственных отзывов.

Так как полученные тексты представляли из себя набор из нескольких предложений, было принято решение избавиться от лишних знаков и выделить отдельные слова — для этого текст разбивался на слова регулярным выражением «`^ [\s\W] *| [^\w] \s (?= [\W\s] |$) (?u)`». Все полученные слова были переведены в нижний регистр.

Алгоритм для словарного подхода

В качестве определения тональности для словарного подхода, было принято использовать метод, описанный Peter D. Turney в 2002 году [6]. Идея заключалась в следующем: каждое слово текста рассматривалось на наличие в словаре, как слово, несущее положительную или отрицательную тональность. Если слово встречалось в словаре, как слово с положительным весом, то счетчик для слов, несущих положительную тональность увеличивался. Аналогично счетчик для слов несущих отрицательную тональность, увеличивался, если слово имеет отрицательный вес.

Тональность текста определялось большим количеством того или иного счетчика.

Методы оценки построенного алгоритма

Для того чтобы понять, насколько хорошо построенный алгоритм работает с данными, необходима численная метрика его качества.

Точность и полнота

Точность (precision) — это доля документов, которые действительно принадлежат классу относительно всех документов, который классификатор отнес к этому классу.

Полнота (recall) — это доля найденных системой документов, принадлежащих классу относительно всех документов рассматриваемого класса в тестовой выборке.

Данные значения можно рассчитать на основе таблицы контингентности

Таблица 1. Таблица контингентности

Категория i		Экспертная оценка	
		положительная	отрицательная
Оценка системы	положительная	TP	FP
	отрицательная	FN	TN

Где:

TP — истинно-положительное решение

FP — ложно-положительное решение

FN — ложно-отрицательное решение

TN — истинно-отрицательное решение

Тогда полнота и точность вычисляются следующим образом:

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

Высокая точность и полнота означают качественное построение модели.

F-мера

F-score (F-мера) это метрика, которая объединяет информацию о точности и полноте алгоритма. F-score пред-

ставляет собой гармоническое среднее между точностью и полнотой. Данная метрика стремится к нулю, если полнота и точность стремятся к нулю

$$F = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

Результаты

Для точного определения работы подхода, основанного на словарях, было принято решение разделить исходный корпус данных на 30 выборок, состоящих из 500 отзывов, собранных случайным образом.

Используя метод, описанный Peter D. Turney для словаря сентиментов, полученного ручной обработкой по тональности из словаря ключевых слов, были получены следующие результаты:

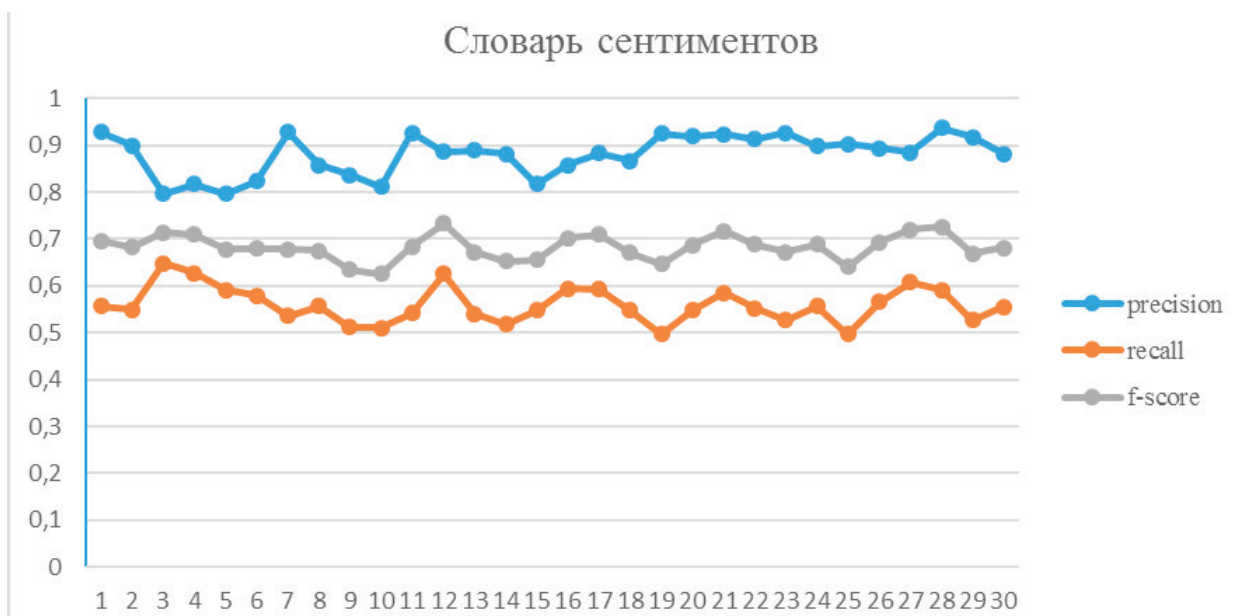


Рис. 1. Результаты словаря сентиментов

Таким образом, представив данные в виде диаграммы можно заметить высокую точность. Недостаточно высокую полноту можно объяснить малым словарным запасом словаря, который состоит из 2000 слов.

Рассмотрим, теперь отдельно оценки определения позитивных и негативных отзывов. Результаты получились следующими:

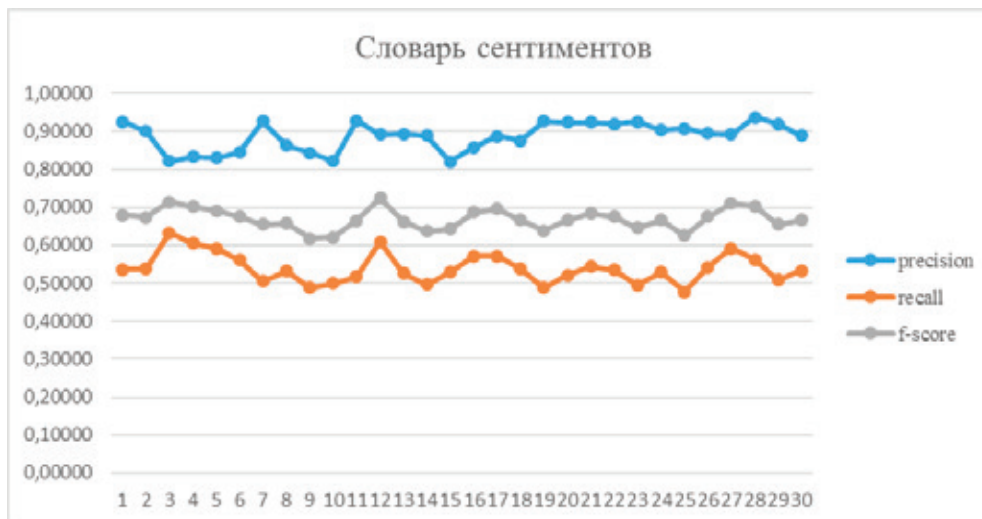


Рис. 2. Результаты оценки определения положительных отзывов для словаря сентиментов

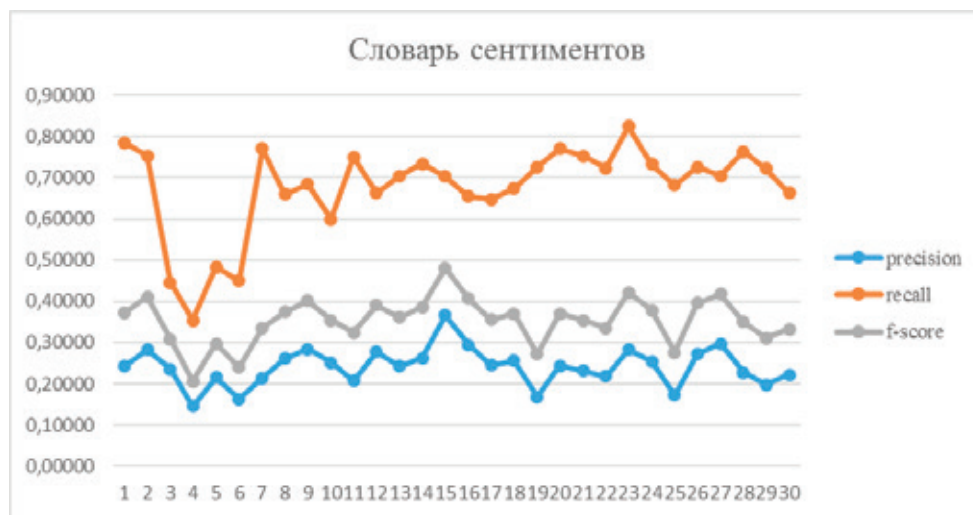


Рис. 3. Результаты оценки определения отрицательных отзывов для словаря сентиментов

Результаты оценки словаря сентиментов показали более высокую точность определения положительных отзывов, чем отрицательных, и более высокий общий результат.

Выводы

Целью данной статьи являлось исследование и разработка метода анализа тональности отзывов пользова-

телей. Для исследования задач классификации был выбран подход, основанный на словаре, была рассмотрена проблема его построения и различные методы решения данной задачи. В итоге был получен собственный словарь сентиментов, который показал достаточно высокие результаты определения точности. Исследование показало, что определение отрицательных отзывов более трудная задача, чем определение положительных отзывов.

Литература:

1. T. Xu, Q. Peng, and Y. Cheng, Knowledge-Based Systems 35, 279 (2012).
2. M. Hagenau, M. Liebmann, and D. Neumann, Decision Support Systems 55, 685 (2013).
3. I. Maks and P. Vossen, Decision Support Systems 53, 680 (2012).
4. Chetviorkin, I. I., Loukachevitch N. V. Extraction of Russian Sentiment Lexicon for Product Meta-Domain // In Proceedings of COLING 2012: Technical Papers, pages 593–610 (2012).
5. <http://romip.ru/>
6. Turney, P. D. Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews // Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL'02), Philadelphia, Pennsylvania, 417–424 (2002).

Метод распознавания шрифта текста с изображения

Коробов Дмитрий Владимирович, студент;

Патин Михаил Владиславович, студент

Санкт-Петербургский государственный университет

В настоящее время в интернете и социальных сетях прослеживается тенденция к передаче и обмену информацией с помощью цифровых изображений. В связи с этим большим спросом пользуются программы, позволяющие среднестатистическому пользователю работать с данными, полученными с изображения.

В данной работе описывается алгоритм распознавания шрифта текста с изображения. Для проведения исследовательской работы была написана программа для мобильных устройств на платформе iOS.

Ключевые слова: компьютерное зрение, распознавание шрифта, контурный анализ, OpenCV, iOS

В настоящий момент развитие цифровых технологий позволяет людям передавать и принимать информацию с большой скоростью. Доступность и развитость современной техники сделало особо распространенным такой вид данных, как цифровые изображения. В связи с этим большим спросом пользуются программы, позволяющие среднестатистическому пользователю получать и работать с данными, полученными с изображения. Эти задачи можно решить с помощью методов цифровой обработки. Первоначально эти методы разрабатывались и исследовались специалистами, работающими в области прикладной математики. Сейчас создано несколько общедоступных библиотек, например OpenCV, позволяющих вести работу с цифровыми данными, однако их использование все еще требует определенного технического образования. Такие библиотеки компьютерного зрения не содержат готовых решений и являются лишь инструментом для создания рабочих алгоритмов.

Алгоритм

Для распознавания шрифта текста используется методы контурного анализа. Контур буквы или цифры с входного изображения сравниваются с контуром этого же символа

в различных шрифтах. Для данной работы первоначально была создана демонстрационная база шрифтов, которая содержит в себе эталонные изображения букв и цифр.

Алгоритм можно поделить на 3 этапа: предобработка изображения, выделение символов текста на изображении и сравнение контуров. Рассмотрим первый этап более подробно:

Предобработка изображения

1. Первым делом входное изображение следует перевести в градации серого. В OpenCV это делается с помощью функции `cvtColor()`.

2. Далее используется медианный фильтр, для удаления шумов. Он хорошо справляется с шумами типа «перец и соль», а так же данный фильтр улучшает контур (рис. 1), что очень важно для дальнейшего контурного анализа. В OpenCV для этого ис

3. Используется функция `medianBlur()`.

4. Следующий шаг включает в себе бинаризацию изображения или перевод изображения в черно-белое (рис. 2) с помощью функции `thresholdMat()`. Бинаризация проводится по пороговому значению яркости пикселя, которое равняется 127.

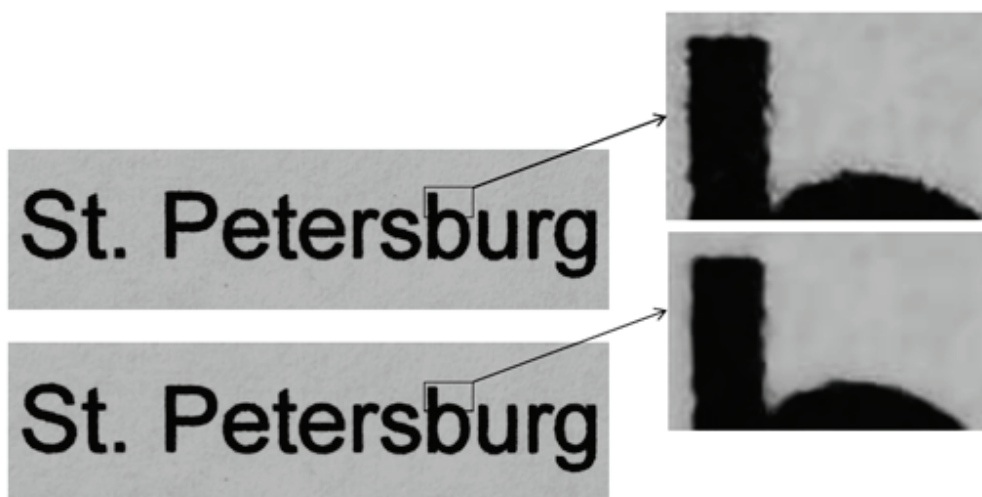


Рис. 1. Фильтрация шумов

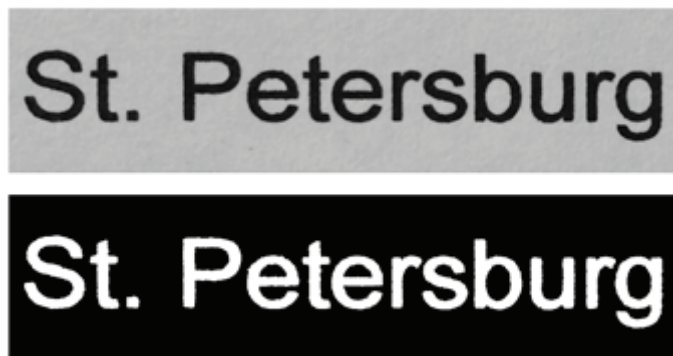


Рис. 2. Бинаризация изображения

Выделение символов

На данном этапе происходит выделение каждого символа в отдельное изображение.

1. С помощью детектора границ Кэнни, реализованного в OpenCV в виде функции `Canny()`, на изображении выделяются границы всех объектов (рис.3).



Рис. 3. Выделение границ

2. Затем с помощью функции `findContours()`, можно получить контуры объектов в виде набора точек с координатами. Имея набор точек для каждого контура можно

определить наименьший прямоугольник, который будет содержать область внутри контура. Найти прямоугольник можно с помощью функции `boundingRect()` (рис. 4).

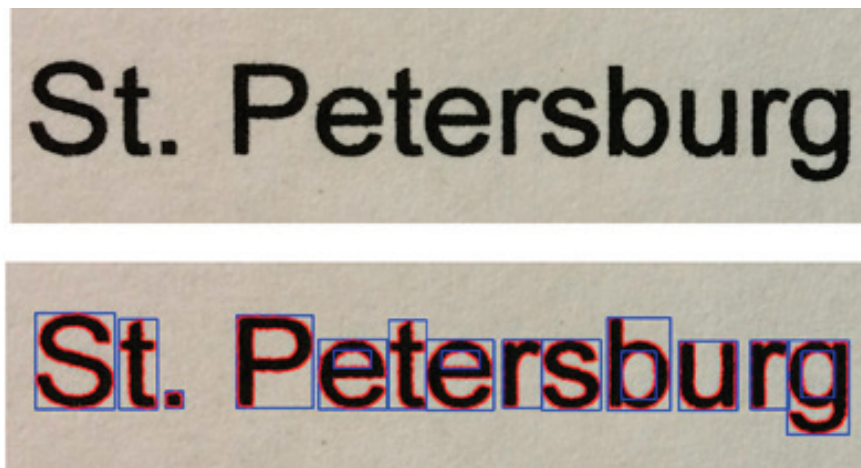


Рис. 4. Выделение контуров

Прямоугольник будет опять же представлен в виде набора точек его контура. Используя эту информацию, выделяем на начальном изображении контур каждого пря-

моугольника и сохраняем внутреннюю область в виде нового изображения (рис. 5).



Рис. 5. Выделенные объекты

Сравнение контуров

Контур каждого символа сравнивается с контуром этого же символа в различных шрифтах с помощью функции `computeDistance ()` из библиотеки `OpenCV`.

Принцип работы функции следующий: на вход подается два контура, а на выходе численное значение. Чем больше число, тем меньше похожи контура. В основе алгоритма лежит сравнение точек контуров между собой (рис. 6).



Рис. 6. Сравнение контуров

Отыскиваются похожие точки и строится гистограмма различий, по которой считается расстояние. Подробная реализация алгоритма, заложенного в основе функции `computeDistance ()` описана в статье *Belongie S., Malik J., Puzicha J. «Shape matching and object recognition using shape contexts»* [1].

Рассмотрим результаты полученные при использовании функции `computeDistance ()`. Для примера были взяты изображения трех символов в семи различных шрифтах: Arial, Calibri, Cambria, Candara, Javanese Text, Mongolian Baiti, Times New Roman (рис. 7–9).

Исходное изображение	Arial	Javanese Text	Cambria	Mongolian Baiti	Times New Roman	Calibri	Candara
	0,0296	0,0301	0,0632	0,0709	0,0829	0,1061	0,1145

Рис. 7. Результаты для буквы S

Исходное изображение	Arial	Candara	Calibri	Cambria	Mongolian Baiti	Times New Roman	Javanese Text
	0,0308	0,0331	0,0431	0,1418	0,1499	0,1574	0,2696

Рис. 8. Результаты для буквы t

Исходное изображение	Arial	Candara	Calibri	Cambria	Javanese Text	Mongolian Baiti	Times New Roman
	0,0406	0,0497	0,0636	0,4006	0,9278	1,0038	1,0982

Рис. 9. Результаты для буквы P

На рисунках 7–9 показано, как меняются значения функции `computeDistance()` от шрифта к шрифту. Не вооруженным глазом видно, что похожие буквы имеют близкие выходные значения функции, и как эти значения увеличиваются с при добавлении различных элементов на букву. Например, буква P имеет характерные засечки в шрифтах Cambria, Javanese Text, Mongolian Baiti, Times New Roman. Можно наблюдать, что выходное значение делает большой скачок при переходе от шрифта Calibri к Cambria.

Определение шрифта

Процедура сравнения с эталонными изображениями в различных шрифтах производится для каждого символа. Первые три наилучших шрифта, т.е. имевшие наименьшие выходные значения функции `computeDistance()`, получают от 1 до 3 баллов соответственно. После того, как контуры всех символов будут сравнены, выбираются три шрифта с наибольшими баллами. Именно эти шрифты выбираются, как наиболее похожие с шрифтом текста с входного изображения.

Литература:

1. Garcia, G. B., Suarez O. D., Aranda J. L. E. Learning Image Processing with OpenCV. ISBN-13: 978–1–78328–765–9. Birmingham: Packt Publishing, 2015. 319 p.
2. Belongie, S., Malik J., Puzicha J. Shape matching and object recognition using shape contexts // IEEE Transactions on Pattern Analysis and Machine Intelligence. 2002. Vol. 24. No 4. P. 509–522.
3. Dawson-Howe, K. A Practical Introduction to Computer Vision with OpenCV. John Wiley&Sons Ltd, 2014. 235 p.
4. Suzuki S, Abe K. Topological Structural Analysis of Digitized Binary Images by Border Following // CVGIP. 1985. Vol. 1. P. 32–46.
5. OpenCV documentation. <http://docs.opencv.org/>.

Выводы

В ходе экспериментов было установлено, что функция `computeDistance()` дает достаточно точные результаты и чувствительна только к качеству входного изображения и эталонного изображения символа. К сожалению, существует один значимый недостаток — время работы. Так как сравнение контуров идет попиксельно, при хорошем входном изображении контуры могут достигать размерности от 250 до 500 пикселей. Неоднократный перебор такого массива требует большой вычислительной мощности.

Для исследований данный алгоритм был реализован в виде мобильного приложения для устройств на платформе iOS. Тестирование велось на устройстве iPhone 5s. Его характеристики: 2-ух ядерный процессор Apple A7 64-bit, 1.3 Гц, 1Gb RAM. На данном устройстве распознавание шрифта текста на изображении (рис. 1) заняло около 10 секунд.

Эффективность применения CRM-системы

Лещёв Владимир Александрович, студент

Национальный исследовательский университет «Московский институт электронной техники»

В настоящее время доступ к Всемирной паутине (WorldWideWeb) имеют 1/3 населения Земли (2 млрд. человек). Учитывая такую популярность, в Глобальной сети существует около 500 млн. веб-порталов, большую часть из которых составляют интернет-магазины (ИМ).

Главная цель последних — продажа своих услуг как можно большему числу лиц и как можно эффективнее. Для этого ИМ собирают всю возможную информацию о действиях пользователей.

Собирая историю покупок, историю просмотренных продуктов, историю переходов по веб-порталу (откуда пришли, куда ушли), можно предугадывать действия пользователей и предлагать последним актуальные товары в виде контекстной рекламы или электронных сообщений.

Часть таких данных позволяют обрабатывать сторонние корпорации, например, «Google Analytics» и «Яндекс.Метрика». Оставшуюся часть обработки данных берёт на себя Customer Relationships Management, в дословном переводе «управление взаимоотношениями с клиентами» (далее — CRM). CRM — деловая стратегия привлечения и управления клиентами. Это система, которая на входе имеет сведения о клиенте, а на выходе — информацию о том, как нужно поступать компании в целом или её подразделениям, включая персонал.

Прототипами современных CRM-систем являются амбарные и долговые книги, в которых вёлся учёт взаимоотношений с клиентами. Современные подобные системы начали появляться в 1987 году, вместе с компьютерная программа по управлению контактами.



Рис. 1. Цикл взаимодействия с клиентом

Основное применение CRM системы связано с организацией и управлением взаимосвязью с клиентами. Поэтому, в первую очередь, преимущества CRM системы проявляются в увеличении показателей продаж, в частности, увеличивается объем продаж, повышается их эффективность, снижается стоимость привлечения клиентов.

К основным преимуществам CRM системы можно отнести:

а. Повышение скорости принятия решений.

За счет объединения разрозненных данных о клиентах ускоряется процесс обработки и анализа данных. В результате, ответственные за взаимодействие с клиентами могут видеть всю историю контактов, более оперативно отвечать на запросы и принимать по ним решения.

б. Повышение эффективности использования рабочего времени.

CRM системы позволяют автоматически отслеживать важные события, связанные с клиентами, и выдавать уве-

домления. Персоналу нет необходимости искать эту информацию в разрозненных источниках.

с. Повышение отдачи от маркетинговых мероприятий.

Т. к. CRM системы хранят всю информацию о клиенте и историю взаимодействия, то маркетинговые акции становятся более клиентоориентированными. У компании появляется возможность организовать маркетинговые мероприятия, направленные на каждого конкретного клиента.

d. Повышение достоверности отчетов.

Систематизация информации повышает достоверность отчетов и точность прогнозов по продажам.

e. Определение ценности каждого клиента.

Позволяет организации определить и спланировать потребности в ресурсах для работы с тем или иным клиентом. CRM системы позволяют установить приоритет привлечения ресурсов в зависимости от ценности клиента.

f. Сокращение бумажного документооборота.

За счет автоматизации процесса взаимодействия с клиентом все документы могут быть переведены в электронный вид.

g. Сокращение оттока клиентов.

За счет применения CRM системы у персонала появляется доступ ко всем деталям взаимодействия с клиентом. Это улучшает качество и оперативность обслуживания запросов потребителей.

h. Устранение дублирования задач.

CRM системы могут интегрироваться с другими системами управления деятельностью, что устраняет двойную работу по передаче и обработке данных.

i. Упорядочивание процессов.

CRM системы позволяют объединить все процессы взаимодействия с клиентами в единую систему. Входы и выходы процессов становятся доступными для разных

процессов, что упрощает управление контрактами, проектами, событиями, продуктами и пр., которые связаны с каждым конкретным клиентом.

j. Повышение культуры управления.

Автоматизация процесса снижает зависимость решаемых задач от субъективных действий каждого из сотрудников. CRM системы задают единые правила работы и взаимодействия с клиентами.

k. Защита и сохранность данных.

За счет применения CRM системы можно организовать централизованное управление доступом к данным о клиентах и обеспечить их сохранность.

Современные CRM системы можно подразделить на несколько типов. Типы CRM систем выделяются в зависимости от решаемых задач и этапов цикла взаимодействия с потребителями, на поддержку которых направлены эти системы.



Рис. 2. Виды CRM систем

Выделяют три основных типа систем:

a. Операционные.

Этот тип CRM систем позволяет автоматизировать оперативную деятельность. Операционные CRM системы осуществляют обработку различных наборов данных, «привязанных» к каждому конкретному клиенту. Они необходимы для оперативной поддержки отделов продаж, маркетинга, а также отделов обслуживания клиентов (сервисная поддержка). Как правило, наборы данных, с которыми работают операционные CRM системы, включают в себя всю контактную информацию о клиенте, историю взаимодействия, виды, объем и количество покупок, каналы коммуникации с клиентом и пр. Информация из этих систем является базовой для работы аналитических CRM систем.

b. Аналитические.

Этот тип систем необходим для поддержки этапов маркетинга и продаж на стратегическом уровне. Они позволяют планировать маркетинговые компании и выби-

рать наиболее эффективные стратегии продаж. Аналитические CRM системы обрабатывают информацию из различных баз данных, проводят систематизацию информации, на основе определенных алгоритмов выявляют наиболее эффективные тенденции взаимодействия с клиентами.

c. Совместные.

Этот тип CRM системы обеспечивает совместную работу с клиентами, т.е. клиенты имеют возможность непосредственно взаимодействовать с CRM системой компании. Взаимодействия могут осуществляться через web-страницы, электронную почту, автоматическую голосовую связь и пр.

Эти типы систем являются базовыми. Производители CRM систем предлагают комбинации из указанных трех типов.

По мнению экспертов, развитию CRM-концепции в современной России мешает ряд причин, которые можно сгруппировать в 4 основных блока, (рисунок 1. 1).

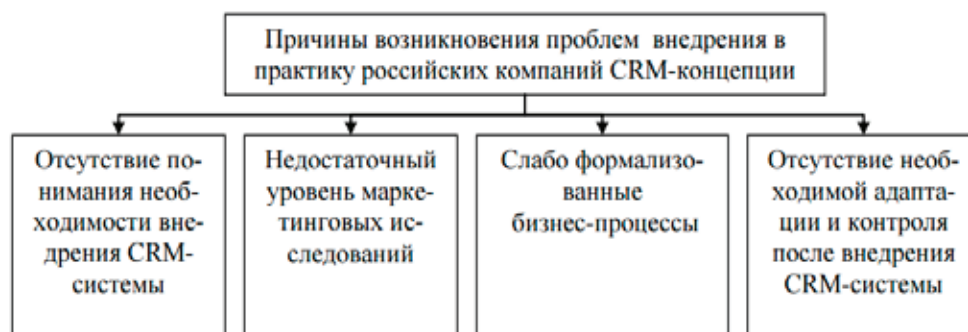


Рис. 3. Причины проблем внедрения CRM-систем

Многие компании, преодолев сомнения и внедрив систему, не получают необходимого результата. Причина этого в том, что руководство компаний не понимает CRM-концепцию. Такие компании можно условно разделить на две группы. Первая — это те, которые искали современный инструмент управления, но в проектах не было конкретных целей развития бизнеса. Были расплывчатые пожелания: «удобней работать в системе», но никакого конкретного измерителя не было. И вторая группа — компании, которые считали правильным внедрение лучших систем, если есть лишние деньги, но по каким-либо причинам оказались в кризисе и финансирование уменьшили. При этом те проблемы, с которыми они хотели бороться, не исчезли. Эти компании в отсутствие конкретных целей решили, что можно подождать, пока эти проблемы исчезнут сами собой.

Немаловажной проблемой является и сама организация процесса адаптации в компании CRM-системы. Её продвижение должно содержательно учитывать специфику отрасли. Существуют CRM-системы, созданные разными компаниями, такие как SAP, Oracle, Salesforce.com, Microsoft, Amdocs и другие. Каждый из созданных продуктов имеет одну идею, но разные функциональные возможности, в связи с чем в каждом случае требуется свой подход к хранимой информации, а, следовательно, изменение конфигураций CRM-системы, поскольку лишь только при корректной организации работы в CRM-системе можно добиться планируемых весомых результатов.

В настоящее время в России CRM, произведённый Microsoft, используют такие компании, как S7 Airlines, «Альфа Капитал», радиостанции СИТИ-FM и Relax FM (Газпром Медиа), группа компаний «ФНК», «Афиша», Торговый дом «Солнечные продукты», «Синтерра», «Уралмаш-Инжиниринг», завод «Теплоприбор», издательские дома «Статус Кво 97» и «Альпина Бизнес Букс», строительно-инвестиционная корпорация «Девелопмент-Юг», финансовый брокер «ФОСБОРН ХОУМ», банки «КМБ Банк», «ЕврофинансМоснарбанк» и «Возрождение» и многие другие компании из разных отраслей.

Наибольшую пользу CRM-системы приносят компаниям сектора «business-to-business», использующим

метод прямых продаж товаров и услуг конечному потребителю. Продукт или услуга здесь — товар, выгода от использования, которого потребителю не всегда видна сразу. Или же они (товар или услуга) находятся в высоко конкурентном рынке, предоставляющем клиенту многообразие выбора. Примерами таких компаний являются компьютерные, рекламные, консалтинговые фирмы, банки и др. Прямые продажи подразумевают непосредственное продолжительное взаимодействие сотрудников компании-продавца с клиентом. И успех взаимодействия часто зависит от того, насколько качественно менеджер подготовился к встрече с клиентом. Именно качество работы с клиентом, обеспечиваемое полнотой информации о нём, становится важнейшим конкурентным преимуществом компании. Особенно в случаях, когда бизнес компании построен на сделках с длительным циклом их осуществления.

Правильно внедрённая и грамотно контролируемая CRM-система в бизнесе данных компаний существенно улучшает работу отделов, которые непосредственно общаются с клиентом (отдел продаж, сайт и интернет-магазины, отдел маркетинга, сервисные отделы, абонентские службы, call-центр), что, в свою очередь, даёт возможность:

- планировать и отслеживать историю взаимодействий с клиентами;
- обеспечивать независимость сведений о клиентах от конкретных менеджеров;
- обнаруживать организации, с которыми не было взаимодействий долгое время, что, в свою очередь, может повысить эффективность работы с клиентской базой;
- проводить сегментацию клиентов (например, по отрасли, статусу и/или региону и т.д.);
- планировать и отслеживать будущие сделки (договоры) и платежи по ним;
- иметь систему напоминаний и оповещений сотрудникам и руководителям (например, об истечении срока выполнения задачи или поступлении нового клиента в работу и т.д.).

Таким образом, использование CRM систем позволит получить преимущества в бизнесе компаний и системе их отношений с клиентами, в деятельности её структурных подразделений, что несомненно позволит принимать гра-

мотные управленческие решения в отношении работы с клиентами, а также выстроить оптимальные стратегии

развития компании на перспективных для неё рынках товаров и услуг.

Литература:

1. Николаевская, Т. Н., научный руководитель — доцент Разнова Н. В. Сибирский федеральный университет «Актуальность и проблемы внедрения в практику Российских компаний CRM-систем».
2. Гринберг Пол. CRM со скоростью света (CRM at the speed of light.) СПб.: Символ Плюс, 2007. — 528 с.
3. <http://www.kpms.ru/> CRM системы
4. <http://www.tadviser.ru/> Возможности CRM систем

Технология тестирования программных модулей

Лещёв Владимир Александрович, студент

Национальный исследовательский университет «Московский институт электронной техники»

Тестирование программ является одним из самых длительных и ответственных этапов разработки. Особое значение придается ему в парадигмах структурного и надежного программирования.

При тестировании самыми важными являются два вопроса: в каком порядке тестировать модули и как готовить и задавать тестовые данные.

Трудоемкость тестирования связана с количеством самих ошибок, в связи с чем, надо четко выделить основные причины их появления:

- а. неудовлетворительное организационное, методическое обеспечение всего процесса разработки;
- б. большое число требований и их изменений по ходу работы;
- с. недостаточная квалификация разработчика.

Тест — это набор контрольных входных данных совместно с ожидаемыми результатами. Выделим следующие этапы тестирования:

- а. Тестирование модулей, которое производится в случае, если модуль может быть проверен автономно от других модулей. При этом необходимо убедительно продемонстрировать, что модуль работает корректно во всех граничных условиях.

б. Тестирование подсистем, целью которого является проверка сопряжения между собой модулей, входящих в подсистему, корректную передачу данных между ними.

с. Тестирование системы. На этом уровне выявляются ошибки сопряжения, сложные ошибки быстродействия и емкости, логические ошибки.

Для обнаружения всех ошибок в программе необходимо выполнить исчерпывающее тестирование. Чаще всего тестирование и отладка программного обеспечения проводится по знанию внутреннего объекта тестирования (рис. 1).

Белый ящик — это техника тестирования, которая позволяет проверить внутреннюю структуру программы, ее логику и корректность работы. Проводя тестирование Белого ящика, специалист руководствуется определенными знаниями программного кода, чтобы изучить значения выходных данных. Определяя принцип работы программы, он замечает отклонения программы от своей цели. При работе специалист имеет доступ к коду, тем самым тестируя внутреннюю структуру программы.

Серый ящик — это техника тестирования, которая позволяет проверить не только что делает программа, но и как она это делает. Знания о внутреннем устройстве программы, позволяют специалисту более точно под-



Рис. 1. Виды тестирования по знанию системы

бирать входящие значения и проверять выходящие, тем самым покрывать тестами более обширную область возможных дефектов. В этом случае специалист имеет доступ к внутреннему устройству программы, но тестирование производит с точки зрения конечного пользователя.

Чёрный ящик — это техника тестирования, которая позволяет оценить функциональность системы, руководствуясь её спецификацией. Специалист при проведении тестирования программы ограничивается прогонами на небольшом подмножестве всех возможных данных. Для этого он выбирает наиболее подходящие подмножества (подмножества с наивысшей вероятностью обнаружения ошибок). Здесь специалист использует только внешние рычаги взаимодействия с программой: с помощью пользовательского интерфейса или подключившись к тестируемой системе.

Если не известна тестируемая система, то используется метод Чёрного ящика. Если известны все детали реализации тестируемой программы, то используется метод Белого ящика. Если известны только некоторые особенности реализации тестируемой системы, то используется метод Серого ящика. Суть этих методов не сложная, но эффективность тестирования с помощью каждого из них требует хороших знаний и навыков.

При конструировании программы необходимо определить адекватный способ задания исходных данных — с клавиатуры (в текстовом или графическом режиме) или из файла.

При тестировании реальной программы набор возможностей задания данных должен быть существенно расширен, что позволяет выполнять тестирование более гибко, эффективно и с меньшими затратами. Можно отметить несколько различных способов задания исходных данных, использованных при создании демонстрационных программ, которые рекомендуется применять при тестировании пользовательских программ:

- a. ввод с клавиатуры в текстовом режиме (в ответ на программный запрос, указывающий содержательное название данного и множество допустимых значений);
- b. генерация в программе случайных чисел, задающих, как правило, элементы составных данных (масивов, файлов);
- c. генерация данных определенной структуры или с определенными свойствами с помощью генераторов тестов;
- d. тестовые файлы небольшого размера, содержащие специально подобранную последовательность элементов, на которых наглядно демонстрируется исполнение алгоритма;
- e. тестовые файлы большого размера (с произвольными или обладающими определенными свойствами данными) для получения количественных оценок алгоритма;
- f. произвольные пользовательские файлы, имена которых запрашиваются программой;
- g. графические объекты, вводимые с помощью графического курсора и клавиш.

В зависимости от целей программы исходные данные могут отображаться на экране программой или оставаться скрытыми от пользователя. В одной и той же программе можно предусмотреть несколько способов задания исходных данных.

Рассмотрим следующие правила тестирования программ:

– Правило № 1.

Тестирование является ключевой задачей разработки потому, оно не только является довольно длительным процессом, а потому что тестирование — очень ответственный этап разработки, и от того, как оно будет происходить зависит правильность и надёжность получаемой в итоге программы.

Кроме того, как правило, в процессе тестирования учебных программ выясняется необходимость выполнения ряда доработок программы, связанных с оценкой:

- a. интерфейса (удобства ввода данных и их контроля, точности и полноты выдаваемых сообщений);
- b. приемлемости времени исполнения программы;
- c. необходимого представления (типа) данных и точности вычислений.

– Правило № 2.

Цели тестирования для учебных программ весьма скромны — показать, что программа при ее исполнении ведет себя правильно, в соответствии с решаемой задачей, т.е. выдает правильные результаты для допустимых исходных данных и сообщает о невозможности их обработки (желательно на ранней стадии обработки) для неверных данных или их недопустимой комбинации. Осознание этих целей должно побудить разработчика (или тестировщика) спланировать достаточно полный пакет тестов, а также сроки разработки с учетом сложности тестирования программы.

– Правило № 3.

Оценка ожидаемых выходных данных основана на определении теста как двух наборов данных — исходных данных и ожидаемых результатов работы программы. Даже если тестирование не документируется (например, если исходные данные теста вводятся с клавиатуры, а результаты выдаются на экран), то перед тем, как вводить данные или в процессе их ввода, нужно обязательно предсказать ожидаемые результаты работы программы.

– Правило № 4.

Подготовка тестов как для правильных, так и для неправильных входных данных является важной частью. Под правильными данными здесь понимаются данные таких типов, диапазонов, форм задания, размеров (для агрегатных данных), которые допускает программа. Однако, программа будет правильной (надежной), если она будет адекватно воспринимать и недопустимые данные, т.е. контролировать правильность задания данных как в отдельности, так и в совокупности (возможные наборы значений), и в случае неправильного задания данных, выдавать об этом вразумительные сообщения или выполнять некоторые действия.

– Правило № 5.

Пакет тестов представляет собой совокупность тестов (исходные данные и ожидаемые результаты), по возможности охватывающих все случаи задания исходных данных задачи.

Это, в частности, означает, что в пакете тестов должны быть тесты, задающие как допустимые, так и недопустимые программой значения исходных данных, находящиеся:

- a. внутри допустимого интервала (множества),
- b. на каждой из границ интервала,
- c. вне допустимого интервала.

Кроме того, должны быть рассмотрены все «вырожденные» случаи задания исходных данных:

- a. пустые строки, множества и файлы,
- b. последовательности (представляемые массивами) нулевой длины и длины, превышающей допустимую программой (по каждому из измерений массива).

– Правило № 6.

Когда исходные данные теста задаются для того, чтобы посмотреть, «что получится» — это бессмысленная работа. Даже если прогон теста не документируется, ввод его исходных данных с клавиатуры должен быть осмысленным: это должны быть данные, которые можно воспроизвести еще раз, т.е. представления данных (числовые, строковые) должны быть короткими, однообразными, подчиненными каким-либо правилам, одним словом, запоминающимися.

– Правило № 7.

Детальное изучение результатов каждого теста необходимо. Лучшим средством анализа результатов тестирования является документирование результатов (запись их в файл) и последующее их сравнение с ожидаемыми (тоже задокументированными) результатами. Это сравнение нетрудно выполнять автоматически некоторой программой, даже если сравниваемые данные могут совпадать не

в точности, а с некоторой прогнозируемой погрешностью (например, вещественные числа). Но если делается даже простой визуальный анализ выдаваемых на экран значений, он должен быть тщательным и исчерпывающим. Непременным требованием при этом должно быть сохранение на экране исходных данных теста (а иногда и формулировки задачи) для того, чтобы можно было воспроизвести ожидаемые результаты.

– Правило № 8.

По мере того, как число обнаруженных ошибок увеличивается, растет также относительная вероятность существования в ней необнаруженных ошибок.

Возрастание числа обнаруживаемых ошибок в программе при ее тестировании обычно обуславливается следующими обстоятельствами.

a. Программа изначально составлена (спроектирована) очень плохо, так что никакие ее локальные исправления, выполняемые при тестировании и отладке, не могут ее скорректировать. В этом случае необходимо вернуться к ее перепроектированию, отбросив, возможно полностью, старый вариант программы;

b. Неправильные изменения программы в процессе ее тестирования, причем с каждым изменением программа может все более отдаляться от правильного варианта.

Выход из этой ситуации — вернуться к одному из предыдущих вариантов программы. Для этого рекомендуется при тестировании и отладке время от времени (перед существенным ее изменением) сохранять вариант текста программы, как-то идентифицируя (например, номером) разные его варианты.

Сегодня существует множество подходов к решению задачи тестирования и верификации программного обеспечения, но эффективное тестирование сложных программных продуктов — это процесс в высшей степени творческий, не сводящийся к следованию строгим и четким процедурам или созданию таковых.

Литература:

1. <http://www.gateslab.com/> Тестирование «Белого ящика».
2. <http://www.galight.com/> Тестирования методом серого ящика.
3. <http://www.wikiwaqnd.com/> Тестирование по стратегии чёрного ящика.
4. <http://www.programmer-lib.ru/> Тестирование программ.

Обзор методов и средств автоматизированного сбора информации с новостных лент

Романова Виктория Олеговна, магистрант
Московский технологический университет

Настоящая работа посвящена обзору источников данных с новостных лент в сети Интернет, программных средств хранения неструктурированных текстовых данных, лингвистических и статистических методов представления текстовой информации.

Сегодня Интернет теснит привычные медиа: все меньше людей читают газеты или смотрят телевизионные выпуски новостей. А в интернете те же люди сидят все больше и больше. Если прибавить к этому стремительные темпы развития технологий беспроводного Интернета, то мы получаем реальную перспективу массового распространения интернет-ориентированного потребителя новостной информации.

Трафик интернет-новостей за последние три года вырос в полтора раза, об этом свидетельствуют данные сервиса «Яндекс.Новости». По результатам исследования, ежедневно в сети публикуется около 47 тысяч информационных сообщений. Согласно исследованию, проведенному группой экспертов «Яндекса» в октябре 2014 г., на данный момент уже «каждый четвертый пользователь Сети в стране хотя бы раз в месяц читает новости в электронном виде».

Новостная лента представляет собой формат данных, используемый для доставки пользователям часто обновляемой информации. Лента состоит из некоторого ограниченного числа статей, а также из служебной информации, например, наименование поставщика ленты, адрес домашней странички. При появлении новых сообщений они добавляются в ленту, вытесняя старые статьи. Обычно в ленте бывает не больше 10–20 статей.

В РФ одним из самых популярных новостных сайтов, которые подают интернет-пользователям самые свежие новости политики, науки, спорта, культуры и техники является Lenta.ru.

Lenta.ru — одно из ведущих российских новостных интернет-изданий, основанное в 1999 году Антоном Носиком при содействии Фонда эффективной политики. Работает круглосуточно, освещая мировые и внутрироссийские новости.

По данным Alexa.com на март 2014 года сайт Lenta.ru занимает 16 место по популярности в России. В 2010 году Гарвардский университет провел исследование русской блогосферы, которое признало «Ленту.ру» наиболее часто цитируемым в русскоязычных блогах источником новостей. Согласно исследованию компании comScore, проведенному в апреле 2013 года, сайт занял 5 место по посещаемости среди европейских новостных сайтов.

Не теряет своей популярности и РИА Новости. Этот новостной сайт выкладывает новости России и других мировых стран, позволяя своим читателям всегда быть в курсе происходящего. Интернет-сайт агентства стабильно входит в десятку самых популярных электронных СМИ Европы. Ежедневно на сайте выходит более 50 новостных лент. Подобно сайту Lenta.ru каждая новость содержит заголовок, сам текст новости, дату публикации, имя автора, фотоматериалы. Пользователь может прокомментировать статью, оценить ее, ознакомиться с количеством просмотров, поделиться ссылкой на новость в таких социальных сетях, как Facebook, ВКонтакте, Одноклассники, LiveJournal, Twitter, МойМир.

Как отмечалось выше, ежедневно в сети публикуется около 47 тысяч информационных сообщений. В ответ на возникшие потребности на ИТ-рынке появился целый ряд решений, обозначаемых общим термином «Big Data», которые предлагают качественно новый подход к хранению и использованию постоянно растущих объемов информации.

Big Data — это серия подходов, инструментов и методов обработки структурированных, слабо структурированных и неструктурированных данных огромных объемов и значительного многообразия для получения воспринимаемых человеком результатов, эффективных в условиях непрерывного прироста, распределения по многочисленным узлам вычислительной сети, альтернативных традиционным системам управления базами данных и решениям класса Business Intelligence. В данную серию включают средства массово-параллельной обработки неопределённо структурированных данных, прежде всего, решениями категории NoSQL, алгоритмами MapReduce, программными каркасами и библиотеками проекта Hadoop.

Среди основных характеристик решений класса Big Data можно выделить такие как:

- большой объем — способность хранить массивы данных огромных размеров (от сотен Тб до десятков Пб);
- большое многообразие — возможность хранения в рамках одного массива всех видов структурированной и неструктурированной информации: таблицы баз данных, текстовые документы, видео- и аудио-информация, данные измерительных приборов, логи приложений и многое другое;
- высокое быстродействие — скорость загрузки и обработки информации в массиве близкая к режиму реального времени.

На рынке есть достаточно много мощных, легко масштабируемых решений для организации хранилищ данных и работы с ними. Среди них можно выделить Greenplum, Netezza и Teradata, которые предлагают эффективную производительность при обработке информации благодаря использованию массивно-параллельной архитектуры.

Технологии Big Data предусматривают высокую степень распределения данных на уровне хранения. Распределенная система хранения не только обеспечивает отказоустойчивость, но и позволяет распараллелить обработку данных, что крайне важно при работе с объемами Big Data.

Ключевой технологией хранения в мире Big Data является Hadoop. Hadoop представляет собой программную среду с открытым исходным кодом, которая обеспечивает распределённое хранение огромных объёмов данных (в масштабе петабайт) на недорогих компьютерах. С помощью Hadoop-приложения Big Data взаимодействует с узлами хранения без непосредственного обращения к физическим серверам.

Hadoop состоит из двух основных компонент:

1. Распределенная и в высокой степени масштабируемая файловая система HDFS (Hadoop Distributed File System).

2. Подсистема MapReduce на уровне приложений, которая обеспечивает обработку запросов в пакетном режиме.

HDFS построена по принципу однократной записи и многократного чтения (write-once-read-many) и имеет блочную структуру, в каждом блоке которой можно хранить файл или часть файла.

В общем виде, все методы обработки текстовой информации могут быть разделены на две группы, по используемым в их реализации подходам: статистическому и лингвистическому.

Лингвистический подход предполагает анализ различных единиц языка, начиная от морфем и заканчивая словами и предложениями и определение связей между ними для решения конкретных задач. Из-за этого лингвистический подход зависит от конкретного языка.

Лингвистический подход при решении задачи автоматической обработки текста предполагает последовательный анализ языка как иерархической системы. Выделяют 4 последовательных этапа анализа текста: лексический, морфологический, синтаксический и семантический. К этим базовым уровням анализа также может быть добавлен один над-уровень: прагматический анализ.

Если лингвистический подход к обработке текстовой информации предполагает разработку подробной лингвистической модели для решения конкретных задач, то статистический подход предполагает, что обработка больших объёмов текстовых данных с помощью компьютеров позволит извлечь из них определённые сведения, применимые для решения задачи автоматической обработки текста.

Статистический подход к решению задач автоматической обработки текста подразумевает использование некоторой математической модели текста, применимой для решения конкретной задачи. Под математической моделью понимается некоторый объект-заместитель объекта-оригинала, обеспечивающий изучение некоторых свойств оригинала. Таким образом, модель является своего рода эквивалентом объекта, отражающим в математической форме его важнейшие свойства — законы, которым он подчиняется и связи, присущие составляющим его частям.

В основе статистического подхода лежит использование в качестве основных единиц текста слов: на их

основе рассматриваются различные числовые характеристики, такие как распределение слов по тексту, количество повторений слов и т.д., а также статистические закономерности между этими характеристиками, позволяющие решать конкретные задачи. Достоинством статистического подхода является его независимость от конкретного языка. На практике чаще всего применяются комбинации двух этих подходов.

Освещая проблему выявления агрессии в сообщениях новостных лент, следует учитывать, что в СМИ именно через слово воздействуют на массовое сознание. Заголовки статей, привлекая внимание читателя, одновременно воздействуют на читателя и настраивают на отрицательное отношение к тем фактам, о которых идёт речь. Что касается агрессивно написанных статей, то такие материалы больше всех привлекают внимание читателей, воздействуют на эмоциональную сферу человека и настраивают на решительное действие.

В современных газетных текстах, помимо проявления речевой агрессии, можно также наблюдать случаи описания физической агрессии. Такие статьи, в основном, информируют о совершении разных насильственных действий, когда идет детальное описание агрессивных актов, начиная от побоев и заканчивая совершением убийства. В языковом сознании русских понятие речевой агрессии в СМИ почти всегда связано с понятием власти.

Одним из способов выражения агрессии является немотивированное, затрудняющее понимание текста использование иноязычной лексики. Например, *«Барак Обама на этой неделе имел большое европейское турне — Брюссель, Гаага, Рим, саммит Евросоюза, НАТО, визит к папе Римскому, что, кстати, интересно, потому что Обама сам позиционирует себя как inbeliever — неверующий»*.

В последнее время в СМИ активно используются слова, являющиеся неологизмами среди жаргонных слов. Например, *«Двадцать пять лет американцы «троллили» Россию»*.

В процессе исследования существующих методов и средств автоматизированного сбора информации с новостных лент, была рассмотрена одна из первостепенных задач — задача автоматизации сбора и хранения текстовой информации.

Литература:

1. http://www.onlinegazeta.info/portal/LENTA.RU-oficialniy_sait_internet-izdanie_lenta-ru-novosti-na-lente-redakciya.htm (дата обращения: 26.04.16)
2. <http://www.rg.ru/2014/10/22/novosti-site-anons.html> (дата обращения: 26.04.2016)
3. <http://www.kakprosto.ru/kak-846340-samye-populyarnye-novostnye-sayty#ixzz471ZHыLQr> (дата обращения: 26.04.2016)
4. Новая газета. — 2014. — № 24 от 3.04.2014

Анализ систем управления содержимым для создания информационного портала

Рудников Антон Валерьевич, студент;

Шувалов Александр Александрович, студент

Национальный исследовательский университет «Московский институт электронной техники»

В наше время благодаря всемирной паутине интернет можно отыскать любую интересующую информацию в считанные минуты, познакомиться с интересными людьми, делиться любимыми музыкальными композициям, весело организовать свой досуг. При таких огромных возможностях, веб-сайты стали неотъемлемой частью жизни подавляющего большинства людей. Когда-то не существовало такого понятия, как «сайтостроение», а сами сайты считались редкостью и представляли собой простые статичные страницы, не имеющих даже намека на стилевое оформление. Сейчас же зачастую можно увидеть примеры гигантских информационных порталов с тысячами страниц содержимого, посвященных той или иной тематике. В зависимости от тематики и выполняемых функций, веб-сайты можно классифицировать на:

- корпоративные сайты — сайты, предоставляющие информацию о компании;
- интернет-магазины — задача таких сайтов заключается в продаже товаров;
- социальные сети — сайты, направленные на общение людей между собой;
- поисковые системы — предназначены для поиска страниц в сети Интернет;
- почтовые сервисы — представляют собой интерфейс для работы с электронной почтой и т.д.

В условиях современного рынка, невероятно сложно представить себе успешный старт-ап, не имеющего своего сайта. Сайт для бизнеса — это инструмент, который знакомит покупателя с Вашей деятельностью, продаёт товары покупателю или помогает найти контакты потенциальных партнёров для дальнейшего сотрудничества, вакансии соискателям на работу и заинтересованным потребителям. Но мало просто создать и разместить в сети сайт. Перед этим нужно провести тщательное маркетинговое исследование, которое даст понять, в какое русло направить разработку.

Объем информации на современных сайтах настолько огромен, что становится трудно управлять ею. Со стремительным ростом числа информации на веб-сайтах, появилась необходимость в системах управления содержимым (CMS). С приходом таких систем управления, традиционные ручные технологии разработки сайтов, требующие знания HTML и CSS, стали уступать место автоматизированным системам наполнения сайта, не требующие знаний в области веб-разработки. К тому же, разработка собственного информационного портала с нуля очень трудоёмкое и дорогостоящее занятие, хотя и позволяет создать свою уникальную структуру для реализации конкретной задачи. Нынешние системы управления имеют

большой функционал, которого вполне достаточно для осуществления запланированных задач, хотя для создания новых уникальных модулей и дизайна стоит заручиться поддержкой программиста. Оперативное редактирование содержания страниц, настройка модулей системы — эти операции может совершать рядовой пользователь, не вникая и не затрагивая исходный код.

Расширение различных сфер бизнеса и открытие новых компаний значительно увеличивает потребность в создании собственных сайтов. В связи с этим развитие готовых систем управления содержимым позволит снизить затрачиваемые ресурсы на создание сайта. Это является актуальной задачей современности.

В связи с большим количеством разнообразных систем управления контентом возникает сложность выбора наиболее подходящего программного решения. Для избежания подобных трудностей выбора был проведен анализ нескольких популярных CMS.

Исходя из предположения об их популярности, был сделан вывод о том, что следующие CMS обладают наибольшей функциональностью:

- WordPress;
- Drupal;
- Joomla!
- DataLife Engine.

Рассмотрим каждую из систем управления более подробно:

WordPress

WordPress является наиболее популярной и бесплатной системой управления содержимым. По некоторым данным, 20% самых популярных сайтов из рейтинга Alexa Top работают под управлением этой системы [1].

Данная система управления изначально разрабатывалась, как CMS для создания блогов, но со временем функциональные возможности значительно возросли, позволило системе приобрести инструмент для создания новостных ресурсов и информационных порталов любой сложности.

К тому же, WordPress обладает обширной документацией, в том числе и на русском языке, позволяет легко расширять функциональность системы с помощью готовых модулей, а также самописных расширений.

Drupal

В последнее время всё большую популярность приобретает бесплатная CMS Drupal. Система славится обширным и постоянно совершенствующимся функ-

ционалом. Гибкие настройки содержимого страниц позволяют создать по настоящему уникальный сайт. Правда для плодотворной работы с этой системой требуются начальные навыки программирования и верстки, что зачастую вызовет трудности с разработкой у рядового пользователя.

Базовая версия движка предназначена для создания блогов, новостных ресурсов и интернет-форумов. Но благодаря расширяемости можно создавать сайты другой направленности. Drupal обеспечивают быстрый поиск информации на сайте, категоризацию содержимого, гибкость и динамичность настройки меню, ролевою модель доступа к сайту, сменные темы оформления, перевод информации сайта на разные языки.

Joomla

Joomla — бесплатная CMS для разработки информационных порталов. Система активно развивается, обладает качественной документацией на русском языке, является более дружелюбной для пользователя в отличие от системы Drupal. По статистике, Joomla является самой популярной бесплатной многофункциональной CMS. Её простота и удобство административной части позволяет без труда формировать страницы и наполнять их контентом, создавать навигационные меню и размещать и настраивать модули.

CMS обладает хорошей расширяемостью и открытостью системы, имеет большой выбор готовых тем оформления и дополнений, а также отличается высоким уровнем безопасности. Широкая реализация функциональной части, рассчитанная на простоту использования системы рядовым пользователем, чревата недостатками системы, таких как высокая нагрузка на сервер и низкая скорость работы.

DataLife Engine

DataLife Engine или DLE — это платная система управления содержимым, созданная отечественными раз-

работчиками. Основным направлением системы является создание сайтов для публикации новостей и статей. Большую популярность CMS получила среди больших информационных порталов, форумов и блогов, т.к. функционал системы в основном направлен на работу с новостными блоками. Гибкость и легкость в настройке и интеграции собственного дизайна, позволяет быстро вникнуть в структуру шаблонов и разработать уникальный дизайн для сайта. Вокруг этой системы сложилось большое сообщество русскоязычных пользователей, которые разрабатывают темы оформления и расширения функциональности для нее. Система широко распространена в России и признана одной из самых производительных отечественных систем управления содержимым.

Сравнительный анализ систем.

Для сравнительного анализа систем были выбраны наиболее важные аспекты для разработчика сайта:

1. Безопасность — защита от взлома, стабильность работы сайта.
2. Поддержка — наличие регулярных обновлений системы.
3. Наличие русскоязычной документации.
4. Визуальный редактор — редактор, позволяющий создавать материалы на сайте без знания HTML и CSS.
5. Работа с изображениями — встроенные средства для обработки изображений.
6. Расширения функциональности — количество и качество наработок сторонних разработчиков, которые расширяют функциональность сайта.
7. Комментирование — работа с комментариями пользователей, анти-спам защита.
8. Темы оформления — наличие базы готовых шаблонов оформления для сайта.
9. Легкость создания сайта — насколько легко неопытному пользователю, незнакомому с HTML и CSS, использовать систему.
10. Стоимость системы.

В таблице 1 представлена сравнительная характеристика рассматриваемых CMS. Каждый критерий оценен по шкале от 0 до 10 баллов.

Таблица 1. Сравнительная характеристика систем управления содержимым

Критерии	Системы			
	WordPress	Drupal	Joomla	DLE
Безопасность	8	9	10	9
Поддержка	8	10	8	7
Документация	7	8	10	7
Визуальный редактор	10	10	10	9
Работа с изображениями	9	8	9	10
Расширения	10	7	8	6
Комментирование	6	8	9	8
Темы оформления	8	6	10	7
Легкость	9	5	8	8
Стоимость	Бесплатная	Бесплатная	Бесплатная	3190 руб.

При создании сайта на основе CMS нужно четко понимать, какого рода содержание будет представлено на сайте, на какую аудиторию рассчитан веб-портал, и какого рода функционал необходим для реализации поставленной цели. Если это сайт-магазин, уклон нужно сделать на поиск и описание товаров и услуг. Для информационных сайтов нужно четко определить тематику контента

и придерживаться её на протяжении всего существования сайта. Так или иначе, для каждой конкретной задачи будет удобнее использовать ту или иную CMS, хотя стандартный функционал будет у всех примерно одинаков, и выбор, на какой системе управления содержимым останутся полностью зависит от предпочтений владельца сайта.

Литература:

1. Jesse Russell. CP/CMS. — 2012. — 96 с.
2. Грачев, А. С. Создаем свой сайт на WordPress. — М., 2011. — 160 с.
3. Колисеченко, Д. Н. Выбираем лучший CMS Joomla! и Drupal. — П., 2010. — 200 с.
4. Tom Canavan. CMS Security Handbook. — 2011. — 330 с.
5. WordPress Statistics // <https://hackertarget.com/wordpress-statistics-top-500k/> (дата обращения: 07.06.2016).

Автоматизированное заполнение плана проведения медицинских осмотров для сотрудников любого предприятия (на примере медицинского центра ООО «Олеся»)

Салтецкая Татьяна Васильевна, старший преподаватель;

Конарёва Екатерина Александровна, студент

Технический институт (филиал) Северо-Восточного федерального университета имени М. К. Аммосова в г. Нерюнгри

В настоящее время человечество стремится к повышению производительности труда или, иными словами, к автоматизации производства. Согласно определению, автоматизация — это «направление научно-технического прогресса, которое использует технические средства и математические методы с целью освобождения человека от участия в процессах получения, преобразования, передачи и использования какой-либо информации, либо существенного уменьшения степени этого участия или трудоемкости выполняемых операций» [2].

Говоря простыми словами, автоматизация есть применение самоуправляющихся машин или программ для выполнения различных процессов на производстве и в потреблении.

Сейчас уже разработаны и внедрены такие крупные системы автоматизации производства, как производственные роботы, программы и устройства проектирования и т.п. Все больше вводятся в эксплуатацию малые автоматизированные процессы: интернет-магазины, электронные журналы ведения складского учета и т.п.

Несмотря на это, люди не перестают упрощать свою жизнь, ведь есть еще огромное множество мелких отраслей промышленности, где все еще ведутся работы, использующие большое количество человек ресурсов, что отнимает силы, время и способствует появлению ошибок. Одним из таких примеров является ведение документооборота либо на бумажных носителях, либо с помощью печати на компьютере вручную.

Не являются исключением для попыток провести автоматизацию и медицинские организации: начиная от создания рабочего места врача или регистратора (составление медицинских карт, рецептов, выписка справок по шаблонам и формам и т.п.) и заканчивая комплексными решениями для лечебно-профилактических учреждений (электронная регистратура, on-line запись на прием, проведение медицинских осмотров и т.п.).

Здесь появляется такое понятие, как автоматизация медицинских учреждений — создание единой информационной системы, позволяющей значительно упростить некоторое количество рабочих процессов и повысить их эффективность, что должно привести к улучшению условий работы и качества предоставляемых услуг. Информационная система, в свою очередь, — это совокупность содержащейся в базах данных информации и обеспечивающих ее обработку информационных технологий и технических средств или, другими словами, комплекс методов, программ, технических, информационных и организационных средств, которые обеспечивают поддержку процессов функционирования информатизируемой организации [9].

Если рассматривать информационные системы в контексте медицинских учреждений, то уже медицинские — далее они будут применяться в узком смысле данного понятия — как информация непосредственно о человеке как о пациенте [5].

В качестве примера будет рассмотрен медицинский центр ООО «Олеся», который был зарегистрирован

в 1992 г. в г. Нерюнгри в качестве частной фармацевтической компании, а затем в 1999 г. получивший статус общества с ограниченной ответственностью. Его основными видами деятельности являются фармацевтическая и медицинская.

Фармацевтическая деятельность представлена крупной сетью аптечных учреждений со складом. Ассортимент лекарственных средств и изделий медицинского назначения насчитывает огромное количество наименований. Через аптечный склад предприятие осуществляет заказную систему лекарственного обеспечения. ООО «Олеся» ведет сотрудничество с крупнейшими фармацевтическими компаниями России. В аптеках работают профессиональные провизоры и фармацевты [6].

К фармацевтической деятельности ООО «Олеся» также относится салон-магазин «Оптика», где осуществляется изготовление очков любой рецептурной сложности, реализация населению готовой очковой оптики, очковых оправ, контактных линз и средств по уходу за ними, аксессуаров различных ценовых категорий для взрослых и детей [6].

Медицинская деятельность осуществляется в консультативно-диагностическом центре ООО «Олеся». Центр включает кабинеты отделения профилактических осмотров и клиническую лабораторию. Все виды медицинской деятельности осуществляются в соответствии с законодательством РФ. Прием ведут квалифицированные врачи различных специальностей на договорной основе, договора заключены и с другими организациями для проведения на их базах дополнительных медицинских работ или исследований. Кабинеты оснащены необходимым оборудованием и техникой, которые регулярно осматриваются медицинским техником. Необходимые медикаменты и одноразовые инструменты центр «Олеся» получает по заявкам со своего аптечного склада [6].

Одним из направлений деятельности медицинского центра является проведение профилактических осмотров.

Данный процесс проводится строго в соответствии с приказом Минздравсоцразвития России от 12.04.2011 № 302-н «Об утверждении перечней вредных и (или) опасных производственных факторов и работ, при выполнении которых проводятся обязательные предварительные и периодические медицинские осмотры (обследования), и Порядка проведения обязательных предварительных и периодических медицинских осмотров (обследований) работников, занятых на тяжелых работах и на работах с вредными и (или) опасными условиями труда» [8].

Медицинские осмотры — работа трудоемкая не сколько в плане работы с пациентами, сколько в оформлении как первичной, так и последующей документации. ООО «Олеся» является небольшим предприятием, здесь ведется бумажного рода документооборот, а электронные документы заполняются вручную. Одной из таких работ является заполнение медицинской сестрой отделения профилактических осмотров планов проведения медицинских осмотров для обследования работников различных предприятий города Нерюнгри в документах Microsoft Office Excel. Сейчас появилась необходимость автоматизировать процесс данный процесс.

Указанный документ содержит шаблон — таблицу с большим количеством строк и столбцов, заполняемых на основе поименных списков работников, направленных на осмотр предприятиями, и приказа Минздравсоцразвития России № 302-н «Об утверждении перечней вредных и (или) опасных производственных факторов и работ...». Опираясь на первый, необходимо заполнить данные о работниках, а данные по врачам и лабораторным исследованиям заполнить выбранным символом (например, «1») в соответствии со вторым документом, учитывая, что не должно происходить повторений у каждого обследуемого, т.е. если имеется два различных вредных фактора, с которыми работает один и тот же врач, символ последнему присваивается лишь единожды.

Таблица 1. Шаблон плана проведения медицинских осмотров с внесенными данными о работниках

№	ФИО	Пол	Дата рождения	Профессия, должность	Вредные производственные факторы и (или) условия труда	профпатолог	гинеколог+цитология+флора	лор	хирург	...
1	Акимов В. П.	М	09.09.1958	слесарь-ремонтник 5 р	Пр.1 п. 2.4; 3.5					
2	Баринцова Н. И.	Ж	15.02.1957	подсобный рабочий 2 р	Пр.1 п. 1.3.3; 2.4					

Сейчас для автоматизации лечебно-профилактических учреждений имеется множество как комплексных, медицинских информационных систем, так и простых инфосистем по отдельным направлениям в медицине. К ним от-

носятся такие системы, как: 1С: Предприятие. Охрана труда, Комплексная медицинская информационная система (КМИС), web-сервис «Электронный медосмотр» и др. Далее кратко о каждой.

«Охрана труда» — конфигурация для 1С.Предприятие, которая помогает автоматизировать учет на рабочем месте специалиста по охране труда и промышленной безопасности. Ведется учет в соответствии с приказами и законами, имеется множество подсистем и различные возможности по загрузке, выгрузке, хранению данных. В каждой подсистеме существуют помощники, отслеживающие состояние информационной базы и, в случае обнаружения ошибок и несоответствий, формирующие задачи для пользователя. Все подсистемы автономны, но при этом они тесно взаимодействуют между собой [7].

КМИС — программный продукт, предназначенный для выполнения полноценной автоматизации лечебно-профилактических учреждений. Подходит для поликлиник, многопрофильных стационаров с различными отделениями, санаториев. Программный комплекс удовлетворяет всем требованиям законодательства. КМИС имеет большое количество подсистем, как специализированных, так и дополнительных возможностей для врачей-специалистов [3].

«Электронный медосмотр» — web-сервис, автоматизирующий процессы, сопровождающие организацию и проведение медицинских осмотров в медучреждениях. Сервис интегрирован с системой электронного документооборота «КриптоВеб», при работе с которым каждое учреждение получает индивидуальное зашифрованное сертифицированными алгоритмами ГОСТ хранилище данных. «Электронный медосмотр» обеспечивает выполнение единого последовательного процесса взаимодействия медицинских учреждений и предприятий, т.к. подразумевает подключение к системе как одних, так и других [4].

Обобщив сказанное выше, можно выделить основные преимущества рассмотренных программных комплексов, к которым относятся: повышение эффективности труда и качества обслуживания пациентов, облегчение и упро-

щение работы медицинского персонала, предоставление надежного доступа к медицинской информации, полное соответствие законодательству.

Однако у данных систем существуют и недостатки, к которым относятся: высокая стоимость покупки и обслуживания программного продукта, необходимость в дополнительном обучении, дополнительные расходы за найм программиста для доработки системы, загруженность интерфейса и ненужный организации функционал, комплексность решения.

С учетом всех преимуществ и недостатков было принято решение о необходимости создания собственной автоматизированной медицинской информационной системы, которая будет отвечать требованиям приказов, содержать только необходимые документацию и функционал и способствует повышению эффективности работы.

Для разработки системы автоматизированного заполнения документации для ООО «Олеся» была выбрана среда разработки Visual Studio Community 2015 и язык программирования C# в силу его простоты, гибкости и объектно-ориентированности [1].

Чтобы реализовать автоматизирующее приложение, необходимо структурировать и систематизировать основные положения приказа № 302-н «Об утверждении перечней вредных и (или) опасных производственных факторов и работ...», также взять за основу такие документы, как шаблон плана проведения медицинских осмотров и поименный список лиц, направленных на проведение медицинского осмотра.

Программа представляет собой окно с расположенными в нем элементами управления и компонентами. Рабочая область располагается на двух вкладках: «Информация о работниках» и «Вредные производственные факторы и работы и профессии, с ними связанные».

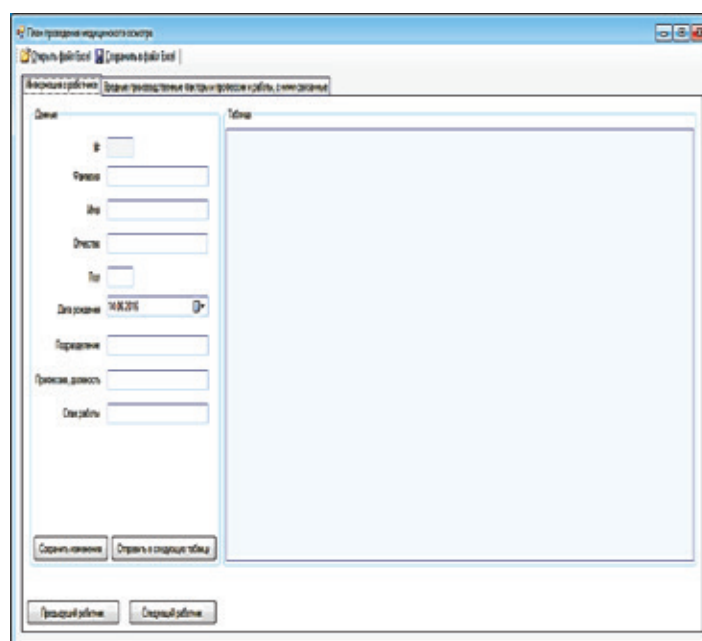


Рис. 1. «Информация о работниках»

В область таблицы второй вкладки с запуском программы автоматически загружается шаблон плана проведения медицинских осмотров.

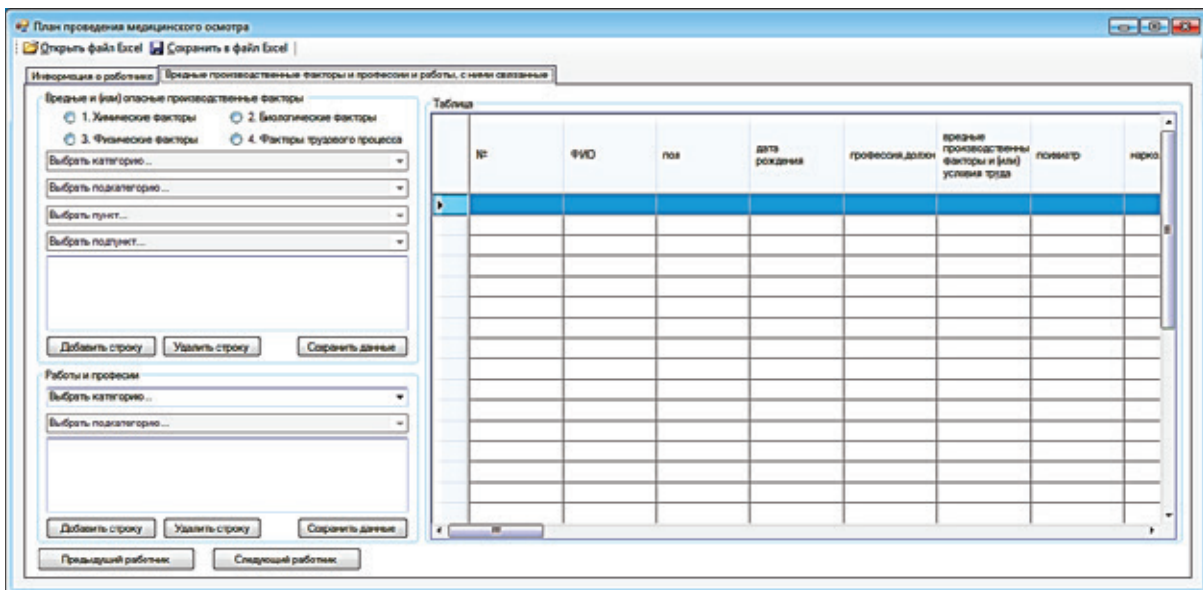


Рис. 2. Вкладка «Вредные производственные факторы...»

Для работы «Информацией о работниках» необходимо загрузить в область таблицы список лиц, направляемых на медосмотр. Данная операция реализуется с помощью

кнопки «Открыть файл Excel» и стандартного диалогового окна выбора файла для открытия.

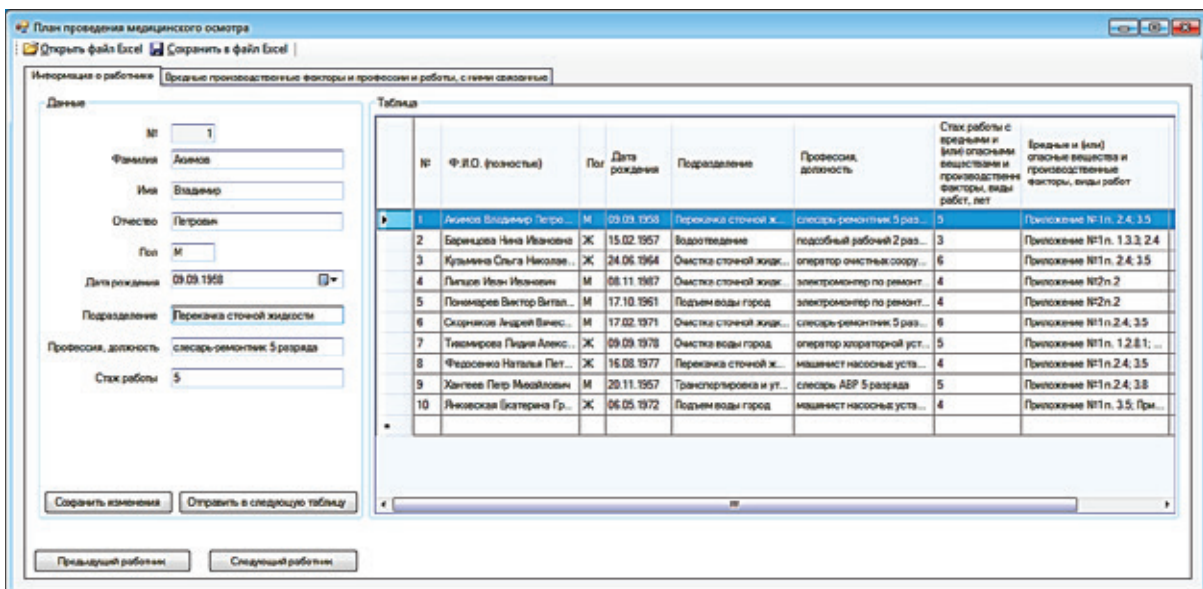


Рис. 3. Данные, загруженные в таблицу

Первая строка таблицы выделяется автоматически; перемещаться по таблице можно кликая левой кнопкой мыши в любое место нужной строки или с помощью кнопок «Предыдущий работник» и «Следующий работник». Данные выделенной строки заполняются в текстовые поля группы «Данные», которые

при необходимости можно редактировать и сохранить обратно в таблицу нажатием кнопки «Сохранить изменения».

По нажатию «Отправить в следующую таблицу» произойдет перенос данных таблицы с вкладки «Информация...» в таблицу на вкладке «Вредные факторы...».

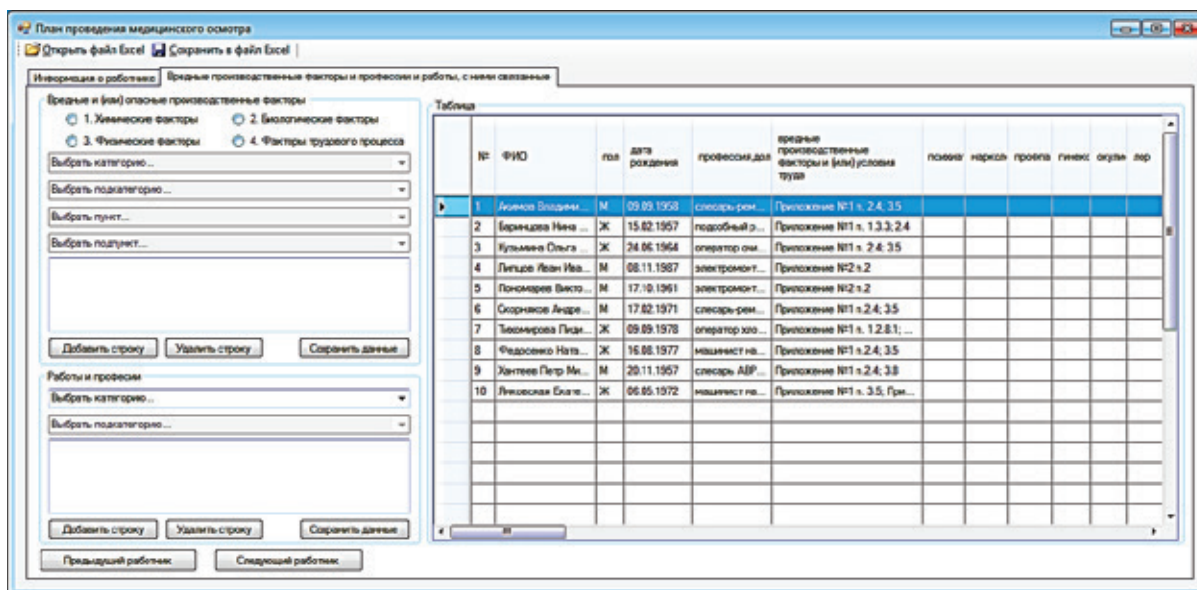


Рис. 4. Заполненная таблица вкладки «Вредные факторы...»

Используя выпадающие списки групп «Вредные и (или) опасные производственные факторы» и «Работы и профессии» необходимо выбрать пункты, соответствующие значениям из столбца «Вредные производственные факторы и (или) условия труда» и нажать кнопку «Доба-

вить строку» для записи последнего выбранного значения в текстовое поле. При необходимости строки можно удалять с помощью нажатия соответствующей кнопки. Сохранение значений в таблицу производится по нажатию на «Сохранить данные».

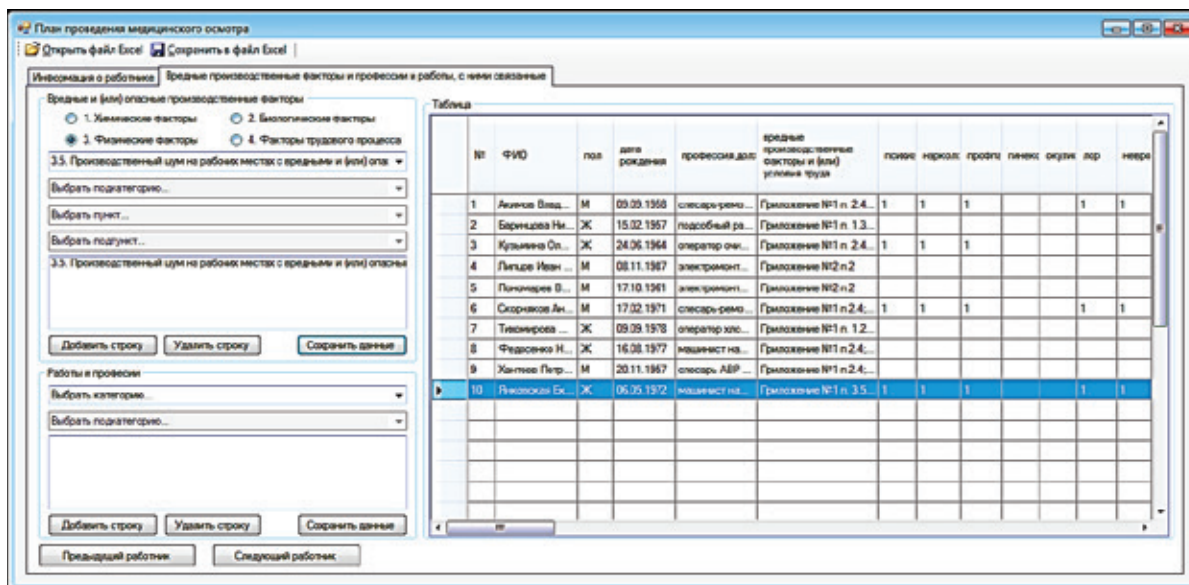


Рис. 5. Заполненные в соответствии с приказом № 302-н ячейки таблицы

По завершении заполнения нужно сохранить документ: нажать кнопку «Сохранить в файл Excel», выбрать имя файла и место для сохранения в окне «Сохранить как...».

Таким образом было создано приложение для автоматизированного заполнения плана проведения медицинских осмотров для сотрудников любого предприятия в медицинском центре ООО «Олеся». Программный

продукт осуществляет такие функции, как: импорт и экспорт файлов Microsoft Office Excel, добавление и удаление в случае необходимости данных, заполнение необходимого шаблона.

С учетом наличия на рынке аналогичных программных продуктов со своими преимуществами и недостатками, созданное приложение имеет следующие достоинства: отвечает требованиям приказа, просто

Таблица 2. Полностью заполненный план проведения медицинских осмотров

№	ФИО	пол	дата рождения	Профессия, должность	Вредные производственные факторы и (или) условия труда	психиатр	нарколог	профпатолог	гинеколог+цитология+флора	окулист	лор	невролог	хирург	дерматолог	биомикроскопия сред глаза	офтальмотонометрия
1	Акимов В. П.	М	09.09.1958	слесарь-ремонтник 5 р	Пр.1 п. 2.4; 3.5	1	1	1			1	1		1		1
2	Баринцова Н. И.	Ж	15.02.1957	подсобный рабочий 2 р	Пр.1 п. 1.3.3; 2.4	1	1	1	1		1			1	1	1
3	Кузьмина О. Н.	Ж	24.06.1964	оператор очистных сооружений 3 р	Пр.1 п. 2.4; 3.5	1	1	1	1		1	1		1		1

в использовании, не требует обучения, имеет привычный обычному пользователю интерфейс, содержит

только необходимый рассматриваемому медицинскому учреждению функционал.

Литература:

1. MSDN — сеть разработчиков Microsoft — URL: <https://msdn.microsoft.com/ru-ru/dn308572.aspx>
2. Википедия — свободная энциклопедия. Автоматизация — URL: <https://ru.wikipedia.org/wiki/Автоматизация>
3. К-МИС — комплексные медицинские информационные системы — URL: <http://www.kmis.ru/site.nsf/pages/index.htm>
4. КриптоВеб — защищенный документооборот — URL: <http://www.cryptoweb.ru/service>
5. Назаренко, Г.И., Гулиев Я.И., Ермаков Д.Е. Медицинские информационные системы: теория и практика / Под ред. Г.И. Назаренко, Г.С. Осипова. — М.: ФИЗМАТЛИТ, 2005. — 320 с.
6. ООО «Олеся» — URL: <http://www.olesya-irk.ru/qa/about.html>
7. «Охрана труда» для 1С: Предприятия 8. Комплексные системы безопасности труда — URL: <http://www.ot-soft.ru/>
8. Приказ Минздравсоцразвития России от 12.04.2011 № 302н — URL: http://www.consultant.ru/document/cons_doc_LAW_120902/
9. Федеральный закон от 27.07.2006 № 149-ФЗ «Об информации, информационных технологиях и защите информации» — URL: http://www.consultant.ru/document/cons_doc_LAW_61798/

Разработка программных модулей обработки многомерных данных различной природы в среде EXCEL

Степаненко Андрей Александрович, магистрант;
Владивостокский государственный университет экономики и сервиса

В настоящее время актуальной является проблема обработки многомерных данных различной природы, характеризующих различные социально-экономические и политические процессы в обществе. Проблема исследователя заключается в поиске доступных инструментальных методик, автоматизирующих процесс обработки и мониторинга больших объемов статистических данных. Огромный дефицит таких средств и высокая цена на рынке способствует развитию отечественных методик и инструментальных средств обработки таких данных. В рамках исследования, направленного на разработку доступных инструментальных методов обработки и мониторинга многомерных статистических данных различной природы в среде EXCEL, была разработана инструментальная методика, позволяющая проводить мониторинг календарного гра-

фика сбора данных с построением частотного ряда по заданным временным периодам для трех типов данных (числовых простых, простых текстовых и составных текстовых) и определением границ длины ряда для числовых простых.

Ключевые слова: алгоритмы, обработка данных, мониторинг, статистика, Excel, макрос, диаграмма, частотный ряд, анкетный опрос

В настоящее время в любой сфере государственного управления и бизнеса для повышения уровня стратегического управления решениями требуется полезная, реалистичная и качественная информация, направленная на решения задач устойчивого социально-экономического развития государства. Одним из источников полезной социально-экономической информации являются данные анкетных опросов различных групп общества. Многократное анкетирование позволяет получать динамические статистические представления состояния исследуемых групп общества, их мнений касательно изучаемого вопроса. Систематического сбор, мониторинг и обработка такой информации позволяет спрогнозировать возможные риски, а также улучшить процесс принятия решений, предотвращающих социальную политическую и прочую напряжённость в обществе.

Проблема исследователя заключается в том, что объем и характер таких данных можно обработать только техническими и инструментальными средствами. Присутствующие же на рынке статистические инструменты по обработке такой информации не обеспечивают решение всего спектра задач анализа анкетных данных. В основном применение данных средств ограничено рамками классических статистических методов анализа данных числовой природы. Кроме того, отечественный рынок сталкивается с огромным дефицитом таких средств, а существующие зарубежные инструменты требуют значительных денежных затрат и времени для их освоения. Поэтому выбор среды для разработки статистического инструмента определялся в первую очередь своей доступностью, популярностью, простотой использования и гибкостью настройки. Microsoft Excel очень хорошо подходит под эти требования.

В представленной статье авторами рассматривается разработанный алгоритм мониторинга и анализа данных признаков различной природы. Представленный алгоритм реализован в виде технического средства, являющегося надстройкой к EXCEL и предназначенного для обработки данных, собираемых с помощью анкетных форм. Данная надстройка была разработана при учете требований, предъявляемых ранее разработанным на базе ВГУЭС комплексом программ обработки и анализа анкетных данных Мартышенко С. Н. [2].

Рассмотрим постановку задачи по разработке программного средства обработки и анализа многомерных данных при мониторинге в анкетных опросах. Прежде всего, необходимо проанализировать процесс сбора данных при мониторинговых исследованиях и определить некоторые понятия, связанные с ним.

Главной особенностью сбора данных при мониторинговых исследованиях является то, что в каждой анкете должна быть указана дата сбора данных, которая при компьютерной обработке заносится в отдельный столбец таблицы данных.

Чаще всего анкетные данные собираются с использованием бумажного носителя, а затем уже вводятся в компьютер для обработки. В реальных исследованиях ежегодно по одной анкете может опрашиваться значительное количество респондентов. Сбор данных осуществляется в течение некоторого временного периода. В сборе данных может быть задействовано много технических работников, которые непосредственно контактируют с респондентами. Данные для ввода в компьютер от работников, осуществляющих сбор анкетных данных, поступают неравномерно. Каждая анкета должна иметь свой уникальный номер. По этому номеру можно идентифицировать анкету. После ввода в компьютер все данные собираются в единую таблицу, в которой данные упорядочиваются не по дате сбора данных, а по номеру анкеты. Это удобно для сверки компьютерных данных с оригиналом на бумажном носителе при выявлении ошибок. Данные на бумажном носителе сохраняются на весь период, пока существует перспектива сбора данных по данной анкете.

Следует различать понятия «периоды сбора данных» и периоды, на которые распространяются полученные оценки состояния системы — «периоды оценки состояния системы». В большинстве случаев, оценки распространяются на календарный год, но могут быть назначены и другие временные рамки.

Будем считать, что в конечном итоге будет сформирована некоторая таблица EXCEL, данные которой и нужно обработать. Разработанный программный комплекс требует, чтобы «Лист EXCEL», где располагается таблица «объект-свойство», имел название «Данные». Никаких прочих данных или расчетов на листе «Данные» быть не должно, кроме данных таблицы «объект-свойство» [1,5].

Для описания основных пользовательских требований была разработана диаграмма вариантов использования (рис. 1).

Алгоритм подразумевает два основных сценария использования, представленных двумя макросами, реализованными в VBA Excel. Взаимосвязь между интерфейсами и главным модулем, объединяющим основные функции и процедуры программы, представлена на диаграмме классов (рис. 2).

Первый макрос позволяет выполнить расчет таблицы календарного графика сбора данных, которая представляет собой мониторинг частоты сбора анкетных данных

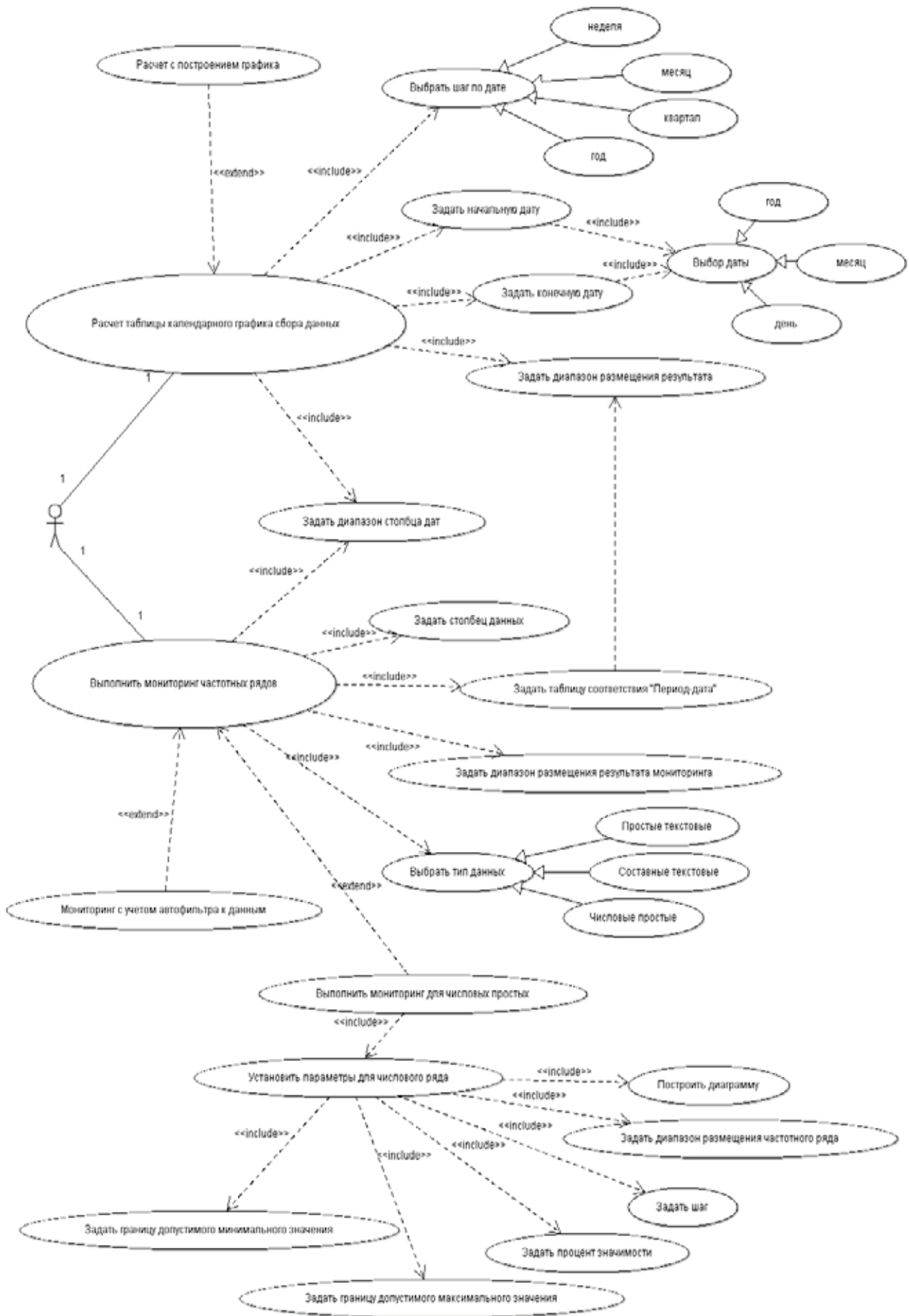


Рис. 1. Диаграмма использования (use-case) приложения

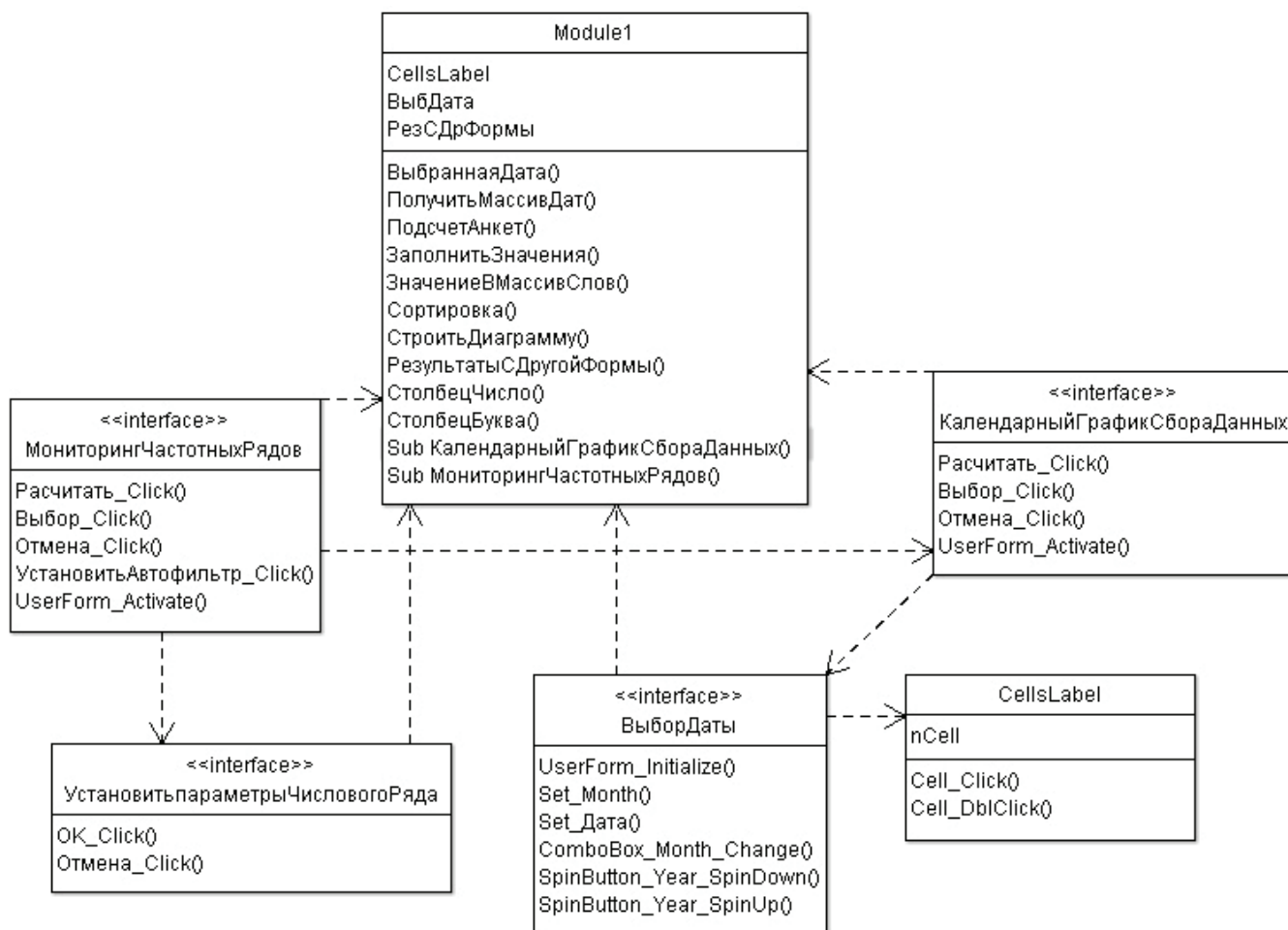


Рис. 2. Диаграмма классов приложения

в заданном диапазоне дат с разбивкой по периодам. В рамках функционирования данного макроса была автоматизирована возможность графического представления расчета. Для выполнения мониторинга пользователю необходимо задать столбец диапазона дат, затем указать начальную и конечную дату диапазона, шаг разбивки диапазона на интервал, а также определить курсором место для размещения результата. Для выбора даты была разработана форма «Выбор даты», инициализируемая по кнопке «Выбрать» (рис. 3).

При этом разбивку исследуемого диапазона можно осуществлять с интервалом длиной в неделю, месяц, квартал или год. Анализ графика процесса сбора данных по анкетному опросу позволяет спланировать работу по решению задач мониторинга исследуемого процесса или явления, также график отражает информацию о том, каким статистическим материалом располагает исследователь к моменту решения задач мониторинга [2,3,4].

Второй макрос позволяет проводить мониторинг частотных рядов по периодам, которые задаются таблицей

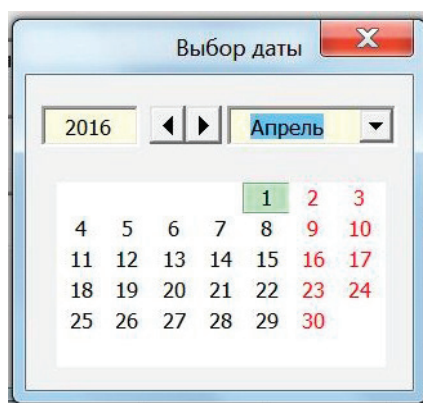


Рис. 3. Форма «Выбор даты»

соответствия «период-дата» [1,5]. Задать таблицу соответствия «период-дата» можно воспользовавшись расчетами, выполненными первым макросом. Также для выполнения расчета необходимо задать столбец дат, столбец данных, тип данных и место для размещения результата.

Часто при мониторинге необходимо построить серию частотных рядов не по всем данным, а по данным, от-

вечающим определенным условиям для некоторой группы респондентов. Для определения условий, описывающих такую группу была разработана кнопка «Установить/снять автофильтр на таблицу данных». По этой команде происходит обращение к системному диалоговому окну EXCEL «Пользовательский автофильтр» (рис. 4).

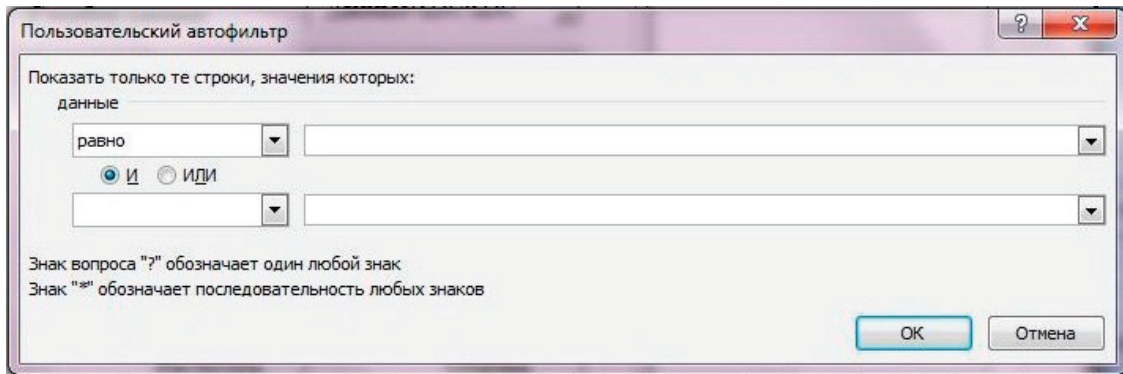


Рис. 4. Вызов диалогового окна EXCEL «Пользовательский автофильтр»

Анализируемые данные можно разделить на три типа: числовые, текстовые и составные текстовые, разработка алгоритма для каждого типа данных имеет особенности [3]. В отличие от алгоритма анализа простых текстовых, для составных текстовых данных необходимо задать маску, по которой происходит разбивка составного признака на несколько простых признаков. При этом частотный ряд для

двух видов текстовых данных представляет собой таблицу частоты повторения признаков в анализируемых периодах. Для числовых данных, кроме таблицы частоты повторения признака строится также и вариационный ряд с определенными предельными параметрами, которые пользователь задает на специально разработанной форме «Установить параметры для числового ряда» (рис. 5).

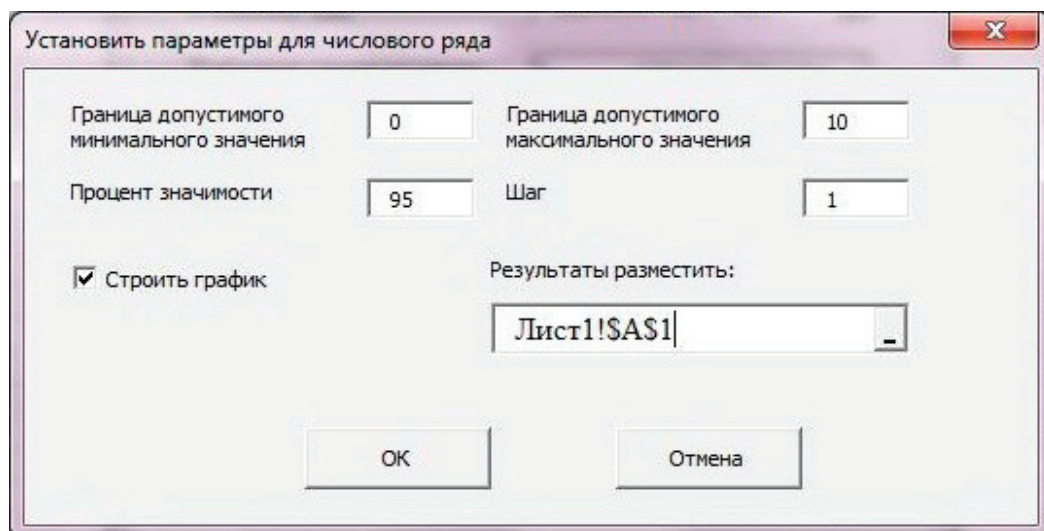


Рис. 5. Форма «Установить параметры для числового ряда»

Рассмотрим возможности, предоставляемые пользователю при установке данных параметров:

- параметр «Граница допустимого минимального значения» определяет нижнюю границу вариационного ряда, при этом по умолчанию значение устанавливается равное нулю;

- параметр «Граница допустимого максимального значения» определяет верхнюю границу вариационного ряда, за которой находятся предполагаемые выбросы.

- параметр «Процент значимости» определяет процент ответов в границе допустимого максимального значения (по умолчанию значение устанавливается равное 95%);

– параметр «Шаг» определяет разницу между верхней и нижней границами интервалов. По умолчанию значение устанавливается равное единице;

– параметр «Результаты разместить» определяет ячейку листа EXCEL, выбранную для размещения результатов.

Также имеется возможность построить диаграмму вариационного ряда, если будет выбран элемент «Строить график».

Значения характеристик выборки без учета выбросов и с учетом выбросов существенно отличаются, поэтому разработанная технология позволяет существенно повысить точность результатов и также существенно сократить время расчета частотных рядов признаков при мониторинговых исследованиях.

Подводя итоги исследования, направленного на разработку доступных инструментальных методов обработки

и мониторинга многомерных статистических данных различной природы, была разработана инструментальная методика, позволяющая проводить мониторинг календарного графика сбора данных и рассчитывать частотный ряд по временным периодам для трех типов данных, а именно числовых простых, простых текстовых и составных текстовых. Так же был разработан инструментарий, позволяющий задать ограничения при построении вариационного ряда для числовых данных.

Проведение мониторинга социально-экономических явлений в обществе с помощью разработанной технологии позволяет автоматизировать процесс анализа исследуемых явлений средствами Microsoft Excel всем участникам стратегического управления решениями в любой сфере государственного управления и бизнеса.

Литература:

1. Мартышенко, Н. С., Мартышенко С. Н. Современные тенденции изменения структуры времяпрепровождения студентов // Социально-экономические явления и процессы. — 2013. — № 1. с. 112–116.
2. Мартышенко, Н. С., Мартышенко С. Н. Технологии повышения качества данных в анкетном опросе // Практический маркетинг. — 2008. — № 1. с. 8–13.
3. Мартышенко, С. Н. Анализ данных мониторинга социально-экономических процессов в муниципальных образованиях // Информационные технологии моделирования и управления. — 2012. — № 6 (78). — с. 506–512.
4. Мартышенко, С. Н. Методическое обеспечение анализа данных мониторинга социально-экономических процессов в муниципальных образованиях // Экономика и менеджмент систем управления. 2012. — Т. 6. — № 4.2. — с. 259–267.
5. Мартышенко, С. Н., Мартышенко Н. С., Кустов Д. А. Цензурирование при обработке анкетных данных // Известия высших учебных заведений. Поволжский регион. — 2006. — № 6 (27). — с. 170–177.

Анализ тональности высказываний в Twitter

Стригулин Кирилл Анатольевич, студент
Журавлёва Любовь Викторовна, студент
Санкт-Петербургский государственный университет

В данной работе рассматривается применение методов машинного обучения с учителем к анализу тональности русскоязычных высказываний в социальной сети Twitter. Подробно разбираются методы предварительной обработки текста, описываются способы анализа тональности текста. Основываясь на теоретических исследованиях социальной сети Twitter, а также особенностях машинного обучения, приводятся оценки различных подходов к решению задачи. Для проведения исследовательской работы был написан комплекс программ на языке Python, который реализует рассмотренные подходы, и на реальных данных проводит численные эксперименты. Продемонстрированы результаты работы программ. Проведено сравнение результатов теоретических исследований и результатов, полученных в ходе эксперимента.

Ключевые слова: анализ тональности, классификация, data mining

За последнее десятилетие значительно возросло использование различных онлайн-ресурсов, в частности, социальных сетей, таких как Twitter. Многие компании и организации определяют эти ресурсы как значимые для маркетинговых исследований [1]. Обычно, чтобы получить обратную связь и понимание того, как покупатели относятся к их продукции, компании проводят

интервью, анкетирования и опросы. Эти стандартные методы часто требуют больших затрат времени и денег; более того, они не всегда приносят желаемый результат.

Каждый день в Интернет загружается огромное количество данных, содержащих потребительское мнение. Такие данные являются, в основном, неструктурированным текстом, из которого компьютеру сложно извлечь

мнение потребителя. В прошлом было невозможно обрабатывать такой большой объём неструктурированных данных, но сегодня это не составляет большого труда. Таким образом обработка естественного языка и анализ тональности играют важнейшую роль в принятии обоснованных решений о маркетинговых стратегиях и дают полезную обратную связь о продуктах и услугах.

Целью данной работы является проведение анализа тональности отзывов в социальной сети Twitter для определения репутации компаний. Данная цель включает в себя выполнение следующих задач:

1. Исследовать методы машинного обучения для задачи анализа тональности;
2. Провести анализ тональности для высказываний в Twitter;
3. Оценить полученные результаты.

Обзор предметной области

Анализ тональности текста — это раздел интеллектуального анализа данных, направленный на выявление закономерностей в текстах, с целью классификации настроения в данных текстах

Существует несколько способов произвести анализ тональности текстов:

- по правилам;
- по словарям;
- машинное обучение с учителем;
- машинное обучение без учителя.

Основанный на правилах:

Цель данного подхода — поверхностный синтаксический анализ на основе какого-то правила. Такой подход показывает хорошую точность при большом количестве правил.

Пример такого подхода: если высказывание содержит прилагательные из набора [‘хороший’, ‘добрый’, ‘новый’]

и не содержит прилагательных из набора [‘плохой’, ‘старый’, ‘ужасный’], то высказывание является положительным.

Основанный на словарях:

Данный подход подразумевает составление, либо использование готовых словарей тональности. Словарь представляет собой список слов с приписанной им тональностью. Каждому слову документа ставится в соответствие тональность из словаря, а затем подсчитывается общая тональность документа.

Основанный на машинном обучении с учителем [2]:

Данный способ представляет собой алгоритм с обучающими данными; эти данные состоят из обучающих примеров с предполагаемыми для них ответами. Алгоритм обучения с учителем использует эти данные, чтобы научиться сопоставлять подаваемые ему новые примеры с ожидаемыми ответами. В данной работе используется именно этот подход.

Основанный на машинном обучении без учителя:

Данный способ не требует обучающих данных, работа идёт с данными, где ответы не известны. Идея данного подхода в том, что чем чаще слова встречаются в одном тексте и чем реже в другом, тем большую значимость они играют для первого. Таким образом, для того, чтобы узнать тональность текста, нужно исходить из тональности таких слов.

Обзор данных

Коллекция обучающих и тестовых данных взята с международной конференции по компьютерной лингвистике и интеллектуальным технологиями «Диалог» [6]. Корпус состоит из 8643 обучающих твитов и 19773 — тестовых. Твиты содержат в себе информацию об операторах социальной связи в России. Высказывания делятся на положительные, отрицательные и нейтральные (таблица 1).

Таблица 1. Данные

Положительные	Отрицательные	Нейтральные
3326	3783	1534

Общий алгоритм анализа тональности

1. Предварительная обработка данных;
2. Построение вектора признаков;
3. Применение метода машинного обучения с учителем;
4. Оценка полученных результатов.

Предварительная обработка данных.

Подготовка данных является важной частью анализа тональности, поскольку анализировать можно любые данные, но для получения качественного анализа, данные должны быть очищены от загрязненных данных. Чтобы правильно подготовить данные, очень важно знать и по-

нимать область исследования, поскольку исследователь должен уметь распознавать, какие данные важны, а какие можно исключить, поскольку они не несут весомой значимости. Предобработка данных включает в себя очистку и преобразование данных.

Очистка данных: удаление дубликатов, знаков пунктуации, цифр, лишних пробелов, ссылок, хэштегов и прочих символов.

Преобразование данных: удаление окончаний (стемминг) и приведение всех слов к нижнему регистру.

Построение вектора признаков:

В задачах обработки текста на естественном языке популярно представление документов в виде n -грамм, где n -

граммы — последовательности слов длины n . В данной работе каждое высказывание определяется как вектор признаков (n -грамм) $d = (w_1, w_2, \dots, w_{|V|})$, где V — множество всех уникальных признаков из обучающей выборки; w_i — вес i -го признака. В качестве веса признака выбран булевский вес, т. е. $w_i = 1$, если признак присутствует в высказывании, либо 0 в противном случае. Из всех векторов d составляется матрица признаков. Данная матрица получается разреженной, т. к. многие признаки являются уникальными и встречаются только единожды. Чтобы упростить вычисления применяется свойство сингулярного разложения матриц [3].

В качестве методов машинного обучения были выбраны Наивный Байесовский метод и метод опорных векторов (SVM) [4].

Наивный Байесовский метод:

Данный подход основан на присвоении высказыванию d класса $c = \arg \max_c P(c|d)$

Согласно теореме Байеса [5] вероятность принадлежности высказывания d классу c выражается следующим образом:

$$P(c|d) = \frac{P(c)P(d|c)}{P(d)}$$

Чтобы оценить вероятность $P(d|c)$, Наивный Байесовский метод раскладывает её на величины f_i , и таким образом получается Наивный Байесовский классификатор:

$$P(c|d) = \frac{P(c) \prod_{i=1}^m P(f_i|c)^{n_i(d)}}{P(d)}$$

данных, $n_i(d)$ — количество встретившихся признаков в документе d

Метод опорных векторов: Суть метода опорных векторов заключается в том, что он отображает признаки в многомерном пространстве, а затем пытается разделить их гиперплоскостью. Если выборка линейно неразделима, то происходит переход от текущего пространства к пространству большей размерности, в котором выборка может быть линейно разделимой.

Оценка

В качестве оценки работы алгоритмов используется F_1 — мера:

$$F_1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$$

$$Precision = \frac{TP}{TP + FP} \text{ — точность; } Recall = \frac{TP}{TP + FN} \text{ —}$$

полнота

TP — истинно-положительные решения;

TN — истинно-отрицательные решения;

FN — ложно-отрицательные решения;

FP — ложно-положительные решения.

Результаты

Анализ тональности высказываний в Twitter посредством методов машинного обучения был реализованы на языке программирования Python.

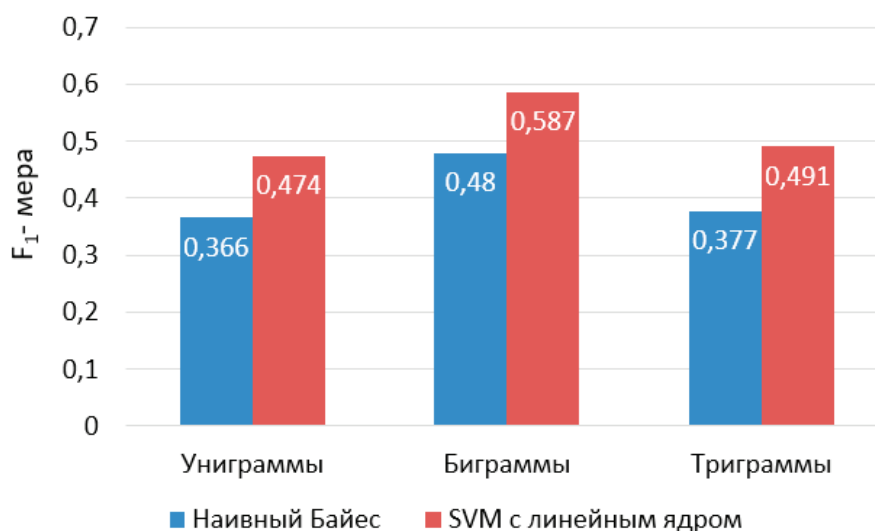


Рис. 1. Результаты работы Наивного Байесовского метода и SVM с линейным ядром

Как можно заметить из рис. 1, SVM с линейным ядром отработал лучше на всех n -граммах, причем наилучший результат был получен на биграммах.

На рис. 2 приведены результаты работы метода опорных векторов с линейным и полиномиальным ядрами.

Можно видеть, что полиномиальному ядру не удалось обойти линейное.

На рис. 3 видно, что лучший результат показал метод опорных векторов с линейным ядром. Лучший показатель у всех методов наблюдается на биграммах.

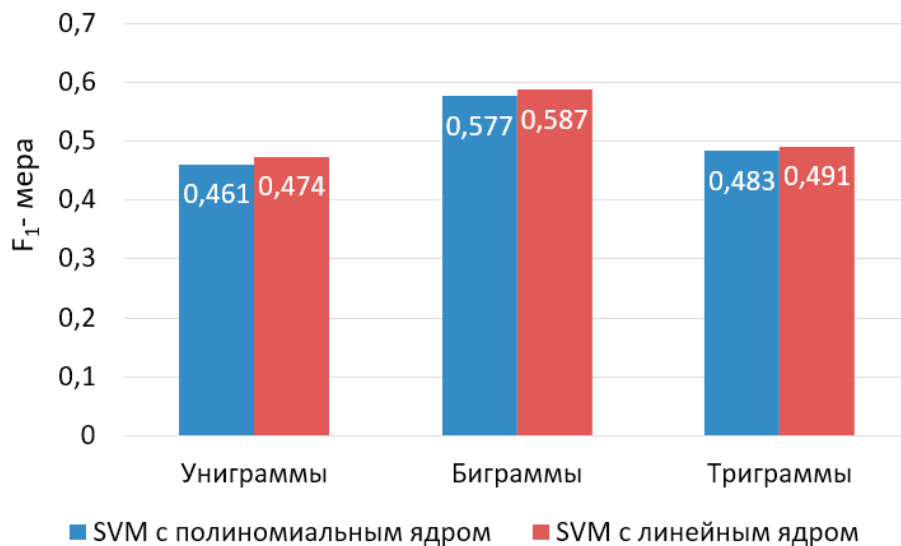


Рис. 2. Результаты работы SVM с линейным и полиномиальным ядрами

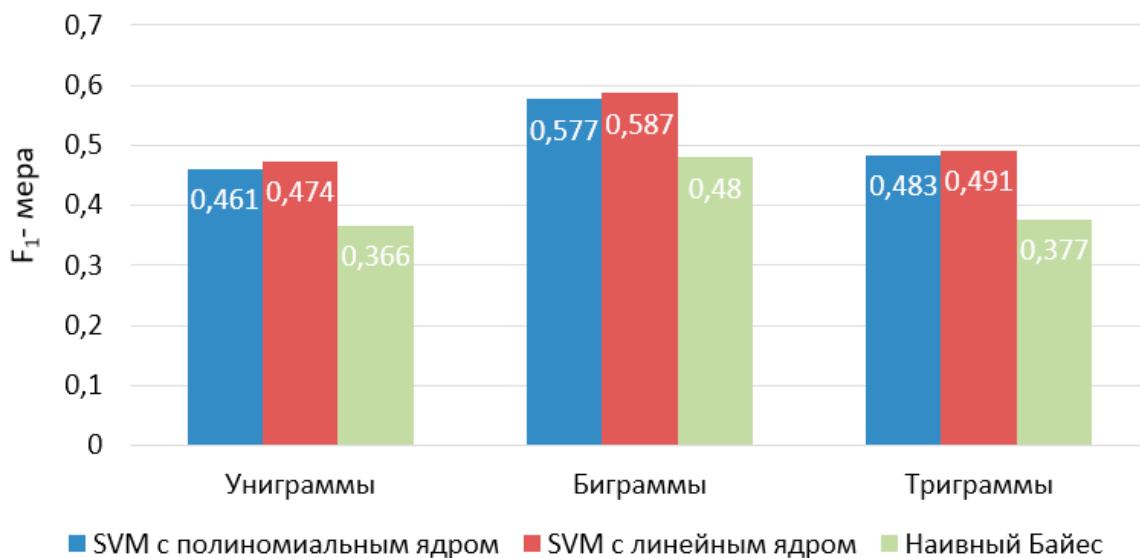


Рис. 3. Общие результаты работы методов

Выводы

Превосходство линейного ядра может быть обусловлено большой размерностью матрицы признаков и линейной разделимостью данных. Результаты с использованием биграмм могут быть объяснены спецификой русскоязычных высказываний в Twitter: ограниченностью

в 140 символов, содержанием орфографических ошибок и сленга.

Результаты работы классификаторов можно попытаться улучшить путём выбора других признаков, удалением стоп-слов, комбинированием n-грамм, использованием лемматизации, а также, возможно, стоит оставить некоторые знаки препинания.

Литература:

1. S. Owens, Your Guide to Twitter Marketing, March 2015.
2. B. Pang, L. Lee, S. Vaithyanathan, Thumbs up? Sentiment classification using machine learning techniques, 2002, pp. 79–86.
3. C. Cortes, V. Vapnik, Support-Vector Networks, 1995, pp. 273–297
4. https://en.wikipedia.org/wiki/Naive_Bayes_classifier
5. <http://www.dialog-21.ru/>
6. https://en.wikipedia.org/wiki/Singular_value_decomposition

Анализ страницы пользователя социальной сети «ВКонтакте»

Сумкин Константин Сергеевич, кандидат технических наук, доцент;

Тараненко Лилия Олеговна, студентка

Московский государственный технический университет радиотехники, электроники и автоматики

Статья посвящена анализу данных пользователя социальной сети «ВКонтакте» на основании информации, указанной на странице пользователя, а также по средствам анализа связей с другими пользователями.

Ключевые слова: Интернет, социальные сети, информация

В настоящее время развитие информационных технологий оказывает влияние и охватывает все сферы деятельности человека: политику, экономику, культуру, науку и другие. Информатизация общества приносит значительную пользу, расширяет и открывает новые возможности. Особую значимость приобретает развитие и использование Всемирной сети Интернет, что открывает перед человечеством безграничные возможности, связанные с поиском, передачей и распространением информации.

В современном информационном обществе Интернет является одной из возможных площадок для самовыражения личности. Особую популярность приобрели социальные сети. Социальная сеть — это интернет-площадка, сайт, который позволяет зарегистрированным на нем пользователям размещать информацию о себе и коммуницировать между собой, устанавливая социальные связи. Контент на этой площадке создается непосредственно самими пользователями [1].

Первая крупная социальная сеть Facebook появилась в США. В мире наиболее распространены сети MySpace, Facebook, Twitter, LinkedIn, Google + [2]. Самыми известными и популярными социальными сетями в России являются ВКонтакте, Одноклассники.ru, Мой Мир, Facebook, Twitter, набирает обороты Google +.

Анализ данных социальных сетей стремительно набирает обороты, так как социальные сети являются уникальным источником личных данных и интересов реальных людей. Это открывает безграничные возможности для решения разного рода исследовательских и бизнес-задач [3].

В данной статье рассматривается анализ страницы пользователя социальной сети на примере самой популярной в России сети — «ВКонтакте». Сам Процесс анализа страницы пользователя представлен на Рисунке 1. Остановимся подробнее на каждом элементе анализа.

Аватар. Одним из главных и самых заметных элементов страницы, а также главным инструментом самопрезентации пользователя является «Аватар», то есть основная фотография пользователя. Благодаря аватару существует возможность установить соответствие между внешностью пользователя и его данными, идентифицировать пользователя на других фотографиях. Также с помощью аватара и любой другой фотографии, опубликованной в социальной сети, есть возможность определить место, в котором сделана фотография исходя из геопозиции, а также время, в которое она была сделана исходя из свойств jpeg файла. Анализ всех фотографий, на которых присутствует поль-

зователь (как опубликованных самим пользователем, так и его друзьями) позволяет выяснить, какие места он чаще всего посещает и, в какое время это, обычно, происходит.

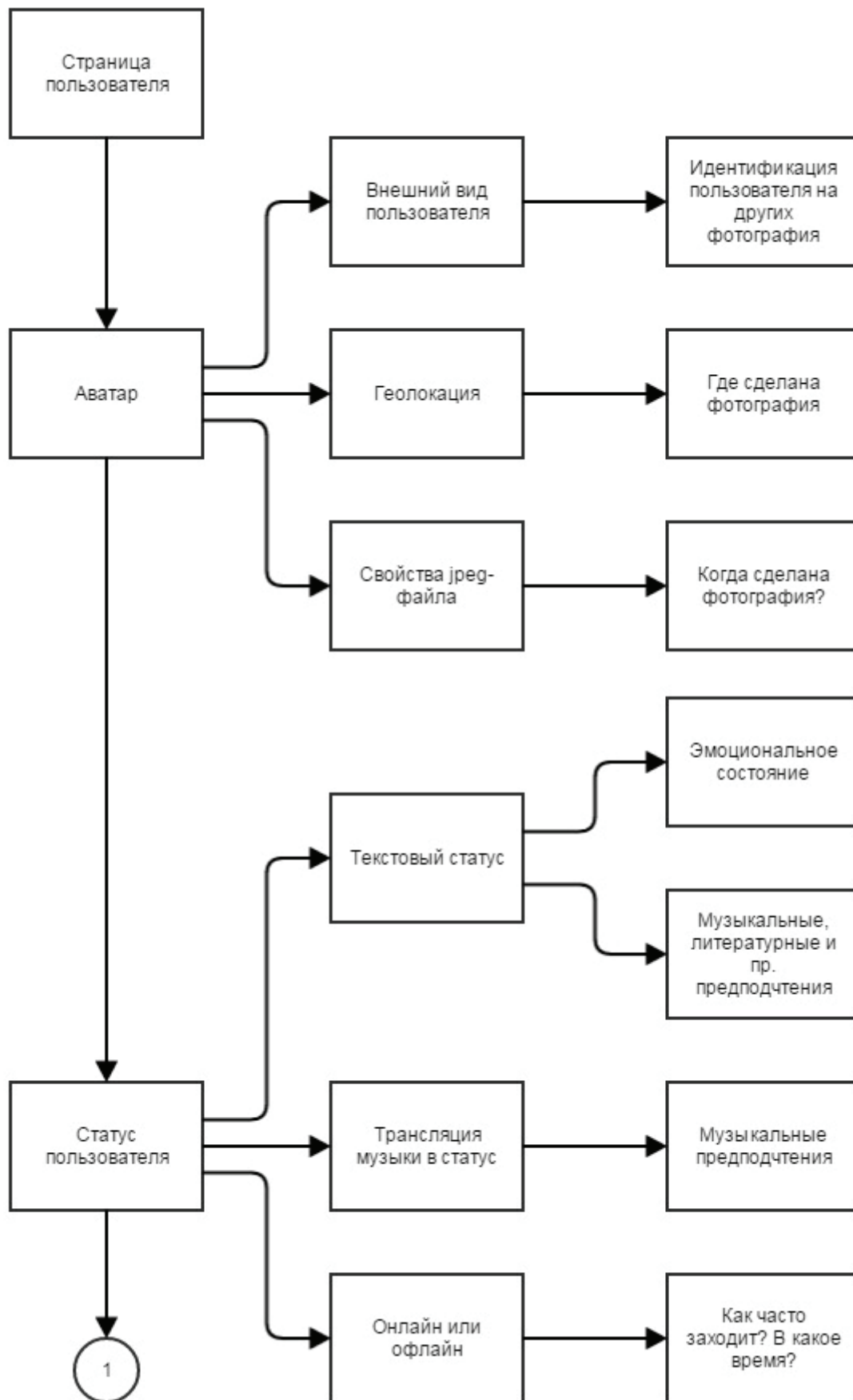
Статус пользователя. Исходя их статуса, можно судить о настроении пользователя. Нередко, пользователи публикуют в статусе цитаты из книг/фильмов/песен, что позволяет сделать вывод о литературных/кинематографических/музыкальных и прочих предпочтениях. Также в «ВКонтакте» есть возможность дублировать проигрываемые музыкальные композиции в статус, что также говорит о музыкальных предпочтениях.

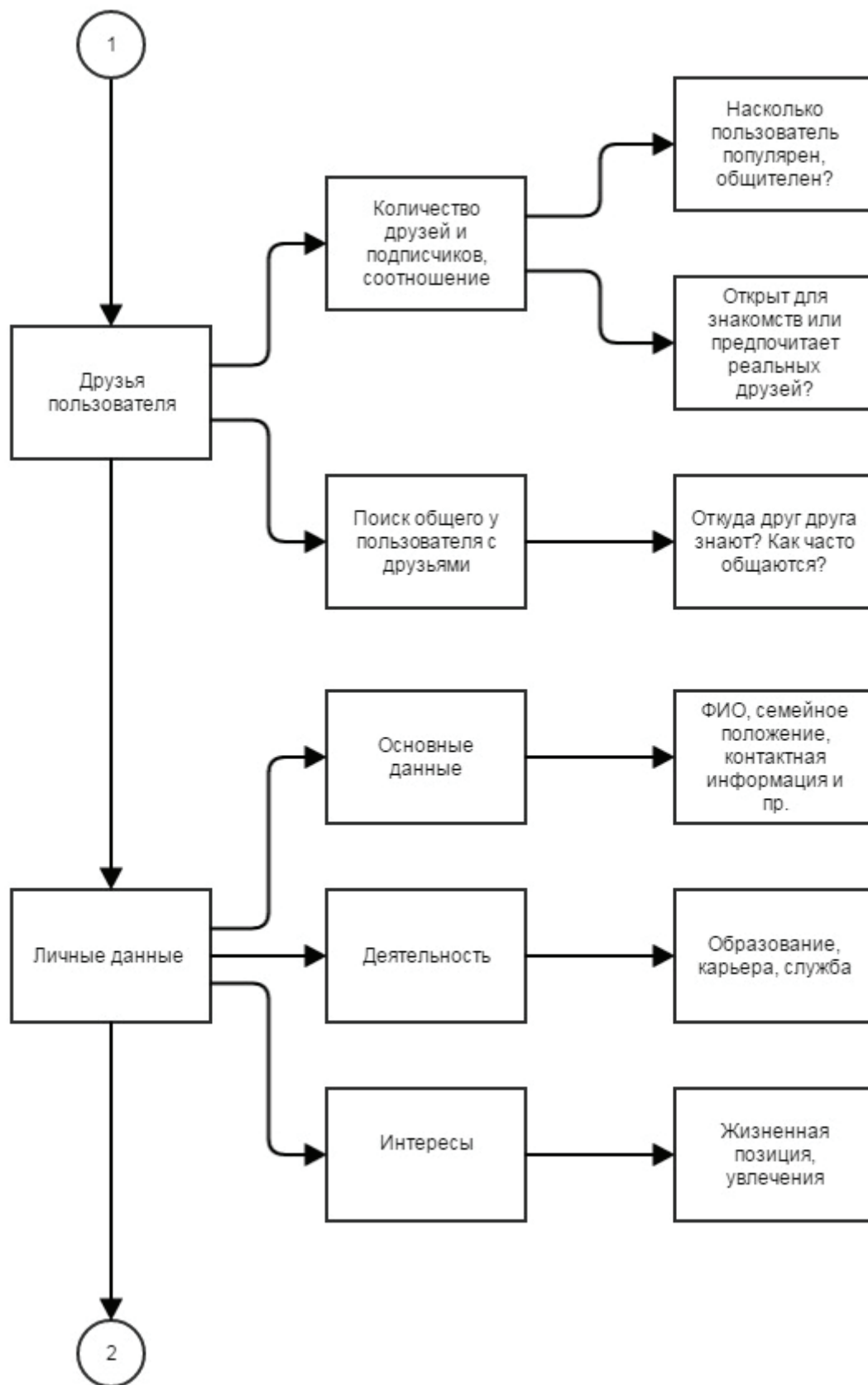
Друзья. Количество друзей свидетельствует об общительности или, напротив, отстраненности. Оценив соотношение друзей и подписчиков, можно сделать вывод о том, насколько пользователь общителен, предпочитает общаться с реальными друзьями или открыт для новых знакомств, насколько он популярен. Значительную часть информации, которую пользователь не указал на своей странице можно почерпнуть исходя из анализа страниц его друзей. Например, если большая часть друзей в графе родной город указало «Москва», вероятнее и сам пользователь проживает в Москве или если значительная часть друзей студенты, то логично предположить, что пользователь также является студентом и его возраст колеблется между 18 и 25 годами.

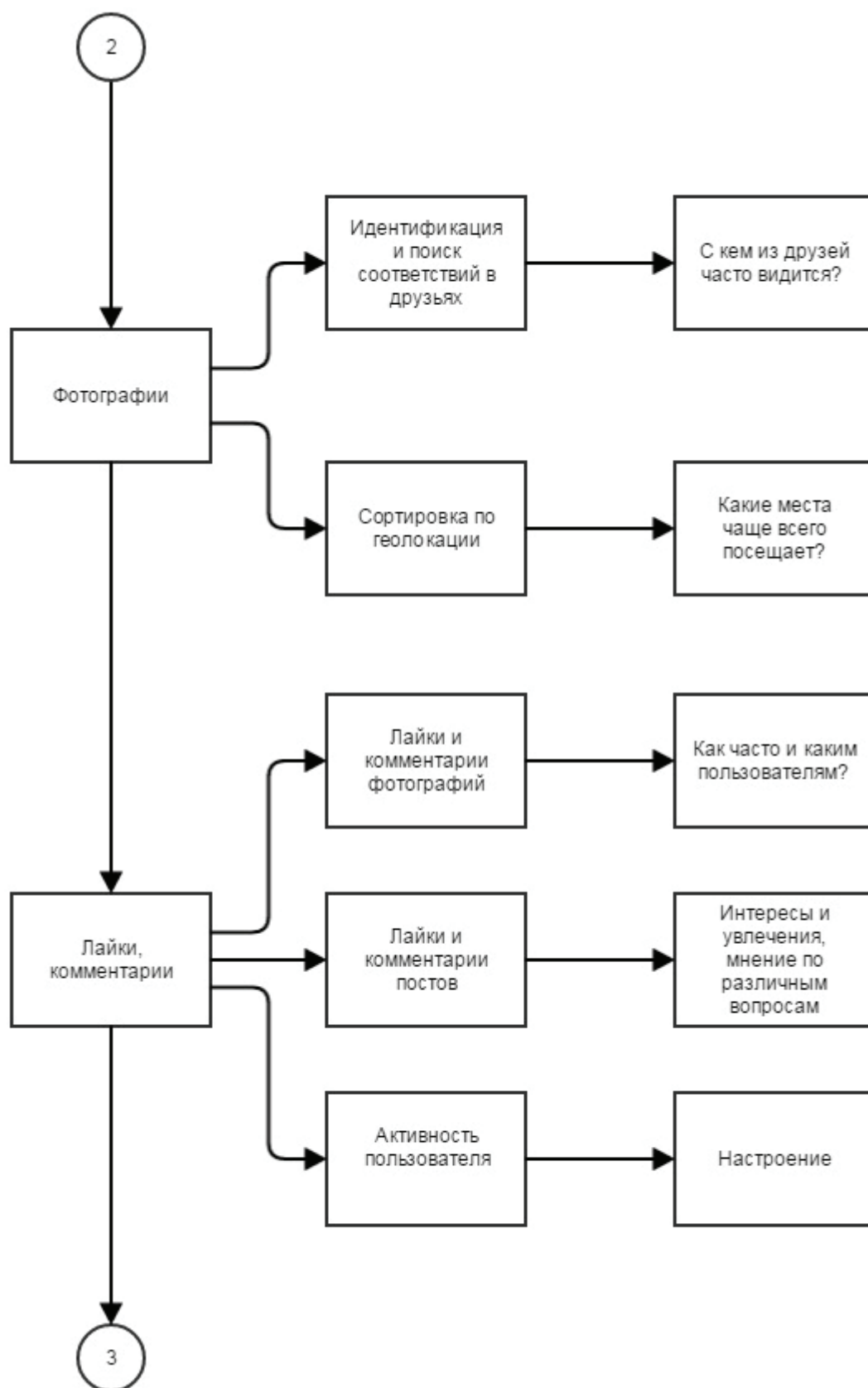
Информация о пользователе. В социальной сети «ВКонтакте» каждый пользователь может поместить личную информацию о себе на своей страничке. К личной информации относятся все текстовые данные, расположенные на странице. Информация о пользователе включает следующие разделы:

1. Основное (ФИО, пол, семейное положение и т.д.).
2. Контактная информация (страна, город, адрес, телефоны, почта, сайты, другие социальные сети и т.д.).
3. Интересы (деятельность, интересы, любимые фильмы/книги/музыка и т.д.).
4. Образование (школа, данные о среднем и высшем образовании и т.д.).
5. Карьера (место работы, должность и т.д.).
6. Служба (войсковая часть и т.д.).
7. Жизненная позиция (политические предпочтения, мировоззрение, отношение к алкоголю/курению и т.д.).

Все эти разделы заполняются при регистрации пользователя и могут быть отредактированы. Наполнение страницы данными облегчает идентификацию и поиск, однако, заполнять все пункты не обязательно, а также, есть возможность скрыть личную информации от других пользователей.







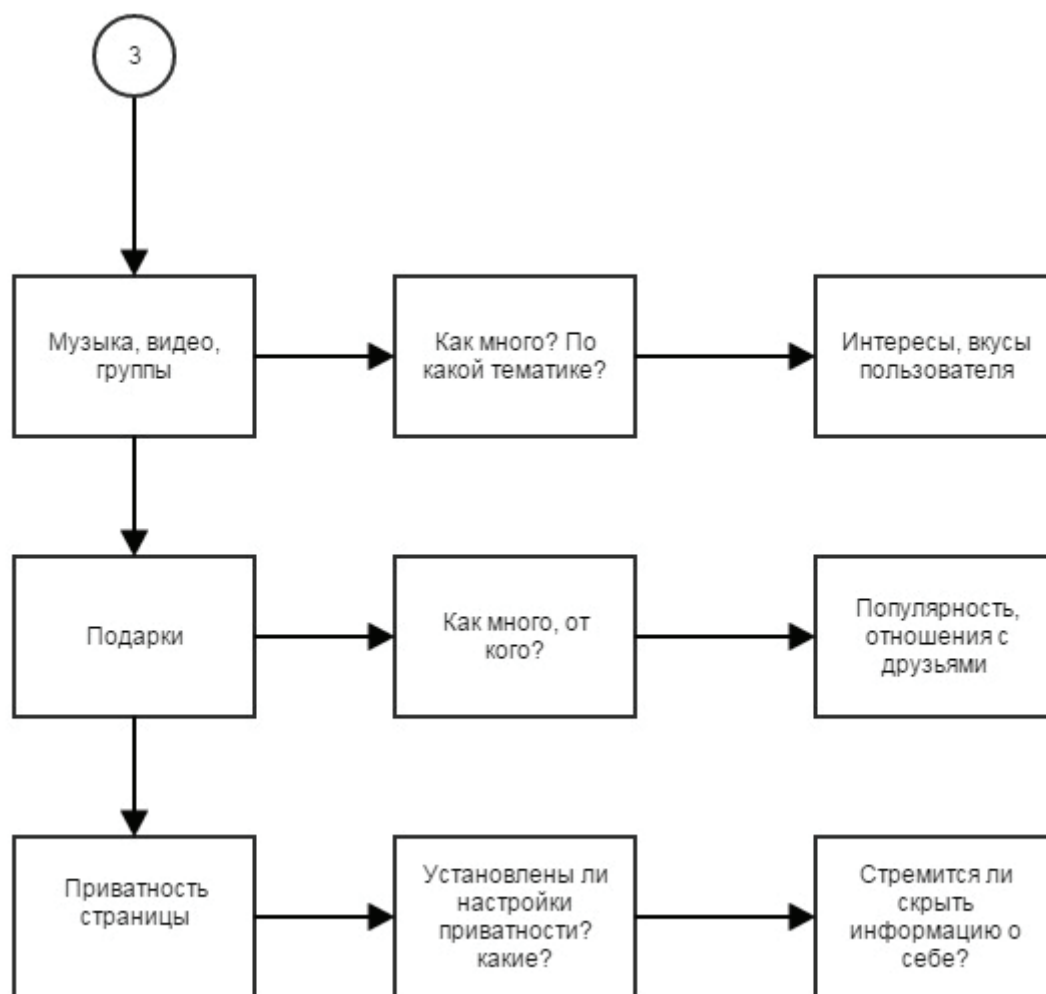


Рис. 1. Анализ страницы пользователя социальной сети «ВКонтакте»

Лайки, комментарии, подарки. Исследователи социальной сети Facebook выяснили, что наибольшую активность пользователь проявляет, находясь в хорошем расположении духа. Поэтому обильное проставление лайков свидетельствует о хорошем настроении пользователя в конкретный день, а отсутствие активности, по сравнению с другими днями, напротив, говорит о плохом настроении. Исходя из того на чьих фотографиях пользователь чаще всего проставляет лайки можно выяснить круг общения пользователя, его близких, друзей, родственников и пр. Помимо фотографий других пользователей лайки проставляются под постами, аудиозаписями, видеозаписями и пр. Это говорит о вкусах пользователя, а оставленные комментарии, помогут сделать вывод о его мнении на различные темы, а также о его стиле общения, степени грамотности и др.

Музыка, видео, группы. Аудио и видеозаписи, опубликованные пользователем, а также группы, в которых он состоит, могут свидетельствовать о его вкусах и увлечениях. Например, если человек подписан на 5 групп о компьютерных технологиях, вероятнее всего такой человек разбирается или учится разбираться в компьютерах, ув-

лекается техникой, и, возможно, работает в соответствующей области. Также группы могут подтверждать некоторую информацию, указанную на странице, например, если пользователь указал, что учиться в университете, наверняка, среди его групп можно найти группу, связанную с учебным заведением.

Приватность страницы. В «ВКонтакте» есть возможность ограничить доступ к своей странице, а также к конкретным данным, ограничить возможность отправки сообщений, оставление записей на стене и пр. В зависимости от того установлены ли настройки приватности можно судить о том, открыт ли пользователь для общения или менее общителен в соц. сетях. Стремится ли он скрыть информацию о себе или предоставляет всем открытый доступ к своей странице.

Исходя из представленной схемы анализа страницы пользователя, а также описания данного алгоритма, можно сделать вывод о том, что социальные сети содержат в себе огромное количество данных множества людей, активно их использующих. Анализ социальных сетей представляет собой эффективную систему для обнаружения и интерпретации личных данных, также ин-

тересы, предпочтений, убеждений, сделать вывод о взаимоотношениях с другими пользователями. Благодаря социальным сетям есть возможность получать инфор-

мацию в режиме реального времени, а данные, представленные на страницах, актуализируются за счёт стараний самих пользователей.

Литература:

1. Антон Коршунов. Задачи и методы определения атрибутов пользователей социальных сетей // Труды 15-й Всероссийской научной конференции «Электронные библиотеки: перспективные методы и технологии, электронные коллекции» — RCDL'2013 Второй источник.
2. Сергей Бартунов, Антон Коршунов. Идентификация пользователей социальных сетей в Интернет на основе социальных связей // Доклады Всероссийской научной конференции «Анализ изображений, сетей и текстов» — АИСТ'2012. Екатеринбург, 16–18 марта 2012 г.
3. Вэй Тан, Брайан Блейк, Иман Салех. Аналитика Больших Данных и социальные сети // Открытые системы. СУБД. — 2013. — № 8. — с. 37–41.

Концептуальная модель масштабируемого сервиса социальной сети

Усачёв Виктор Владимирович, магистрант;
Рагулин Пётр Григорьевич, кандидат технических наук, профессор
Дальневосточный федеральный университет

В статье анализируются возможности влияния архитектуры сервиса социальной сети на способности сервиса к масштабированию и адаптации в контексте постоянно меняющихся условий и бизнес-процессов. Актуальность темы обусловлена возрастающей значимостью социальных сетей для общества и экономики, а также технологическими проблемами разработки и сопровождения сервисов социальных сетей, о чём подробно рассказано. Автор рассматривает возможности микросервисной архитектуры применительно к концепции социальной сети и затрагивает такие важные аспекты как масштабирование, сложность внесения изменений, связанность компонентов системы.

Ключевые слова: концептуальная модель, сервис, социальная сеть, микросервисная архитектура, масштабируемость

В современном обществе существует такое понятие, как «социальная сеть», которое на бытовом уровне отождествляется со специализированным веб-сайтом в сети Интернет. Социальные сети играют роль особого пространства коммуникаций, в основе которого лежат социальные взаимоотношения. Структура этого виртуального пространства в значительной мере повторяет структуру коммуникационной среды реального мира: здесь присутствуют межличностное общение, профессиональные и тематические сообщества, проходят различные мероприятия и рекламные кампании. С позиции бизнеса социальные сети выполняют представительскую функцию, служат инструментом электронной коммерции и маркетинговых коммуникаций. Перспективы коммерческого применения социальных сетей становятся всё более привлекательными в связи с положительной динамикой вовлечения населения в Интернет-среду.

С каждым годом уровень проникновения Интернета в общество неуклонно растёт. Для России это утверждение также справедливо, о чём свидетельствуют исследования маркетинговой компании Comscore за 2011 год [1]. В отчёте компании отмечен интересный факт: среди рос-

сиян количество уникальных пользователей Интернета превысило аналогичный показатель для Германии. По результатам исследования [2], в 2015 году в России уровень проникновения Интернета среди населения в возрасте от 16 до 55 лет уже достиг отметки 70,4%, что немало, учитывая относительно «молодой» возраст самой сети Интернет. Логично предположить, что этот показатель сохранит положительную динамику, тем самым свидетельствуя о возрастающей вовлечённости населения в сферу Интернет-технологий и, в частности, социальные сети.

Согласно данным Федеральной службы государственной статистики на 2015 год среди населения в возрасте от 15 до 72 лет численность пользователей Сети на 100 человек составила 70,1% [3], причём по тем же данным 74,6% пользователей посещают социальные сети [4]. Столь высокие показатели отчасти могут быть объяснены ростом доступности мобильного Интернета. С ростом числа активных пользователей Интернета увеличилось среднее время присутствия пользователей в Сети: по данным 2014 года, в среднем пользователи проводят в Интернете 121 минуту в день (в 2013 году —

100 минут), причём на социальные сети приходится значительная доля этого времени — 36% [5].

С развитием социальных сетей связаны многие коммерческие аспекты, поскольку владельцы подобных сервисов как правило преследуют исключительно коммерческие цели. В основу бизнес-модели Интернет-ресурсов часто закладываются услуги по проведению рекламных кампаний, реже — маркетинговые услуги. Такая стратегия может быть эффективна при условии высокой посещаемости веб-ресурса и высоком уровне лояльности пользователей, но станет проблемой при снижении данных показателей, поэтому важно предусмотреть источники дополнительного дохода. Примером таких источников могут послужить продажа цифрового контента или взимание комиссии с продажи товаров и услуг средствами социальной сети.

Говоря о значении социальных сетей, важно принимать во внимание не только их возможности, но и актуальные проблемы. С одной стороны, социальные сети обладают огромным потенциалом коммерциализации ввиду того, что природа их происхождения имеет социальный характер. С другой стороны, владение социальной сетью сопряжено с большими финансовыми затратами, основная доля которых приходится на создание и сопровождение инфраструктуры информационной системы, обеспечивающей функционирование социальной сети.

Интернет — динамичная среда, и сценарий развития большинства Интернет-проектов практически невозможно точно предугадать, поэтому на ранних этапах развития им свойственна частая смена бизнес-модели, в соответствии с текущим направлением развития. Решения, положенные в основу проекта социальной сети, могут оказать сильное влияние на успех проекта в целом. В сфере Интернет-бизнеса часто возникает ситуация, когда конъюнктура рынка меняется, прогнозы не сбываются, и возникает необходимость адаптировать уже действующий проект к новым условиям. Например, необходимость в таких изменениях может быть продиктована предпочтениями пользователей или влиянием со стороны конкурентов, появлением новых технологий, дающих конкурентные преимущества. В реальных условиях решающими факторами являются скорость реакции и стоимость модернизации существующей системы, которые во многом зависят от особенностей технологического стека и архитектуры программного обеспечения социальной сети, а значит, такая модель развития событий должна быть учтена в концепции социальной сети, на этапе формирования требований и при проектировании.

Изменения в поведении пользователей социальной сети особенно непредсказуемы на ранних этапах развития: социальная природа подобных проектов часто является причиной лавинообразного роста посещаемости, что приводит к мгновенному росту нагрузки на инфраструктуру социальной сети, или причиной пересмотра принципов, заложенных в бизнес-логику сервиса. Для решения первой проблемы необходимо своевременное уве-

личение ресурсов системы (масштабирование), что не всегда возможно в короткие сроки и с приемлемыми затратами. Возможность решения второй проблемы обычно напрямую зависит от особенностей архитектуры и технологического стека проекта. Вопросы выбора технологического стека и архитектуры сервиса социальной сети являются актуальной проблемой.

В ходе эволюции принципов проектирования веб-приложений естественным этапом стало формирование концепции веб-сервиса — некоторого самостоятельного процесса на стороне сервера, который отвечает за выполнение одной простой задачи и обслуживает запросы от клиента. Обычно задача сервиса выбирается в соответствии с бизнес-процессом, с которым отождествляется веб-сервис. Клиентом может выступать другой веб-сервис, браузер пользователя или, например, мобильное приложение, для работы которого необходимо наличие серверной части. На развитие веб-сервисов в настоящее время сильное влияние оказывает ещё один современный подход к проектированию — микросервисная архитектура.

Если рассматривать разработку сложного программного обеспечения, можно выявить ряд недостатков, общих для монолитных систем [6]:

- компоненты системы сильно связаны между собой, что существенно затрудняет их модернизацию и сопровождение;

- сильная связь между компонентами приводит к затягиванию жизненного цикла разработки всего продукта, поскольку участники процесса разработки вынуждены тесно взаимодействовать между собой, согласовывая все действия во избежание ошибок;

- монолитная архитектура делает невозможным масштабирование системы или существенно его ограничивает.

Также нужно отметить, что структура команды разработки оказывает заметное влияние на продукт, возможно и обратное. Джон Хортон Конвей однажды заявил: «Организация которая разрабатывает систему... вынуждена делать систему, по структуре повторяющую структуру коммуникаций внутри организации», и это явление носит социологический характер. Сложность монолитных по своей структуре систем и вытекающие из этого технические и организационные ограничения привели к поискам новых подходов к разработке. Результатом стремления разработчиков упростить взаимодействие между частями цельной системы стало изменение самих принципов проектирования архитектуры: акцент сместился с бизнес-сущностей на бизнес-процессы, что соответствует процессному подходу и лучше отражает реальность. Вместо сложных монолитных систем появились многокомпонентные приложения, состоящие из отдельных, самостоятельных веб-сервисов, выполняющих по одной задаче; фактически, по своей архитектуре приложения стали распределёнными. Такая архитектура получила название микросервисной (англ. *microservice architecture*) [7]. Ми-

кроссервисная архитектура является развитием концепции сервисно-ориентированной архитектуры (SOA, англ. *service-oriented architecture*) [8], которая широко известна в среде корпоративных решений, но до недавнего времени не имела простых и платформонезависимых решений, что сдерживало её распространение.

Итогом пересмотра принципов проектирования стало решение ключевых проблем, присущих монолитным системам, а вместе с тем и проблемы в организации взаимодействия участников разработки. Команды разработчиков стали формироваться таким образом, чтобы структура команды и все коммуникации соответствовали архитектуре продукта. Такой подход к разработке приложений показывает высокую эффективность и позволяет:

- вести разработку единого продукта несколькими слабосвязанными командами;
- использовать разные технологические стеки для разных задач в рамках одного продукта;
- обеспечить разный жизненный цикл разработки разных частей приложения в зависимости от бизнес требований.

Применение микросервисной архитектуры наделяет систему следующими преимуществами:

1) технологическая разнородность. Совместно работающие микросервисы, составляющие единую систему, могут использовать внутри себя различные технологические стеки. Такой подход эффективен, поскольку позволяет для каждой задачи выбирать соответствующий инструментарий;

2) устойчивость. Микросервисная архитектура обеспечивает некоторую изоляцию компонентов системы. При отказе одного из компонентов система может сохранить работоспособность;

3) масштабирование. Можно расширить только те из микросервисов, которые требуют масштабирования.

Стандартизация механизмов взаимодействия пользователя с веб-сервисом, а также разных веб-сервисов между собой, привела к появлению нового архитектурного стиля взаимодействия приложений, который получил название REST (англ. *Representational State Transfer*, «передача состояния представления») [9].

Принципы REST представляют собой согласованный набор архитектурных ограничений, которые учитываются при проектировании распределённых систем [9, 10]:

- клиент-серверная архитектура;
- сервер не хранит клиентское состояние (*stateless-сервер*);
- кешируемость;
- многослойная структура;
- единый интерфейс;
- код по требованию (опционально).

Для обозначения того факта, что дизайн веб-сервиса соответствует принципам REST, служит термин «RESTful», а интерфейс взаимодействия обозначается как RESTful API. Ниже каждый из принципов рассматривается детально.

1. Клиент-серверная архитектура требует разделения системы на серверное приложение, реализующее бизнес-логику, и различные клиенты, использующие это приложение. При такой архитектуре структура сервера становится более простой и масштабируемой, а разработка клиентской части может вестись независимо.

2. *Stateless-сервер* ни в каком виде не хранит состояние клиента. Эта мера упрощает не только структуру сервера, но и его доработку, а также позволяет достичь лучшей стабильности сервера.

3. Кешируемость. Наличие чёткой системы кеширования запросов к серверу может заметно снизить нагрузку на него, повысив производительность.

4. Многослойная структура сервера скрывает от клиента наличие или отсутствие промежуточных компонентов, например, серверов кеширования и балансировки нагрузки.

5. Единый интерфейс выдвигает следующие требования:

- идентификация ресурсов,
- взаимодействие с ресурсами через представления,
- самодостаточные сообщения,
- гипермедиа как двигатель состояний приложения (принцип HATEOAS).

В рамках единого интерфейса каждый ресурс обладает уникальным идентификатором URI (*Uniform Resource Identifier*, «универсальный идентификатор ресурса»). Например: `/users/id0123` или `/post/123`. В качестве идентификатора рекомендуется использовать UUID (*Universally Unique Identifier*, «универсальный уникальный идентификатор»). UUID поддерживается большинством баз данных и поможет обеспечить кросс-системную уникальность идентификаторов.

Клиент взаимодействует с ресурсами через представление: когда клиент хранит представление ресурса, включая метаданные, то имеет достаточно данных для модификации или удаления ресурса. Также представлением можно считать URI ресурса.

Требование самодостаточности сообщений означает, что каждый ответ сервиса должен содержать всю информацию, необходимую для его обработки без обращения к другим ресурсам.

HATEOAS (*Hypermedia As The Engine Of Application State*, «гипермедиа как механизм управления состояниями приложения») означает, что в каждом полученном ответе помимо информации ресурса содержится ссылка на следующий ресурс, благодаря чему происходит смена состояний.

6. Код по требованию — это опциональный элемент структуры, который позволяет клиенту получать программный код для его исполнения на стороне клиента.

Проектирование приложений как RESTful-сервисов наделяет их следующими преимуществами [10]:

- надёжность (за счёт отказа от хранения состояния клиента);
- производительность (за счёт использования кэша);
- масштабируемость;

- прозрачность системы взаимодействия;
- простота интерфейсов;
- портативность компонентов;
- лёгкость внесения изменений;
- способность эволюционировать, приспосабливаясь к новым требованиям.

Указанные преимущества очень ценны сами по себе, но в случае сложных приложений, таких как интернет-магазины, электронные торговые площадки, аналитические и поисковые системы, разного рода корпоративные системы, REST значительно упрощает их архитектуру и приводит к высокой производительности и эффективности. Взаимодействие компонентов в REST простое, прозрачное и стандартизированное, в то время как в монолитных приложениях взаимодействие сложное и разнородное, компоненты сильно связаны. Таким образом, для систем, ориентированных на Интернет, REST — ключ к эффективности.

Обобщая сказанное выше, можно заключить, что построение сервиса социальной сети с применением микросервисной архитектуры и организация взаимодействия микросервисов посредством REST API с высокой долей вероятности позволят реализовать легко масштабируемый и простой в модернизации сервис социальной сети. Исходя из типичного набора функциональных возможностей социальных сетей, всё приложение можно рассматривать как набор взаимодействующих компонентов, каждый из которых отвечает за определённую возможность. В ходе анализа предметной области были выяв-

лены следующие функциональные блоки будущего сервиса:

- аутентификация и авторизация;
- политика безопасности;
- аккаунты и профили пользователей;
- роли пользователей;
- публикационная активность;
- хостинг медиа-контента;
- система аналитики.

Некоторые из представленных аспектов между собой тесно взаимосвязаны и могут рассматриваться только в совокупности, поэтому логично объединить их реализации в программных модулях. Основные компоненты приложения могут быть вынесены в отдельные модули и реализованы как микросервисы, взаимодействующие через RESTful API. Вспомогательные компоненты имеет смысл объединить с теми модулями, в которых их роль наиболее очевидна и логична.

Применение REST и микросервисной архитектуры в данном случае оправдано и позволит решить многие из поставленных задач. Такой подход к разработке даст следующие преимущества:

- разработка всех сервисов приложения может быть распределена между слабосвязанными командами;
- для каждой задачи могут быть использованы наиболее подходящие технологические стеки;
- жизненный цикл разработки разных частей приложения может быть разным, что позволяет быстро адаптировать приложение к новым бизнес-требованиям;

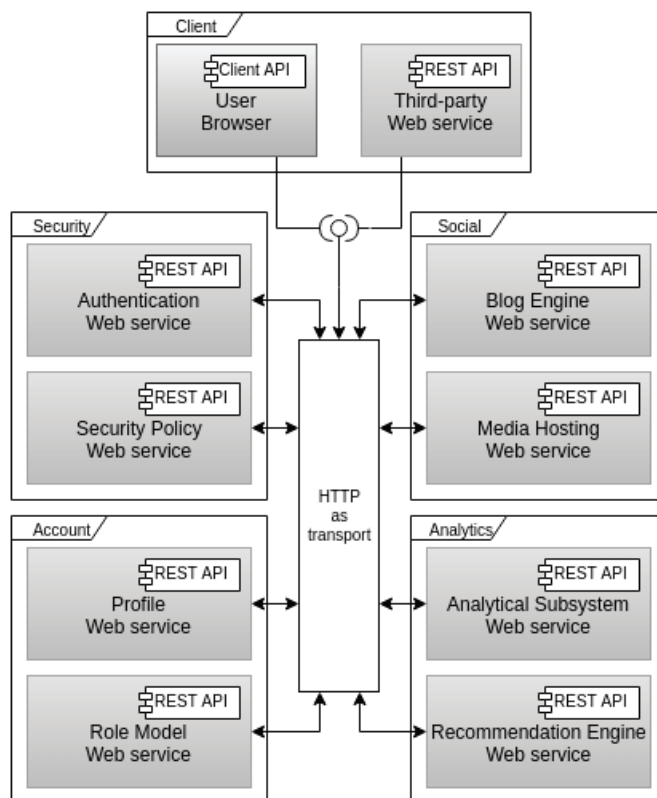


Рис. 1. Архитектура распределённого REST-приложения

— пользовательский интерфейс может быть реализован отдельно и во множестве вариантов, если приложение сделать как одностраничное (англ. Single Page Application, SPA).

Планируемая архитектура приложения показана на рисунке 1. Она учитывает основные требования к сервису и позволит разработать веб-приложение с минимальными затратами.

Литература:

1. Comscore. Comscore releases overview of european internet usage in September 2011. [Электронный ресурс]. 2011. — Режим доступа: http://www.comscore.com/Press_Events/Press_Releases/2011/11/comScore_Releases_Overview_of_European_Internet_Usage_in_September_2011 (дата обращения: 04.05.2016).
2. GfK. «Проникновение Интернета в России: Итоги 2015 года» [Электронный ресурс]. — Режим доступа: <http://www.gfk.com/ru/insaity/press-release/issledovanie-gfk-za-2015-god-internet-auditorija-v-rossii-uvlechilas-eshche-na-4-mln-chelovek/> (дата обращения: 28.04.2016).
3. Использование сети Интернет городским и сельским населением в возрасте 15–72 лет, по Российской Федерации (ноябрь-декабрь 2015 года): Федеральная служба государственной статистики [Электронный ресурс]. — Режим доступа: http://www.gks.ru/free_doc/new_site/business/it/fed_nabl-croc/PublishData/Reports/Files/2015/3.1.xlsx (Дата обращения 04.06.2016).
4. Доля населения, использовавшего сеть Интернет, по целям его использования, типам поселения и полу, по Российской Федерации (ноябрь-декабрь 2015 года): Федеральная служба государственной статистики [Электронный ресурс]. — Режим доступа: http://www.gks.ru/free_doc/new_site/business/it/fed_nabl-croc/PublishData/Reports/Files/2015/3.11.xlsx (Дата обращения 04.06.2016).
5. Интернет в России: Состояние, тенденции и перспективы развития. Отчет по итогам 18-го российского интернет форума РИФ+КИБ 2014 — М., 2014.
6. Михайлов, В. Microservices. Как правильно делать и когда применять? / Habrahabr. М.: Тематические медиа, 2016. Режим доступа: <https://habrahabr.ru/company/dataart/blog/280083> (дата обращения: 15.02.2016).
7. Ньюмен, С. Создание микросервисов. — СПб.: Питер, 2016. — 304 с.: ил. — Серия «Бест-селлеры O'Reilly». — ISBN 978-5-496-02011-4.
8. Коптелов, А. Системы управления процессами и SOA. / КомпьютерПресс. — М.: КомпьютерПресс, 2008. URL: <http://compress.ru/article.aspx?id=19374> (дата обращения: 17.09.2015).
9. REST // Википедия. [2016]. Дата обновления: 07.04.2016. URL: <http://ru.wikipedia.org/?oldid=77621452> (дата обращения: 23.03.2016).
10. Thomas, E. SOA with REST. // Thomas Erl, Benjamin Carlyle, Cesare Pautasso, Raj Balasubramanian. — Prentice Hall, 2013.

ТЕХНИЧЕСКИЕ НАУКИ

Роль сетевой модели OSI в построении промышленных сетей

Александров Андрей Андреевич, студент

Национальный исследовательский университет «Московский институт электронной техники»

Международной организацией по стандартизации (International Standardization Organization — ISO) было разработана сетевая модель OSI (англ. Open System Interconnection Reference Model — модель взаимодействия открытых систем), которая решает проблему стандартизации подключений различного оборудования.

Сетевая модель OSI — абстрактная модель для разработки сетевых протоколов и сетевых коммуникаций, единого представления данных в сетях с различными устройствами и программным обеспечением. Данная модель, реализующая уровневый подход к сети, определяет процедуры и правила передачи данных в различных сетевых средах. Она представляет из себя семиуровневую сетевую иерархию, уровни которой, изображенные на рисунке, выполняют определенные задачи в процессе передачи информации по сети и обслуживают различные части процесса области взаимодействия открытых систем.

Модель ISO определяет цель эталонной модели следующим образом: «Обеспечение общей базы для координации разработки стандартов, предназначенных для взаимодействия систем, с учетом того, что в перспективе существующие стандарты будут описаны в рамках общей эталонной модели».

По существу, OSI является управляющей структурой, которая упрощает передачу данных в иерархической системе из семи уровней. Каждый уровень имеет определенное назначение и взаимодействует с соседними верхним и нижнем уровнями. Стандарты определяются для каждого уровня таким образом, чтобы обеспечить некоторую гибкость, позволяя проектировщикам системы разрабатывать независимые уровни протоколов. Любые два или более уровней вместе образуют так называемый стек протоколов.

Модель OSI, показанная в таблице 1, полезна в обеспечении универсальной основы для всех коммуникационных систем. Однако она не определяет реальный протокол, который должен использоваться на каждом уровне. Предполагается, что группы производителей в различных отраслях промышленности, сотрудничая, выработают стандарты программного и аппаратного обеспечения, наиболее подходящие именно для их отрасли. Те же, кто ищет единый базис для своих специфических требований к обмену данными, с энтузиазмом воспользовались моделью OSI и используют ее в качестве основы для выработки своих промышленных стандартов, таких как Fieldbus и HART.

Таблица 1. Семиуровневая сетевая модель OSI

Номер уровня	Название	Назначение	Пример
1	Физический	Электрические и механические свойства системы	Стандарт 10BaseT Стандарт 10Base2/10Base5 Стандарт 10BaseF
2	Канальный	Передача по физическому адресу по сети. Создание кадров данных и коррекция ошибок	Ethernet Token Ring Протокол Carrier Sense Multiple Access / Collision Detection — Сети с множественным доступом с контролем несущей и обнаружением коллизий
3	Сетевой	Определение пути передачи сообщения от одной сети к другой	Internet Protocol Internetwork Packet Exchange Datagram Delivery Protocol

4	Транспортный	Определение канала передачи сообщений от одного прикладного процесса к другому	Transmission Control Protocol Sequenced Packet Exchange Apple Talk Transaction Protocol — Протокол управления транзакциями Apple Talk
5	Сеансовый	Организация и синхронизация передачи данных	Сервисы печати Разделяемые файловые системы
6	Представительный	Форматирование или кодирование/декодирование данных	Код American Standard Code for Information Interchange — Американский стандартный код для обмена информацией
7	Прикладной	Передача данных, обмен сообщениями	Электронная почта Удаленный доступ

Важно понимать, что эталонная модель OSI является не протоколом или набором правил написания протокола, а общим каркасом, определяющим протоколы. Построение модели OSI четко и конкретно определяет функции или службы, которые должны обеспечиваться каждым из семи уровней (или слоев).

Взаимодействие уровней сетевой модели OSI можно поделить на две модели взаимодействия:

- Горизонтальная модель взаимодействия, которая обеспечивает связь программ и процессов на различных устройствах;
- Вертикальная модель взаимодействия, которая обеспечивает работу между сетевыми уровнями на одном устройстве.

При передаче данных происходит взаимодействие между сетевыми уровнями, изображенными на рисунке 1, устройства-отправителя и устройства-получателя. Такая

связь именуется логической или виртуальной, хотя на самом деле взаимодействие происходит между смежными уровнями одного устройства. Устройство-отправитель получает запрос, на основе которого формируется сообщение стандартного формата, имеющее заголовок определенного уровня и поля данных.

Сообщение проходит по стеку между уровнями, где на основании информации, полученной из заголовка верхнего уровня, выполняются определенные функции и добавляется заголовок текущего уровня, содержащий информацию для того же уровня устройства-получателя. Сообщение, передающееся вниз, «обрастает» заголовками всех уровней и принимаются по физической среде устройством-получателем. И снова, сообщение последовательно перемещается вверх до того же уровня, с которого это сообщение было отправлено устройством-отправителем.

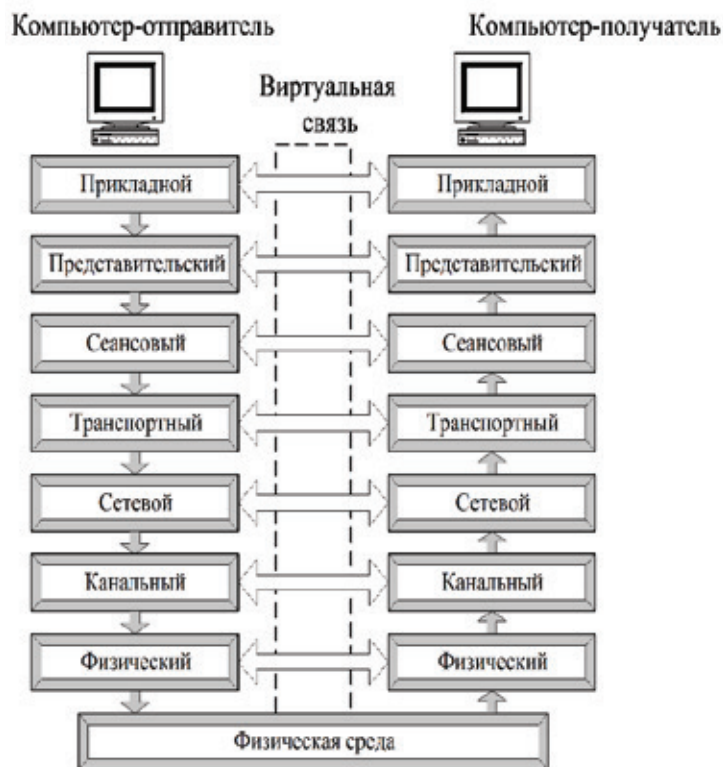


Рис. 1. Взаимодействие устройств в модели OSI

Роль каждого уровня в этой сетевой модели — анализ и обработка заголовков, несущих информацию своего уровня, относительно которой происходит выполнение соответствующих уровню функций, а затем заголовков удаляется и сообщение передается на уровень выше.

При горизонтальной модели взаимодействия двум устройствам требуется общий протокол, реализующий обмен данными между ними. В вертикальной модели соседние уровни обмениваются информацией с использованием API (Application Programming Interface) — интерфейса прикладных программ.

Прикладной уровень является самым верхним уровнем OSI/RM. Он отвечает за предоставление приложениям доступа к сети. Примерами задач прикладного уровня являются передача файлов, электронная почта и сетевое управление.

Для выполнения этих задач прикладной уровень передает запросы программ и данных на уровень представления, который отвечает за кодирование данных прикладного уровня и их перевод в соответствующую форму.

Представительный уровень преобразует данные в форму, подходящую для транспортного уровня. Он преобразует формат и синтаксис данных, обеспечиваемых приложениями, и производит кодирование и сжатие данных.

Сеансовый уровень управляет взаимодействием между устройствами. Он определяет использование программного обеспечения, которое позволяет обращаться к другому устройству по имени, а не с помощью двоичного адреса (логическая адресация). Он также обеспечивает восстановление прерванных сеансов связи.

Транспортный уровень определяет:

- управление связью между двумя концами системы;
- передачу данных с согласованным качеством;
- учет и корректное сегментирование пакетов в больших сообщениях, что улучшает надежность передачи данных.

Сетевой уровень определяет:

- маршрутизацию пакетов информации по сети;
- регламентирование и передачу сообщений состоянием другим устройствам сети;
- разбиение больших пакетов, принимаемых от транспортного уровня, на маленькие пакеты;
- обеспечение прохождения кадров через нижерасположенную сеть.

Из-за определенной направленности промышленной сети из всех семи уровней сетевой модели OSI используются только два: физический и канальный.

Физический уровень является первым уровнем сетевой модели OSI. Он предназначен для объединения с совокупностью физической среды, аппаратных и программных средств, обеспечивающих передачу сигналов между системами. Первый уровень модели OSI определяет метод передачи данных, представленных в бинарном виде, от одного устройства к другому. Единицей измерения, используемой на данном уровне, является бит, то есть физический уровень осуществляет передачу потока

битов по физической среде с использованием соответствующего интерфейса.

Физический уровень состоит из подуровня стыковки со средой и подуровня преобразования передачи. Подуровень стыковки обеспечивает совмещение потока данных с используемым физическим каналом связи. Подуровень преобразования передачи осуществляет преобразования, связанные с применяемыми протоколами. Физический уровень описывает процедуры передачи сигналов в канал и получения их из канала, а также обеспечивает физический интерфейс с каналом передачи данных. На данном уровне определяются различные параметры (электрические, механические, функциональные, процедурные) для физической связи в системах. Преобразование данных в оптические или электрические сигналы происходит от вышележащего канального уровня. Эти сигналы отправляются через среду передачи на приемный узел.

Физический уровень выполняет следующие функции:

1. Идентификация каналов.
2. Прослушивание каналов.
3. Установление и разъединение физических соединений.
4. Передачи и прием сигналов в последовательном канале.
5. Оповещение о появлении неисправностей и отказов в сети.

На физическом уровне происходит выявление определенного класса неисправностей, затрудняющих работу сети: оповещение о столкновении кадров, отправленных сразу несколькими системами, обрыв канала, отключение питания, потеря механического контакта. Предоставляемые канальному уровню виды сервиса, определяются протоколами физического уровня. При подключении к одному каналу, в котором передача сигналов разрешается только одной из групп систем, используется прослушивание каналов, позволяющее определить, свободен ли он для передачи.

Во всех устройствах, подключенных к сети, реализуются функции физического уровня. За выполнением функций физического уровня со стороны компьютера отвечает сетевой адаптер. Единственным типом оборудования, работающим только на физическом уровне, является повторитель.

Физический уровень устанавливает такие виды сред передачи данных как витая пара коаксиальный, оптоволоконно, спутниковый канал передачи данных.

Канальный уровень является вторым уровнем сетевой модели OSI. Он определяет правила получения доступа к среде передачи данных, логическую топологию сети, решает вопросы, связанные с адресацией физических устройств в рамках логической сети и управлением передачей информации (синхронизация передачи и сервис соединений) между сетевыми устройствами.

Единицей измерения канального уровня являются кадры (frame). Кадры — это структура, в которой размещены данные. Канальный уровень осуществляет передачу

кадров между устройствами, подключенных к одному сетевому сегменту, от сетевого уровня к физическому. При передаче бит на физическом уровне не всегда учитывается, что в некоторых сетях линия связи используется попеременно несколькими устройствами сразу. Поэтому проверка доступности среды передачи данных является

одной из задач канального уровня, как и реализация методов поиска и коррекции ошибок.

Помещая определенную последовательность бит, в начало и конец каждого кадра, изображенного на рисунке, канальный уровень обеспечивает корректность передачи каждого кадра, показанного на рисунке 2.



Рис. 2. Структура кадра канального уровня

Далее, суммируя определенным образом байты и добавляя контрольную сумму, кадры доходят до получателя, где снова вычисляется контрольная сумма полученных данных и сравнивается результат с суммой из кадра. При совпадении сумм, кадр считается правильным, в противном случае, регистрируется ошибка.

Канальным уровнем определяется доступ к среде и управление передачей данных по каналу. При получении кадров из них формируются переданные блоки данных, размер которых зависит от качества канала и способа пе-

редачи. При больших размерах блоки данных делятся на кадры и передаются в виде последовательностей.

Канальный уровень выполняет следующие функции:

1. Установление, расторжение и управление канальных соединений и идентификация их портов.
2. Обнаружение и коррекция ошибок.
3. Организация передачи кадров.
4. Передача данных, закодированных различными способами, по логическим каналам.
5. Управление потоками данных.

Литература:

1. Парк Дж., Маккей С., Райт Э. Передача данных в системах контроля и управления. / Парк Дж., Маккей С., Райт Э; [перевод с англ. В.В. Савельева]. — М.: ООО «Группа ИДТ», 2007. — 480 с.
2. Олифер, В.Г., Олифер Н.А. Компьютерные сети. Принципы, технологии, протоколы. / Олифер В.Г., Олифер Н.А. — СПб.: Питер, 2010—944 с.

Метод расчета активного сопротивления цилиндрического провода с учетом поверхностного эффекта

Аронов Леонид Вячеславович, инженер
Рязанский государственный радиотехнический университет

В данной статье рассматривается вопрос влияния высших гармоник тока на активное сопротивление цилиндрического провода в диапазоне частот от 50 до 2000 Гц. Выведена расчетная формула для активного сопротивления цилиндрического провода на произвольной частоте. Проанализированы закономерности изменения активного сопротивления медных и алюминиевых проводов в зависимости от частоты и площади поперечного сечения.

Ключевые слова: *поверхностный эффект, несинусоидальность, активное сопротивление медного провода, активное сопротивление алюминиевого провода*

This article has shown the problem of high-harmonics influence on cylindrical wire resistance according to the frequency range from 50 to 2000 Hz. Cylindrical wire resistance estimation for any frequency has been deduced. We have also analyzed cylindrical wire resistance dependence of copper and aluminum wires due to the frequency and cross section area.

Keywords: skin effect, high-harmonics, copper wire resistance, aluminum wire resistance

В Российской Федерации системы электроснабжения общего назначения работают на фиксированной частоте 50 Гц. Кроме основной гармоники, в сетях присутствуют также высшие гармоники токов и напряжений [1]. Исследования указывают на то, что при наличии в сети нелинейных нагрузок, наибольшим искажениям подвергается форма кривой тока, в то время как кривая напряжения практически не изменяется [5, 6]. Это объясняется тем, что реальные генераторы по своим свойствам близки к идеальным источникам ЭДС. Высшие гармоники тока приводят к дополнительному нагреву проводников линии электропередачи, в результате чего, с одной стороны возникают дополнительные потери мощности, с другой происходит ускоренный износ изоляции, изоляторов, растяжение проводов и другие негативные последствия [2]. По мере увеличения частоты, начинает проявляться поверхностный эффект, приводящий к перераспределению плотности тока в сечении проводника, от центра к поверхности, в результате изменяется его сопротивление, что в свою очередь влияет на потери мощности и энергии в линиях электропередачи электрической распределительной сети.

Задача данного исследования: определить степень изменения плотности тока в сечении провода и его сопротивления в зависимости от частоты гармоники тока протекающего по проводнику. Это позволит более точно рассчитать дополнительные потери мощности на нагрев проводов, обусловленные несинусоидальностью питающего напряжения электрической распределительной сети.

Провод можно приближенно представить в виде цилиндрического проводника, длина которого многократно превышает радиус. Для нахождения плотности тока используют уравнения Максвелла, решение которых производится в цилиндрической системе координат [3]. В результате выражения для плотности тока и напряженности магнитного поля в любой точке сечения цилиндрического проводника:

$$\dot{\delta} = \frac{k \cdot \dot{I} \cdot J_0(k \cdot r)}{2 \cdot \pi \cdot a \cdot J_1(k \cdot a)} \tag{1}$$

$$\dot{H} = \frac{\dot{I} \cdot J_1(k \cdot r)}{2 \cdot \pi \cdot a \cdot J_1(k \cdot a)} \tag{2}$$

где $\dot{\delta}$ — комплексная плотность тока, A/m^2 ; \dot{H} — напряженность магнитного поля, A/m ; \dot{I} — комплексный ток, A ; J_0 — функция Бесселя 1-го рода 0-го порядка; J_1 — функция Бесселя 1-го рода 1-го порядка; r — радиус текущей поверхности тока в проводе, m ; a — радиус провода, m ; $k = \sqrt{-j \cdot \sqrt{\omega \cdot \sigma \cdot \mu_a}}$ — комплексное волновое число, m^{-1} ; $\omega = 2 \cdot \pi \cdot f$ — круговая частота, rad/c ; f — циклическая частота, $Гц$; $\mu_a = \mu \cdot \mu_0$ — абсолютная магнитная проницаемость, $Гн/м$; μ — относительная магнитная проницаемость; $\mu_0 = 4 \cdot \pi \cdot 10^{-7} Гн/м$ — магнитная постоянная.

Формулы (1), (2) не учитывают «эффект близости», т. к. для этого необходимо точно знать пространственное расположение проводников. При расстоянии между проводами многократно превышающем радиус и невысоких значениях тока «эффектом близости» можно пренебречь.

Мощность рассеиваемую цилиндрическим проводом определяют по закону Джоуля-Ленца в комплексной форме:

$$\underline{S} = I^2 \cdot \underline{Z} \tag{3}$$

С другой стороны рассеиваемую мощность находим, используя теорему Умова-Пойнтинга [3]:

$$\underline{S} = \oint_s [\dot{\vec{E}} \cdot \dot{\vec{H}}^*] ds \tag{4}$$

Приравнивая выражения (3) и (4) получаем:

$$I^2 \cdot \underline{Z} = \oint_s [\dot{\vec{E}} \cdot \dot{\vec{H}}^*] ds \tag{5}$$

В результате комплексное сопротивление выражаем формулой:

$$\underline{Z} = \frac{\oint_s [\dot{\vec{E}} \cdot \dot{\vec{H}}^*] ds}{I^2} \tag{6}$$

Вектор напряженности электрического поля $\dot{\vec{E}}$ вычисляем по формуле:

$$\dot{\vec{E}} = \frac{\dot{\delta}}{\sigma} = \frac{k \cdot \dot{I} \cdot J_0(k \cdot r)}{2 \cdot \pi \cdot \sigma \cdot a \cdot J_1(k \cdot a)} \tag{7}$$

где σ — удельная проводимость, $См/м$.

Учитывая, что площадь боковой поверхности цилиндрического проводника равна $S=2\cdot\pi\cdot a\cdot l$ и радиус поверхности равен радиусу провода $r=a$ получаем:

$$\underline{Z} = \frac{\oint[\dot{E} \cdot \dot{H}^*] ds}{I^2} = \frac{k \cdot \dot{I} \cdot J_0(k \cdot a)}{2 \cdot \pi \cdot \sigma \cdot a \cdot J_1(k \cdot a)} \cdot \frac{\dot{I} \cdot J_1(k \cdot a)}{2 \cdot \pi \cdot a \cdot J_1(k \cdot a)} \cdot 2\pi \cdot a \cdot l = \frac{k \cdot J_0(k \cdot a) \cdot J_1(k \cdot a)}{2 \cdot \pi \cdot \sigma \cdot a \cdot J_1^2(k \cdot a)} \cdot l = \frac{k \cdot J_0(k \cdot a)}{2 \cdot \pi \cdot \sigma \cdot a \cdot J_1(k \cdot a)} \cdot l,$$

где l — длина проводника, м.

Таким образом, окончательное выражение для комплексного сопротивления цилиндрического провода приняло вид:

$$\underline{Z} = \frac{k \cdot J_0(k \cdot a)}{2 \cdot \pi \cdot \sigma \cdot a \cdot J_1(k \cdot a)} \cdot l.$$

Активное сопротивление при этом определяем, как действительную часть полного комплексного сопротивления:

$$R = \text{Re}(\underline{Z}) = \frac{l}{2 \cdot \pi \cdot \sigma \cdot a} \cdot \text{Re}\left(\frac{k \cdot J_0(k \cdot a)}{J_1(k \cdot a)}\right). \tag{8}$$

Рассчитаем сопротивления медных проводов и алюминиевых проводов марок А, АКП, АН, АНКП, АЖ, АЖКП. Сечения выберем самые распространенные: 16, 25, 35, 50, 70, 95, 120, 150, 185 мм² [4], хотя каких-либо ограничений по сечению проводов нет. Единственное условие данного метода: длина должна быть много больше радиуса $l \gg a$. Удельная проводимость меди $\sigma_{Cu} = 56 \cdot 10^6$ См/м, а удельная проводимость алюминия $\sigma_{Al} = 37 \cdot 10^6$ См/м. Относительная магнитная проницаемость обоих материалов приблизительно равна $\mu = 1$ (медь является диамагнетиком, а алюминий парамагнетиком). Радиус провода выражаем из формулы площади круга:

$$a = \sqrt{\frac{s}{\pi}},$$

где s — площадь сечения провода, мм².

Расчета активного сопротивления медных проводов различных сечений, выполнен в диапазоне частот 0–2 кГц (рисунок 1.). Аналогичный расчет проведен для алюминиевых проводов (рисунок 2.). Выбор диапазона обосновывается тем, что согласно ГОСТ Р54149–2010, гармоники нормируются до 40-й включительно, что соответствует наибольшему значению частоты 2кГц [1].

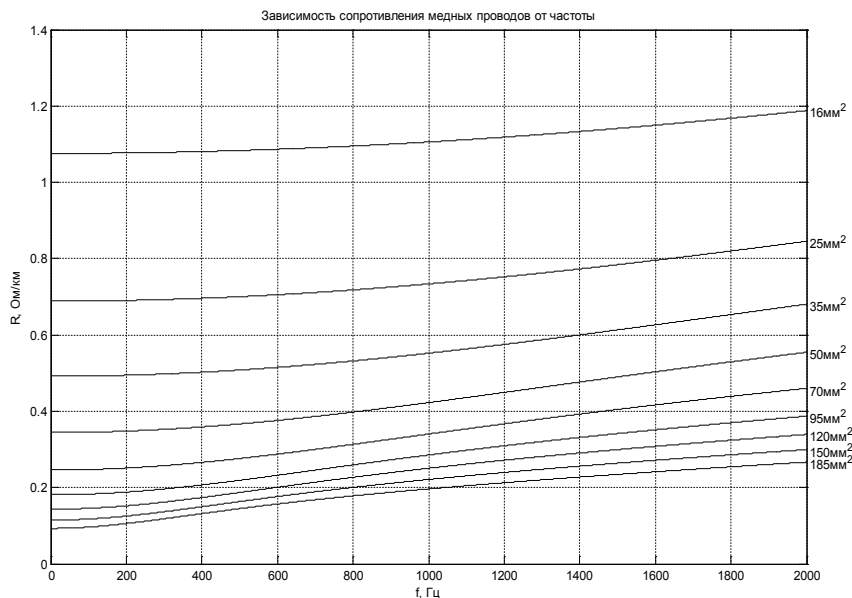


Рис. 1. Зависимость активного сопротивления медных проводов различных сечений от частоты

Семейства графиков (рисунки 1, 2) показывает зависимость погонного сопротивления проводов как от их сечения, так и от частоты. В полосе частот от 0 до 2000 Гц (рисунок 1.) величина возрастания сопротивления от 10,4% до 183,9%, в зависимости от сечения провода. Причем наибольший, в процентном отношении, прирост сопротивления наблюдается на больших сечениях. Например, при сечении 16 мм² погонное сопротивление увеличивается с 1,08 до 1,19 Ом/км, что составляет 10,4%, а при наибольшем сечении 185 мм² погонное сопротивление увеличивается с 0,09 до 0,27 Ом/км, что в пересчете на проценты составляет 183,9%. Это объясняется большей неравномерностью плотности тока при большем радиусе сечений проводов.

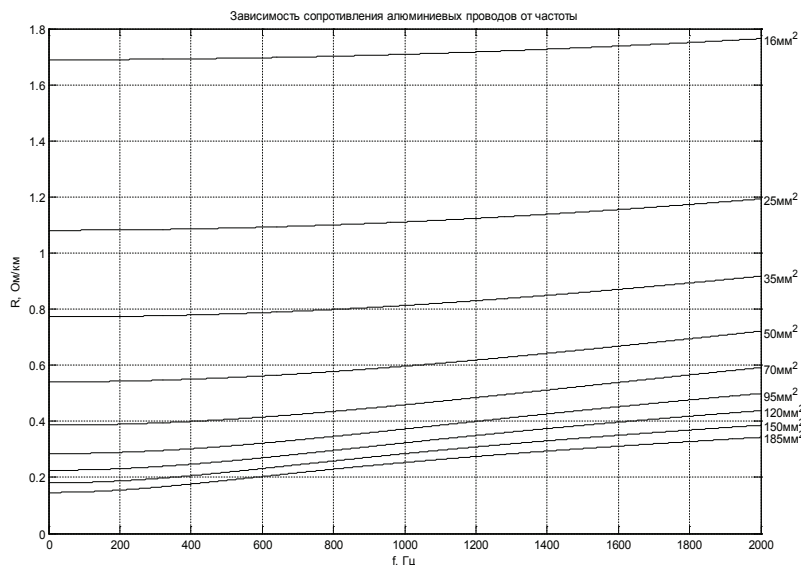


Рис. 2. Зависимость активного сопротивления алюминиевых проводов различных сечений от частоты

Результаты, полученные для алюминиевых проводов (рисунок 2.), аналогичны результатам для медных проводов. Сопротивление возрастает на величину от 4,4% до 133,8%, в зависимости от сечения провода, наименьший прирост погонного сопротивления наблюдается при сечении 16 мм², в этом случае погонное сопротивление увеличивается с 1,69 до 1,76 Ом/км, что составляет 4,4%, а при наибольшем приведенном на графике сечении 185 мм² погонное сопротивление увеличивается с 0,15 до 0,34 Ом/км, что в пересчете на проценты составляет 133,8%.

Таким образом для проводников большего сечения поверхностный эффект оказывает более выраженное влияние на сопротивление (для меди сопротивление возросло на 183,9%, а для алюминия на 133,8%, в полосе частот от 0 до 2000 Гц), уменьшение же сечения приводит к существенному возрастанию активного сопротивления (рисунки 1,2.). Проводники с большей удельной проводимостью более подвержены влиянию поверхностного эффекта. При равном сечении, по мере увеличения частоты, активное сопротивление медного провода возрастает быстрее, чем активное сопротивление алюминиевого провода.

Литература:

1. ГОСТ 32144–2013 Электрическая энергия. Совместимость технических средств электромагнитная. Нормы качества электрической энергии в системах электроснабжения общего назначения. [Текст] — М.: Стандартинформ — 2014 г. — 38 с.
2. Григорьев, О. Высшие гармоники в сетях электроснабжения 0,4 кВ/ О. Григорьев, В. Петухов, В. Соколов, И. Красилов [Текст] // «Новости электротехники», № 6 (18) 2002– 1 (19) 2003.
3. Бессонов, Л. А. Теоретические основы электротехники [Текст] — М.: Высшая школа — 1964 г. — 750 с.
4. Будзко, И. А. Электроснабжение сельского хозяйства/ И. А. Будзко, Т. Б. Лещинская, В. И. Сукманов [Текст] — М: Колос — 2000 г. — 536 с.
5. Бессонов, Л. А. Линейные электрические цепи / Л. А. Бессонов. — М: Высшая школа, 1983. — 336 с.
6. Лосев, А. К. Теория Линейных электрических цепей / А. К. Лосев. — М: Высшая школа, 1987. — 512 с.

Колбасные изделия из мяса индейки, обогащенные йодказеином

Бабченко Лидия Юрьевна, студент;
 Багирян Марина Ашотовна, студент;
 Патиева Александра Михайловна, доктор сельскохозяйственных наук, профессор
 Кубанский государственный аграрный университет

Статья посвящена исследованию и разработке технологии продуктов детского питания для школьного возраста для расширения ассортимента отечественных производителей.

Ключевые слова: детское питание, купаты, мясо индейки, йодказеин, норма потребления, рецептура, требования к составу

Биологически полноценные продукты, выработанные в промышленных условиях играют большую роль в организации здорового, сбалансированного питания детей как в домашних условиях, так и в организованных коллективах.

На сегодняшний день сегмент детского питания по сравнению с остальными товарными группами стал самым быстрорастущим [2].

В соответствии с концепцией сбалансированного питания определяющим фактором для обеспечения оптимального роста и развития детского организма является снабжение его необходимым количеством энергетического и пластического материалов, которые поступают с пищей. При этом должно обеспечиваться поступление в организм пищевых веществ не только в достаточных количествах, но и определенного качественного состава, соответствующего адаптационным возможностям желудочно-кишечного тракта ребенка и уровню его обменных процессов [3].

Рекомендуемая норма потребления белков для детей школьного возраста (7–11 лет) составляет 77 г/сут, из

них 46 г/сут — животный белок, физиологическая потребность в жирах — 70 г/сутки, в углеводах для детей — от 170 до 420 г/сутки, в энергии — от 1200 до 2900 ккал/сутки.

Особое значение предается при разработке продуктов детского питания обогащение их микронутриентами, которых не достает в том или ином климатическом регионе. Практически на всей территории центральной части Российской Федерации потребление йода с пищей и водой снижено. Реальное потребление йода составляет всего 40–80 мкг в день, т.е. в 2–3 раза ниже рекомендованного уровня.

Йод имеет большое значение для нормальной функции щитовидной железы, регуляции центральной нервной и сердечно — сосудистой систем, водно — солевого и других видов обмена. Йододефицитные состояния повышают не только риск развития эндемического зоба, аденомы щитовидной железы, но и являются одной из причин низкорослости, глухонмоты и нарушений умственной деятельности. Нормы потребления йода для детей школьного возраста представлены в таблице 1.

Таблица 1. Нормы потребления йода для детей школьного возраста

Возраст, лет	Нормы потребления, мг/сутки
6	80,0
7–10	100,0
11–13	100,0
14–17	130,0

В организм йод может поступать в двух видах — минеральном (неорганическом) и органическом. Минеральный йод — это йод, не связанный с какой-либо органической молекулой (спиртовой раствор йода, йодиды калия и натрия и т.п.).

Органический йод — это йод, находящийся в химической связи с каким-либо органическим веществом (сахара, полисахариды, аминокислоты).

Минеральный йод — чрезвычайно активное вещество. Он легко проникает в кровь и вступает в химические реакции с органическими веществами организма, изменяя их качества или разрушая их.

Органический йод, в отличие от минерального, находится в связанном состоянии, и в большинство химических реакций с органическими веществами организма не вступает. При этом йод, поступая через пищеварительный тракт в печень, под действием ферментов отщепляется от аминокислоты (тирозин, гистидин) и используется для синтеза гормонов щитовидной железы. Механизм регулирования органического йода, поступающего извне, контролируется через систему гомеостаза, и расщепление

органического йода идет строго индивидуально: организм получает йода ровно столько, сколько ему нужно [4].

Учитывая описанные преимущества органических соединений йода перед его неорганическими формами, в последние годы, наряду с традиционным способом профилактики — йодированием соли, в профилактике йододефицитных состояний все шире используются пищевые добавки для обогащения хлеба, молока и других продуктов и биологически активные добавки (БАД), содержащие органические формы йода. Одной из таких добавок является «Йодказеин» — препарат йода, связанного с белком молока казеином.

Йодказеин — йодированный молочный белок, являющийся полноценным аналогом природного соединения, изготовлен на основе натурального, легко усваиваемого белка молока — казеина, что обуславливает его физиологичность и естественность усвоения человеческим организмом. В ходе всесторонних исследований установлена функциональная пригодность йодказеина, подтверждена высокая степень эффективности и безопасности его применения.

Всемирная организация здравоохранения (ВОЗ) и Международный совет по контролю за йоддефицитными состояниями рекомендуют следующие физиологические дозы ежедневного потребления йода: — 120 мкг для детей (от 7 до 12 лет).

Заданные свойства продукта могут быть реализованы путем комбинирования и сочетания в рецептуре различных компонентов, каждый из которых обладает одним или рядом характеристик, позволяющих обеспечить эти свойства.

Целью исследований научной работы является обоснование использования йодказеина в технологии колбасных изделий для школьников (купатов), разработка рецептуры купатов из индейки, обогащенных йодказеином, определение пищевой ценности разрабатываемого продукта.

С учетом обозначенной цели были определены следующие задачи:

- формализовать медико-биологические требования к составу и качеству мясных купатов для питания школьников в возрасте 7–11 лет;

- произвести подбор ингредиентного состава с учетом нутриентных требований к модели рецептур колбасных изделий для питания школьников;

- изучить качественный состав, биологическую ценность и безопасность мяса индеек;

- разработать рецептуру и технологию колбасных изделий из мяса индейки, обогащенных йодказеином;

- выработать опытные образцы изделий;

- произвести исследование пищевой ценности новых видов сбалансированных мясных продуктов.

Объектами исследований являются: мясное сырье (мясо индейки) и готовые изделия — купаты.

В таблице 2 представлен химический состав и содержание витаминов в мясе индейки [1,5,6].

Таблица 2. Химический состав и содержание витаминов в мясе индейки

Показатель	Мясо индейки	
	1 сорта	2 сорта
Химический состав, г в 100 г продукта:		
белок	19,5	21,6
Жир	22,0	12,0
Углеводы	-	0,8
зола	0,9	1,1
Витамины, в 100 г продукта:		
А, мг	0,01	0,01
β -каротин, мг	следы	следы
Е, мг	0,34	-
В6, мг	0,33	0,33
ниацин (витамин РР, или витамин В3), мг	7,8	8,0
ибофлавин (витамин В2), мг	0,22	0,19
тиамин (витамин В1), мг	0,05	0,07
фолацин (витамин Вс, или витамин В9), мг	9,6	9,4
холин (витамин В4), мг	139	136
Энергетическая ценность, ккал в 100 г продукта	276	197

Содержание токсичных элементов в мясе индейки представлено в таблице 3.

В таблице 4 представлены показатели содержания пестицидов в мясе индеек.

Из исследованного мяса индеек были изготовлены опытные образцы купатов, обогащенных йодказеином,

для питания детей школьного возраста. Опытные образцы фарша для купатов были исследованы на содержание белка, жира, влаги и йода.

Исследования проводились в учебных лабораториях УНПК «Агротехпереработка» КубГАУ, в сертификационных лабораториях НИИ биотехнологии и сертификации

Таблица 3. Содержание токсичных элементов в мясе индеек

Наименование показателей	НД на методы испытаний	Допустимый уровень	Результаты анализа
			м.ср
Свинец, мг/кг	ГОСТ 30178–96	не более 0,5	0,048±002
Мышьяк, мг/кг	ГОСТ 26930–86	не более 0,1	<0,0025
Кадмий, мг/кг	ГОСТ 30178–96	не более 0,05	<0,01
Ртуть, мг/кг	МУ № 5178–90	не более 0,03	<0,005

Таблица 4. Показатели содержания пестицидов в мясе индеек

НД на методы испытаний	Наимен-е показателей	Доп. уровень	Результаты анализа
			m.cр
Метод определения микроколичеств пестицидов в продуктах питания, кормах и внешней среде. Под редак. М. А. Клисенко, т. 1,1992., изд. «Колос»	гексахлорцик-логексан, мг/кг (α , β , γ -изомеры)	не более 0,1	<<0,004
	ДДТ и его метаболиты, мг/кг	не более 0,1	<0,004

пищевой продукции КубГАУ, Северо-Кавказского НИИ животноводства, кафедры технологии хранения и переработки животноводческой продукции.

Методика проведения анализа фарша, для изготовления биточков, осуществлялась на приборе FoodScan (по руководству использования FoodScan).

FoodScan не требует трудоемкой подготовки образцов. Заполняли образцом чашку для образца и выравнивали его, создав плоскую поверхность. Образцы не должны содержать больших полостей или включений воздуха: это может повлиять на результаты.

После того, как определены поля «SampleID» (Идентификатор образца), «SampleDescription» (Описание образца) и «Remarks» (Замечания), они будут сохранены, и в ходе анализа в окне «Comments» будет показан идентификатор образца. Информация в комментариях также автоматически сохраняется в списке результатов.

Ход анализа: Поместив подготовленный образец в инструмент и правильно выбрав пользователя и программу анализа продукта, начинали анализ, выбрав кнопку «Start», на экран выводится окно комментария «Com-

ments», затем результаты анализа выводятся на экран монитора. Методы определения химического состава фарша представлены в таблице 5.

Измерение массовой доли йода выполняют титриметрическим методом. Метод определения массовой доли йода основан на удалении органических веществ, экстракции йодида, окислении йодида в йодат и выделении свободного йода, который оттитровывают серноватистокислым натрием и по расходу которого рассчитывают содержание йода в навеске исследуемого продукта [5].

Результаты исследований опытных образцов фарша на содержание белка, жира, влаги, коллагена и йода представлены в таблице 6.

Разработанная рецептура купатов обеспечивает адекватность состава продукта медико-биологическим требованиям в соответствии с современной теорией сбалансированного питания, учетом возрастных особенностей обменных процессов детского организма. Соотношение белка и жира в колбасных изделиях — 1: (1–1,5), в то время как в колбасах для взрослых — 1: (2–2,5). Уровень животного белка в этих продуктах — не менее 70%.

Таблица 5. Методы определения химического состава фарша

Исследуемый показатель	Наименование
Содержание влаги	FoodScan 78800
Содержание белка	FoodScan 78800
Содержание жира	FoodScan 78800
Содержание коллагена	FoodScan 78800
Содержание йода	Титриметрический метод. Методические указания МУК 4.1.1106–02
pH мяса	«MP по оценке мясной продуктивности, качества мяса и подкожного жира свиней». Утв. В. И. Фисининым, Москва, 1987
Интенсивность окраски	
Влагосвязывающая способность	
Содержание влаги	ГОСТ 9793–61
Содержание белка	ГОСТ 25011–81, п. 2
Содержание жира	ГОСТ 23042–86, п. 2
Содержание зольных веществ	ГОСТ 26929–94
Содержание свинца	ГОСТ 30178–96
Содержание мышьяка	ГОСТ 26930–86
Содержание кадмия	ГОСТ 30178–96
Содержание ртути	МУ № 5178–90
Содержание гексахлорциклогексана (α , β , γ — изомеры)	Методы определения микроколичеств пестицидов в продуктах питания, кормах и внешней среде. Под ред. М. А. Клисенко, т. 1, 1992., изд-во «Колос»
ДДТ и его метаболиты	

Таблица 6. Результаты испытаний фарша для купатов из мяса индейки обогащенных йодказеином для детского питания

Показатели	Метод испытания	Нормы	Результаты испытаний	
			Контрольный образец	Опытный образец
Белок	FoodScan 7800	не менее 12	15,96	15,96
Жир	FoodScan 7800	не более 16	8,93	8,93
Влага	FoodScan 7800	65–70	69,00	69,00
Коллаген	FoodScan 7800	-	0,86	0,86
Йод, мкг в 100 г продукта	МУК 4.1.1106–02	30–50	39,00	47,18

Содержание белка в готовых колбасных изделиях — не менее 12,0%, жира — не более 22,0%, нитрат натрия — 15,0–30,0 мг/кг, тогда как в продуктах общего назначения — 50,0–75,0 мг/кг.

Результаты исследований показывают, что фарш для купатов из мяса индейки, обогащенный йодказеином, является диетическим, полезным для детского организма,

содержит количество йода, необходимое для суточного потребления детским организмом.

Таким образом, с учетом суточного потребления йода для детей школьного возраста (120 мкг), можно установить, что при употреблении 250 грамм купатов, обогащенных йодказеином, ребенок полностью восполнит необходимую суточную дозу йода.

Литература:

1. Тимошенко, Н. В., Патиева С. В. Технология специализированных, лечебно-профилактических детских продуктов на мясной основе: Учебное пособие. — Краснодар: КубГАУ, 2010. — 368 с.
2. Тимошенко, Н. В. Технология хранения, переработки и стандартизации мяса и мясных продуктов: Учебное пособие. — Краснодар, КубГАУ, 2010. — 354 с.
3. Тимошенко, Н. В. Технология переработки и хранения продукции животноводства: Учебное пособие. — Краснодар, КубГАУ, 2010. — 576 с.
4. Устинова, А. В., Тимошенко Н. В. Мясные продукты для детского питания. — М.: ВНИИМП, 1997. — 252с.
5. Устинова, А. В., Тимошенко Н. В. Продукты для детского питания на основе мясного сырья — Учебное пособие — М.: Изд-во ВНИИМП, 2003—438с.
6. Функциональные продукты питания: ресурсосберегающие технологии переработки сельскохозяйственного сырья, гигиенические аспекты и безопасность // Материалы международной научно-практической конференции / Кубанский государственный аграрный университет. — Краснодар, 2009. — 792с.

Разработка баз данных для электроэрозионных станков

Бобков Николай Владимирович, аспирант
Омский государственный технический университет

В машиностроении помимо традиционных материалов (сталей, алюминиевых и латунных сплавов) требуется применение редких металлов, таких как цирконий, ниобий и ванадий.

Для производства тепловыделяющих элементов (рис.1) и дистанционирующих решеток (рис.2), используемых в ядерных реакторах ВВЭР-1000, используются три основных циркониевых сплава — Э635, Э110 и Э125.

Кроме того, цирконий применяется в качестве катодов в установках магнетронного напыления (рисунок 3), для получения многофункциональных покрытий. Наряду с цирконием, в качестве катодов (рисунок 4), так же используются другие редкоземельные металлы, такие как ниобий и ванадий.

Однако при изготовлении деталей из редкоземельных (и как следствие дорогих) металлов, необходимо стремиться к минимизации количества отходов. Например, можно уменьшить ширину реза, сократив тем самым количество стружки. Хорошие результаты в решении этой проблемы были достигнуты на проволочном электроэрозионном станке, так как при диаметре проволочного электрода 0,2 мм ширина реза не превышает 0,3 мм.

С помощью электроэрозионного оборудования возможна обработка любых электропроводных металлов. Однако производители оснащают свои станки набором режимов обработки только для наиболее распространенных материалов (рисунок 5). В случае, когда необходима обработка металлов не входящих в базу данных



Рис. 1. Тепловыделяющий элемент



Рис. 2. Дистанционирующая решетка



Рис. 3. Установка магнетронного напыления



Рис. 4. Катоды из циркония, ниобия, ванадия и меди

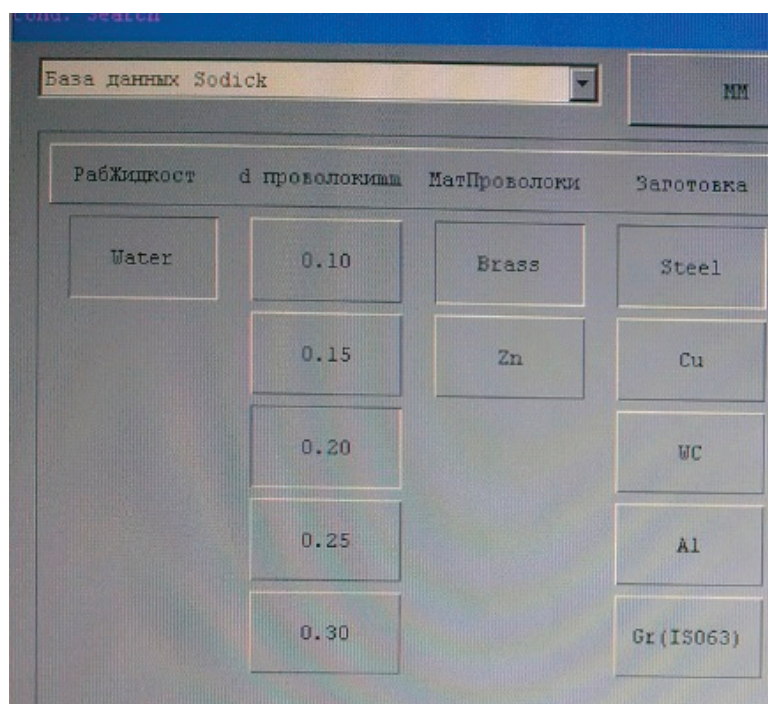


Рис. 5. База данных для расчета режимов обработки

станка (таких как цирконий, ниобий и ванадий), подбор режимов осуществляется экспериментально, что приводит к снижению производительности. Как видно из рисунка 5 в стандартной комплектации станки оснащаются только режимами для обработки стали, меди, алюминия, твердого сплава и графита (ISO 63).

При обработке редкоземельных материалов на несоответствующих им режимах возникает ряд проблем. Во-первых, в ходе эксперимента необходим постоянный контроль работы станка, фиксация результатов, а так же анализ и исправление ошибок выявленных опытным

путем. Во-вторых, при завышенных режимах, инструмент (в нашем случае это латунная проволока) быстрее приходит в негодность, что увеличивает время на наладку и затраты на инструмент. Так же стоит отметить, что при заниженных режимах уменьшается скорость обработки, что приводит к увеличению машинного времени и как следствие увеличение стоимости детали.

Для более детального изучения был проведен ряд экспериментов по обработке циркония, ниобия и ванадия на проволочном электроэрозионном станке Sodick VZ300L (рисунок 6).



Рис. 6. Образец из ванадия, обработанный на станке Sodick VZ300L

В ходе эксперимента были подобраны режимы обработки в один, два, три и четыре прохода [1] (рисунок 7,8).

Полученные режимы показали стабильную обработку. Однако, с помощью датчика контроля, установленного

на станке (рисунок 9), была выявлена неточность в расчетах, что требует ввода не учтенных данных, повторного пересчета и влечет за собой повторное проведение экспериментов.



Рис. 7. Режимы резания для 3 проходов

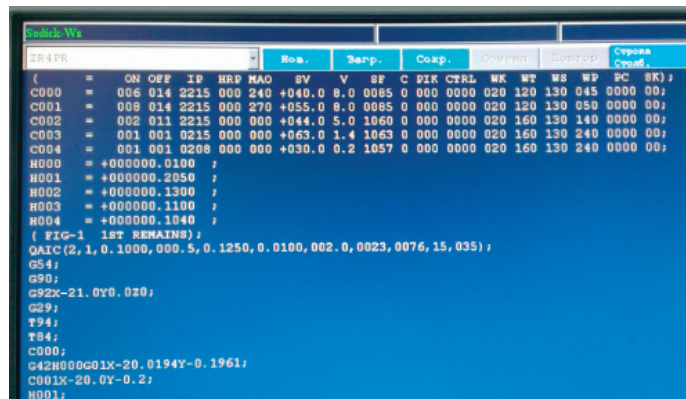


Рис. 8. Режимы резания для 4 проходов



Рис. 9. Показания датчика контроля обработки

Учитывая вышесказанное можно заключить, что обработка металлов, не учтенных базой данных электроэрозийного станка (как в рассматриваемом случае), возможна после подбора режимов. Однако для стабильной и наиболее производительной обработки, необходимы расчеты, которые позволят получить оптимальные ре-

жимы резания. Поэтому естественно предположить, что для предприятий, занимающихся производством деталей из редкоземельных металлов, существует необходимость приобретения высокопроизводительного и высокоточного оборудования, имеющего в своей базе данных режимы, подходящие для обработки этих материалов.

Литература:

1. Руководство пользователя к станку Sodick VZ300L.

Выбор геометрических параметров ограничителя переворота гусеничного движителя устанавливаемого на автомобиль УАЗ-Патриот

Бруданов Антон Михайлович, студент
Пермский национальный исследовательский политехнический университет

Статья посвящена обзору разработанной конструкции ограничителя переворота гусеничного движителя, устанавливаемого на автомобиль УАЗ-Патриот. Произведен расчет данной конструкции на прочность. Приведен анализ определения геометрических параметров и выбора материала для изготавливаемой конструкции.

Ключевые слова: ограничитель переворота, вездеходные гусеничные движители, вездеходные устройства, автомобиль, крепление гусеничных устройств

Choice of geometric parameters of revolution limiter caterpillar tracks mounted on a UAZ-Patriot

A. M. Brudanov

The article provides an overview of the structure developed by the revolution limiter caterpillar tracks mounted on UAZ-Patriot. The calculation of the design strength. The analysis of the geometric parameters and the choice of material for the manufactured design.

Keywords: limiter coup, all-terrain caterpillar tracks, all-terrain unit, a car mount tracked devices

Гусеничные движители, в отличие от колесных, развивались значительно медленнее. Но благодаря тому, что гусеничные движители имеют большую площадь поверхности взаимодействия с грунтом и способны развивать высокую силу тяги, трактора и некоторые машины с таким устройством ходовой части, давно стали применяться как база или погрузочных машин для работы на снегу, влажных почвах или в грунтах с низкой несущей способностью. Гусеничные движители часто использовались на лесохозяйственных тракторах, грузовиках и различной сельскохозяйственной технике. В современный век технологий и конструкторских решений были разработаны гусеничные движители, устанавливаемые и на легковые внедорожные автомобили.

Сегодня современный гусеничный движитель может выглядеть, как и его предшественники: основу балансирной тележки составляет треугольная рама. На самом вершуге рамы находится ведущая звездочка, которая исполняет роль колеса. Основание треугольника состоит из четырех опорных роликов. Передний ролик отвечает лишь за натяжение резиновой ленты и дает больший угол въезда на препятствие. Вся тележка может качаться вокруг оси. С одной стороны это может быть полезным, так как обеспечивает хорошую адаптацию к определенным дорожным

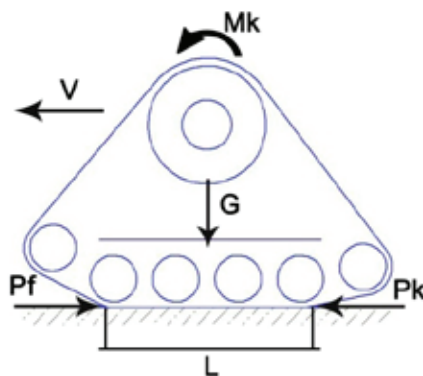


Рис. 1. Схема сил действующих на движитель, где: M_k — крутящий момент; G — вес автомобиля; P_k — сила тяги; P_f — сила сопротивления движению; V — направление движения; L — длина опорной поверхности гусеничного движителя

рельефам. С другой стороны это может быть не безопасным так, как движитель будет стремиться провернуться во- круг оси. Чтобы этого не произошло на раме тележки необходимо установить ограничитель.

Так как гусеничный движитель представляет собой конструкцию несимметричную относительно оси вращения, дви- житель под действием некоторого момента пытается провернуться вокруг оси под препятствие или при упоре переднего или заднего угла движителя. Для проектирования механизма ограничения переворота необходимо представлять вели- чину момента, которая будет стремиться перевернуть движитель. Определяем крутящий момент одного колеса.

- Максимальный крутящий момент $M_d = 210 \text{ Н} \cdot \text{м}$;
- Передаточное отношение первой передачи $i_1 = 3,7$;
- КПД первой передачи $\mu_1 = 0,95$;
- Передаточное отношение раздаточной коробки $i_p = 2,542$;
- КПД понижающей передачи $\mu_2 = 0,97$;
- Передаточное отношение главной передачи $i_{гп} = 4,11$;
- КПД главной передачи $\mu_2 = 0,96$;

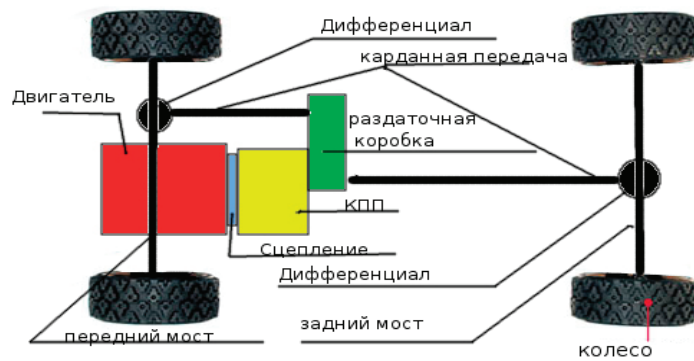


Рис. 1.1. Схема трансмиссии полноприводного автомобиля

$$M_k = \frac{M_d \cdot i_1 \cdot \mu_1 \cdot i_p \cdot \mu_2 \cdot i_{гп} \cdot \mu_2}{4} = \frac{210 \cdot 3,7 \cdot 0,95 \cdot 2,542 \cdot 0,97 \cdot 4,11 \cdot 0,96}{4} = 1782,61 \text{ Нм}$$

Крутящий момент на одном колесе M_k и будет приближенным моментом, стремящимся перевернуть гусеничным движитель.

В программной среде SolidWorks (Солидворкс) была спроектирована конструкция ограничителя переворота для ав- томобиля УАЗ-Патриот (рис.2). Материал простая углеродистая сталь (предел текучести 220 Мпа, толщина металла 8 мм).

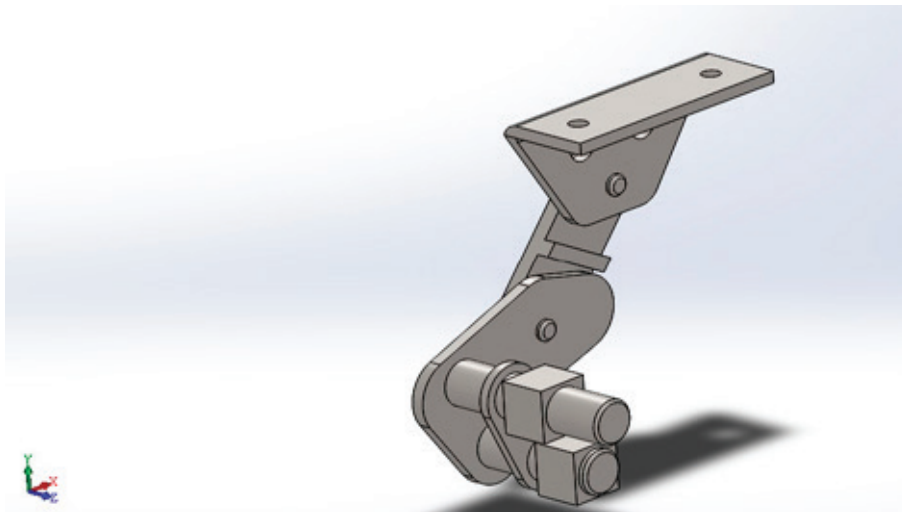


Рис. 2. Ограничитель переворота гусеничного движителя для автомобиля УАЗ-Патриот

Был произведен статический расчет на прочность, в результате которого под максимальной нагрузкой запаса прочности материала не хватает (рис.2.1).

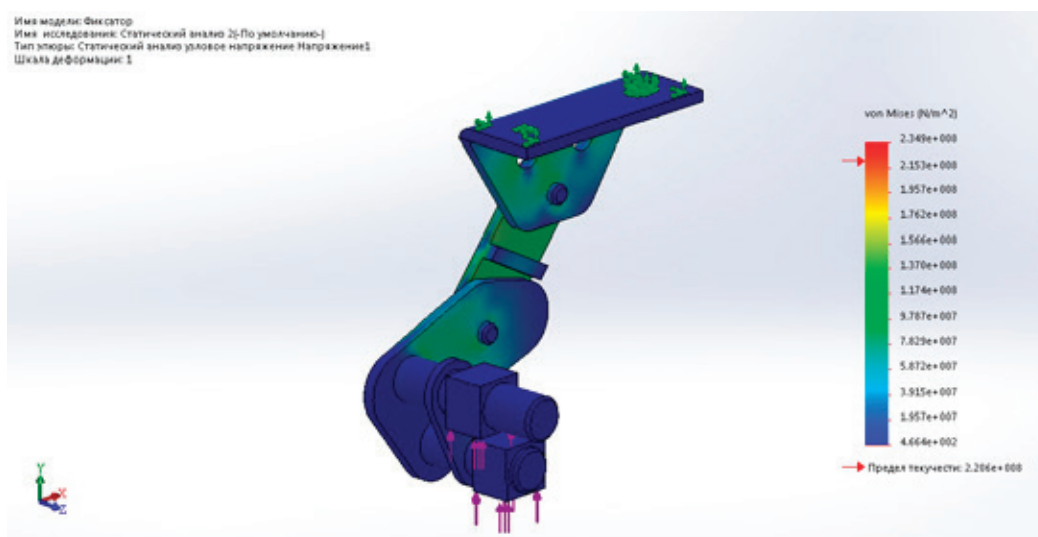


Рис. 2.1. Ограничитель переворота гусеничного движителя для автомобиля УАЗ-Патриот под нагрузкой (предел текучести стали 220МПа, толщина связывающей проушины 8мм)

Было принято решение, заменить сталь на углеродистую с запасом прочности 280 МПа и увеличить толщину металла связывающей проушины до 10мм. В результате принятых изменений коэффициент прочности равняется 2, что удовлетворяет заданным условиям.

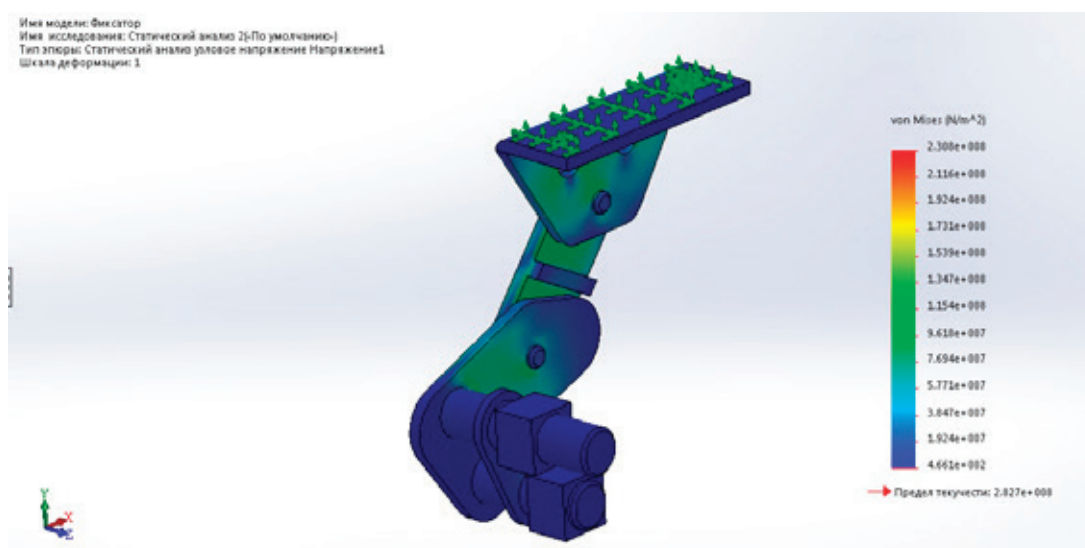


Рис. 2.2. Ограничитель переворота гусеничного движителя для автомобиля УАЗ-Патриот под нагрузкой (предел текучести стали 280МПа, толщина связывающей проушины 10мм)

Подводя итоги, хочется сказать, что использование гусеничных движителей по грунтам с малой несущей способностью возможно и дает преимущества, перед пневмоколесными движителями, но с другой стороны это может быть не безопасным так, как движитель будет стремиться повернуться вокруг оси. Чтобы этого не произошло на раме тележки необходимо устанавливать ограничитель. Приведенные в данной статье расчеты наиболее нагруженных мест ограничителя гусеничного движителя, показали, что данную конструкцию можно применять на опытных образцах автомобиля УАЗ-Патриот. Следует помнить, что для каждой модели автомобиля ограничитель будет иметь уникальную конструкцию, которую необходимо рассчитывать на прочность.

Литература:

1. Носов, Н. А., Галышев В. Д., Волков Ю. П., Харченко А. П. Л., Расчет и конструирование гусеничных машин. «Машиностроение», 1972—560с.
2. Стрелков, А. Г. Конструкция быстроходных гусеничных машин: учеб. пособие — Москва. МАМИ 2005—664с.
3. Кивокурцев, О. А., Пугин К. Г., Утев М. А. Классификация многопрофильных роботизированных платформ по тягово-сцепным характеристикам: Бюллетень транспортной информации. 2014. № 8 (230). с. 28—33.
4. Чобиток, В. А., Данков Е. В., Брижинев Ю. Н. и др. Конструкция и расчёт танков и БМП. Учебник. — М.: Воениздат, 1984. — 376 с.
5. Дунаев, П. Ф., Леликов О. П. Конструирование узлов и деталей машин. — 4-е изд., перераб. и доп. — М.: Высшая школа, 1985. — 416 с.
6. Наумов, Е. С., Платонов В. Ф., Ходовая система гусеничного трактора. Учебное пособие для студентов специальности «Автомобиле- и тракторостроение»/ Под ред. В. М. Шарипова. — М.: МГТУ «МАМИ», 2011. — 64 с.
7. Пугин К. Г., Бургонутдинов А. М. Машины для строительства, ремонта и содержания автомобильных дорог. ч.1. Дорожные катки и одноковшовые погрузчики — Пермь: Пермский нац. исслед. политехн. ун-т». Пермь, 2011. — 172с.

Обзор существующих конструкций для повышения проходимости автомобиля категории М1

Бруданов Антон Михайлович, студент

Пермский национальный исследовательский политехнический университет

Статья посвящена обзору существующих конструкций для повышения проходимости автомобилей в условиях бездорожья. Приводится сравнение эффективности использования устройств, при эксплуатации в различных дорожных условиях, рассмотрены сильные и слабые стороны конструкций. Проведен анализ удобства установки устройств на автомобиль. Рассмотрена технико-экономическая эффективность использования устройств, для повышения проходимости автомобилей.

Ключевые слова: цепи противоскольжения, повышение проходимости, вездеходные устройства, гусеничная лента, автомобиль, дискретные шириатели, шины низкого давления

All terrain vehicle category M1 when installing crawler system

A. M. Brudanov

Article is devoted to cross-country vehicles off-road tracked using a specially designed propellers. The comparison of the efficiency of the use of wheeled and tracked propulsion when driving on soils with low bearing capacity. Options considered terrain vehicle when driving on snow cover.

Keywords: caterpillar tracks, crawler, all-terrain device Track car

Полноприводные автомобили занимают немалую часть парка автомобильной техники в нашей стране, широко используются в различных сферах деятельности человека. Чаще их называют — автомобили многоцелевого назначения. Такие машины необходимы при осуществлении транспортных операций в промышленном и дорожном строительстве, устранении чрезвычайных ситуаций, геологоразведке, нефте- и газодобыче, при геологоразведке, сельском хозяйстве, обслуживании предприятия энергетики. Условия эксплуатации таких автомобилей предусматривают движение в различных до-

рожных условиях, при этом большую часть пути автомобили находятся в условиях бездорожья. В связи с чем, актуальным остается вопрос повышения проходимости.

Сложность при создании внедорожного автомобиля в том, что поверхности с низкой несущей способностью такие как, снег, песок, болотистая почва по своим физическим свойствам, воспринимают вертикальную нагрузку и сопротивляются приложенному тяговому усилию, очень разнообразно. Сложно создать универсальную вездеходную машину, одинаково успешно передвигающуюся при различных дорожных условиях.

Повысить проходимость колесных машин, можно за счет различных устройств, таких как: цепи противоскольжения, браслеты и скобы, противобуксовочные колодки, уширители различных конструкций, шины низкого давления или иных приспособлений, которые монтируются на штатные пневмоколесные движители.

Широкое применение получили цепи противоскольжения, различных конструкций, включая мелкозвенчатые, траковые и плицевые цепи (см. Рисунок 1). Существенным образом увеличивают сцепление колеса с грунтовой, заснеженной или обледенелой дорогой. Сила тяги, реализуемая колесами, возрастает до 60% [1].

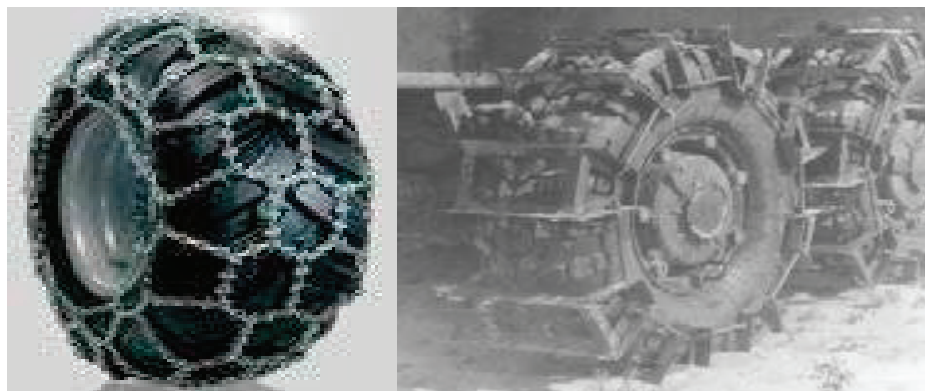


Рис. 1. Цепи противоскольжения: а) мелкозвенчатые; б) плицевые конструкции НГТУ

При эксплуатации машин по снежной целине, характеристики цепей противоскольжения меняются. Экспериментальные исследования показали, что при высоте снега более 0,6 радиуса колеса, наряду с приростом тяги в 30–40% имеют свойство интенсивно увеличивать глубину колеи, а так же увеличивается сопротивление движению до 60% [3]. Ощутимый эффект в увеличении проходимости наблюдается лишь тогда, когда высота снега находится в пределах меньших, чем половина радиуса колеса. Сила тяги в данном случае может увеличиваться до 30%, а сила сопротивления движению увеличивается

при этом несущественно до 10% [2]. Цепи противоскольжения удобно использовать, так как монтаж не занимает много времени. Экономическая эффективность оправдана, так как конструкция имеет низкую цену.

Еще одним способом увеличения проходимости машины является применение дискретных уширителей (см. Рисунок 1.1). Но проверка эффективности экспериментальным путем этих устройств показала, что применение этих устройств дает ощутимый эффект только для машин с колесной формулой 8х8, при этом сила тяги возрастает на 25–30% [4].



Рис. 1.1. Дискретные уширители

Использование ленточного уширителя, представляющего собой две резиноканевые ленты, соединенные между собой металлическими грунтозацепами, увели-

чивает тяговые свойства машины на снежной целине до 25%, при этом сопротивление движению снижается на 30%. Однако применение уширителей значительно уве-

личивает габаритную ширину машины, а при поворотах уширители имеют склонность к спаданию.

Осуществлялись попытки применения гусениц для повышения проходимости автомобиля. Впервые такие попытки были предприняты в первой половине 20 века. Однако несмотря на то, что гусеничные движители су-

щественно увеличивают проходимость машины по снегу, данное направление не получило дальнейшего развития. Это связано с тем, что машины с колесно-гусеничным движителем имеют большую на 50–70% массу, низкую надежность и являются более сложными и дорогими.



Рис. 1.2. Полугусеничная машина на базе автомобиля ГАЗ-51

Следующим направлением в вопросах о повышении проходимости колесных машин стало применение высокоэластичных пневмоколесных движителей сверхнизкого давления на базе существующих широкопрофильных шин. Шины сверхнизкого давления имеют тонкостенную резино-кордную оболочку с каркасом, состоящим, как правило, из двух или четырех слоев корда, благодаря чему обеспечивается высокая эластичность. Для достижения максимальной проходимости вездехода на шинах сверхнизкого давления в условиях слабонесущих, пластично деформируемых грунтов необходимо обеспечить такое давление в шине,

чтобы избыточное давление в шине сравнялось с давлением грунта на шину на дне колеи, а на шине возникало плоское пятно контакта равное $1/3$ – $1/4$ диаметра колеса. Границей, разделяющей шины низкого давления и шины сверхнизкого давления, считается внутреннее давление в шине равное 0,3 МПа. При взаимодействии с грунтом такая шина, так же как и пневмокоток, не разрушает его поверхность и приобретает способность «обтекать» неровности пути, а выступы и впадины беговой дорожки, повторяющей профиль поверхности пути, выполняют роль своеобразных грунтозацепов, увеличивая сцепление с опорной поверхностью.



Рис. 1.3. Вездеходное транспортное средство производства Трэкол

Еще одним эффективным способом повышения проходимости колесных машин по снегу и грязи является применение вездеходных гусеничных движителей (полугусеничных и гусеничных ходов). Основная идея и разработка конструкции принадлежит американцу Глену Брэйзиру.

Конструкция гусеничного движителя изображена на рисунке 2. За счет угла создаваемого между плоскостью дороги и передним натяжным роликом поз.1 улучшается проходимость за счет более легкого въезда на препятствия. Так же подобный угол будет устроен и у заднего опорного ролика.

Всего в конструкции будет пять опорных поз.2 и один натяжной ролик поз.1 [2]. Плавность хода будет основываться на стандартной подвеске автомобиля и за счет вращения балансирной тележки поз.4 вокруг оси колес. Вращение те-

лежки будет контролироваться ограничителем переворота, без которого эксплуатация автомобиля на данных движителях невозможна. Натяжение движителя может осуществляться разведением переднего натяжного ролика поз.1.

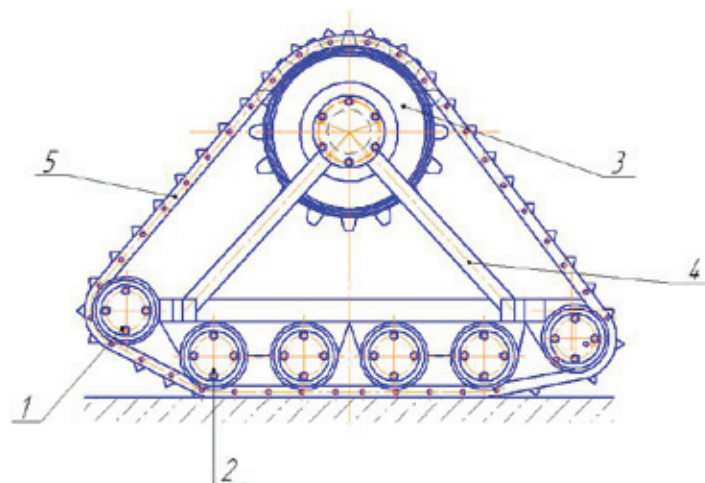


Рис. 2. Гусеничный ход для автомобиля категории M_1 , где: 1 — ведомый натяжной ролик; 2 — опорные ролики; 3 — ведущее колесо; 4 — треугольная балансирная тележка; 5 — движители

Одно из основных преимуществ гусеничной машины по сравнению с колесной — значительно меньшее удельное давление [3]. Опорная поверхность колес автомобиля меньше опорной поверхности гусеничных движителей. Из-за большого удельного давления колеса автомобиля при движении по мягкой почве слишком глубоко погружаются в грунт. Ввиду этого, сопротивление качению автомобиля может оказаться столь значительным, что автомобиль не сможет двигаться.

Вездеходные гусеничные движители универсальны и устанавливаются на внедорожный автомобиль категории M_1 без изменения его конструкции на ступичный узел взамен пневмоколесных движителей.

Гусеничные системы являются альтернативой гусеничной техники для перемещения в труднопроходимых ус-

ловиях. Технично-экономические расчеты показали, что гусеничные движители обходятся намного дешевле, чем приобретение новой гусеничной техники. Применение их может оказать не заменимую помощь для служб МЧС и Скорой медицинской помощи в отдаленных районах, куда колесная техника добраться не может. Гусеницы для автомобилей можно использовать в разных целях, будь то работа или отдых, развлечения, например: охота, рыбалка, туризм, сельское хозяйство, строительство, поиск и спасение в непроходимых местах, научно-исследовательские ситуации, и многое другое. Автомобили, оборудованные гусеничными движителями, превосходно ведут себя на различных покрытиях.

Литература:

1. Михалин, П. А., Шины и цепи колесных лесных тракторов. Труды Международного симпозиума «Надежность и качество», 2 / 2007.
2. Беляков., В.В. Взаимодействие со снежным покровом эластичных движителей специальных транспортных машин: дис. д-ра техн. наук: 05.05.03. НГТУ, Н. Новгород, 1999—485 с.
3. Применение мелкозвенчатых цепей противоскольжения для повышения проходимости автомобилей. Сборник № 2 Аннотации научно-исследовательских работ по проблемам повышения проходимости колесных машин, Москва, 1958 г. ИКТП АН СССР, стр.18—23.
4. Беляков., В.В. Взаимодействие со снежным покровом эластичных движителей специальных транспортных машин: дис. д-ра техн. наук: 05.05.03. НГТУ, Н. Новгород, 1999—485 с.
5. Носов, Н. А., Галышев В. Д., Волков Ю. П., Харченко А. П. Л., Расчет и конструирование гусеничных машин. «Машиностроение», 1972—560с.
6. Стрелков, А. Г. Конструкция быстроходных гусеничных машин: учеб.пособие — Москва. МАМИ 2005—664с.
7. Наумов, Е. С., Платонов В. Ф., Ходовая система гусеничного трактора. Учебное пособие для студентов специальности «Автомобиле- и тракторостроение»/ Под ред. В. М. Шарипова. — М.: МГТУ «МАМИ», 2011. — 64 с.

Результаты определения параметров сопротивления грунтов сдвигу в приборе трехосного сжатия

Буш Илья Валерьевич, студент
Сибирская государственная автомобильно-дорожная академия

В статье кратко описаны методики испытаний грунтов трехосным сжатием и приведены правила вычисления сцепления, угла внутреннего трения и сопротивления недренированному сдвигу. Указана область применения полученных результатов в современных расчетах дорожных одежд.

Ключевые слова: земляное полотно, грунт, сцепление, угол внутреннего трения

Дисперсные грунты нашли широкое применение при устройстве земляного полотна, в том числе их используют для замены особых и слабых торфяных и заторфованных грунтов в основаниях насыпей. Независимо от назначения грунтов необходимо выполнять их проверку по сопротивлению сдвигу, для которой выполняется расчет касательных напряжений или безопасных давлений. Для таких расчетов необходимы данные о величине угла внутреннего трения, сцепления, а иногда сопротивления недренированному сдвигу. Эти параметры принимают по данным нормативной литературы [1], а в ряде случаев находят экспериментально, выполняя требования [2]. Отметим, что сцепление и угол внутреннего трения необходимы для расчета касательных напряжений не только по оригинальному критерию Кулона—Мора, но и ряда современных эмпирических и модифицированных условий пластичности [3–7]. Эти параметры применяют для расчета безопасных давлений [8–

11]. Известно, что величина сцепления и угла внутреннего трения возрастает при увеличении коэффициента уплотнения [12–14], что требует приготовления лабораторных проб с плотностью сухого грунта, соответствующей как требуемым значениям, так и фактическим значениям.

Поэтому для изготовления образцов из суглинистых и песчаных грунтов нами использовался прибор стандартного уплотнения, в котором грунты уплотняли при оптимальной влажности до коэффициента уплотнения 0,98...1,0. Так изготавливали образцы больших размеров, из которых при помощи пробоотборника извлекали образцы меньшего размера необходимого для испытаний в приборе трехосного сжатия.

Тип испытания принимали в соответствии с руководством стандартов РФ, США и Евросоюза. Краткое описание и назначение различных типов испытаний приведено в табл. 1.

Таблица 1. Типы и условия трехосных испытаний грунтов

Тип испытаний	Применяемые стандарты	Условия испытаний	Определяемые параметры
Неконсолидированно-недренированные (НН испытания) ЕНН (НУУ)	ГОСТ 12248–10, ASTM D 2850, BS 1737–7, ISO 17892–8	Без измерения порового давления	Недренированная прочность c_u
Консолидированно-недренированные (КН испытания) ВКН (SCU)	ГОСТ 12248–10, ASTM D 4767, BS 1737–8	С измерением или без измерения порового давления	Параметры прочности в полных или эффективных напряжениях: $j, c, j\phi, c\phi$
Консолидированно-дренированные (КД испытания) ВКД (SCD)	ГОСТ 12248–10, ISO 17892–9	Без измерения порового давления	Параметры прочности в полных напряжениях j, c . Модуль упругости и модуль общей деформации: E_e, E .

Для грунтов с частичным водонасыщением таких, как глины, суглинки (в уплотнённом состоянии) расположенные выше уровня грунтовых вод недренированная прочность (сопротивление недренированному сдвигу) должна быть определена из НН-испытаний образцов, у которых коэффициент пористости и степень водонасыщения соответствует этим же параметрам в реальном массиве грунта (земляном полотне). В КН испытаниях

образец сжимается в условиях полного дренирования при действии всестороннего давления (первый этап испытаний, характеризуемый напряжениями $s_1=s_2=s_3$), а при сдвиге дренирование не допускается (второй этап испытаний — $s_1>s_2=s_3$). Основным ограничением для КН испытаний является длительность проведения вследствие низкой проницаемости глинистых грунтов и малой скорости фильтрации поровой воды. КН-испытания могут

быть выполнены быстрее, чем КД-испытания, а результаты дают примерно одинаковую предельную огибающую при ее построении в эффективных напряжениях. Следовательно, при КН-испытаниях с измерением порового давления прочность грунта может быть выражена функцией эффективных напряжений, а затем использована для решения задач, где есть дренирование.

На рисунках 1 и 2 приведены фрагменты испытаний суглинка легкого и песка среднего по типу КН испытаний, а на рис. 3 показано определение сцепления и угла внутреннего трения суглинка легкого, выполненное построением кругов предельных напряжений и касательной (предельная прямая Кулона—Мора) к ним. Сопротивление недренированному сдвигу суглинка легкого выполнено при помощи НН-испытаний.

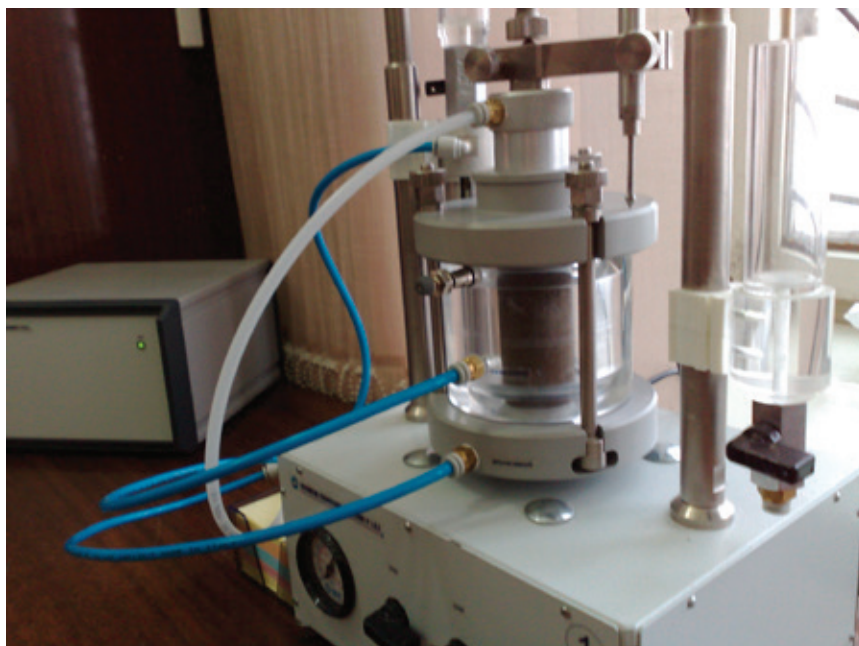


Рис. 1. Испытание суглинка



Рис. 2. Испытание песчаного образца в оболочке

Установленные значение параметров сопротивления сдвигу будут применены в дальнейшем для расчета касательных напряжений и безопасных давлений. При этом

планируется вычислять необходимые для этих расчетов главные напряжения как по традиционным решениям [15, 16], так и с использованием современных методик [17–20].

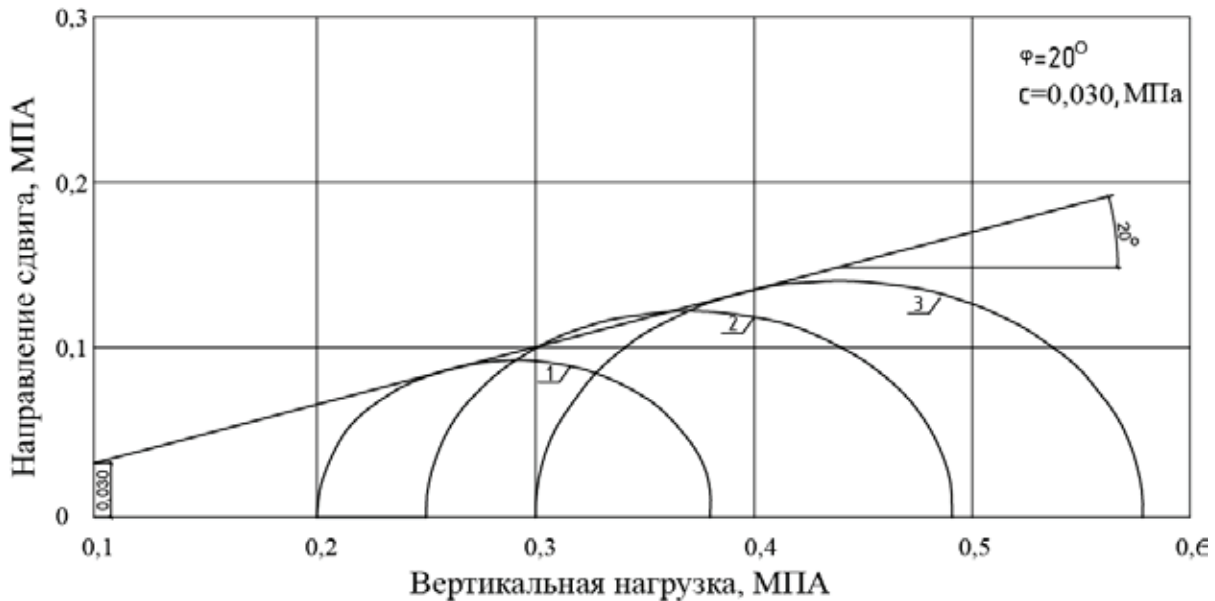


Рис. 3. Круги предельных напряжений и предельная прямая Кулона–Мора

Литература:

1. ОДН 218.046–01. Проектирование нежестких дорожных одежд. — М.: ГСДХ Минтранса России, 2001. — 146 с.
2. ГОСТ 12248–2010. Грунты. Метод лабораторного определения характеристик прочности и деформируемости. Взамен ГОСТ 12248–96 и ГОСТ 24143–80. Введен в действие 01.01.2012. М.: Стандартинформ, 2011.
3. Александров, А. С., Долгих Г. В., Калинин А. Л. Модификация критериев прочности сплошной среды для расчета грунтов земляного полотна по сопротивлению сдвигу // В сборнике: Архитектура. Строительство. Транспорт. Технологии. Инновации Материалы Международного конгресса ФГБОУ ВПО «СибАДИ». — Омск: СибАДИ, 2013. — с. 228–235.
4. Александров, А. С., Калинин А. Л. Совершенствование расчета дорожных конструкций по сопротивлению сдвигу. Часть 1. Учет деформаций в условии пластичности Кулона — Мора // Инженерно-строительный журнал. — 2015. № 7 (59). — с. 4–17.
5. Александров, А. С., Долгих Г. В., Калинин А. Л. Применение критерия Друкера — Прагера для модификации условий пластичности // Наука и техника в дорожной отрасли. — 2013. № 2. — с. 26–29.
6. Чусов, В. В. Перспективы применения эмпирических условий пластичности грунтов и определение их параметров при трехосных испытаниях грунтов Вестник ВолГАСУ. — 2015. № 42 (61). — с. 49–57.
7. Калинин, А. Л. Совершенствование расчета касательных напряжений в дорожных конструкциях. Часть 1. Модификация критерия Писаренко-Лебедева и его применение при расчете касательных напряжений // Молодой ученый. — 2016. — № 6 (110). — с. 108–114.
8. Александров, А. С., Долгих Г. В., Калинин А. Л. О допускаемых давлениях на грунты земляного полотна и слою дорожной одежды // Наука и техника в дорожной отрасли. — 2012. № 2. — с. 10–13.
9. Долгих, Г. В. Расчет грунтов земляного полотна по критерию безопасных давлений // Вестник Сибирской государственной автомобильно-дорожной академии. — 2013. — № 6 (34). — с. 43–49.
10. Долгих, Г. В. Расчет нежестких дорожных одежд по критерию безопасных давлений на глинистые грунты земляного полотна // Автореф. Дис. канд. техн. наук. — Омск: СибАДИ. — 2014. — 20 с.
11. Долгих, Г. В. Применение критерия безопасных давлений для расчета дорожных конструкций по сопротивлению сдвигу в грунте земляного полотна // В сборнике: Политранспортные системы материалы VIII Международной научно-технической конференции в рамках года науки Россия — ЕС. Новосибирск: СГУПС, 2015. — с. 176–182.
12. Александрова, Н. П., Семенова Т. В., Стригун К. Ю. Совершенствование методов экспресс оценки качества уплотнения грунтов земляного полотна строительства автомобильных дорог / Н. П. Александрова, // Вестник СибАДИ. — 2015. — № 4. — с. 46–57.
13. Семенова, Т. В., Долгих Г. В., Полугородник Б. Н. Применение Калифорнийского числа несущей способности и динамического конусного пенетрометра для оценки качества уплотнения грунта // Вестник СибАДИ, 2014, № 1 — с. 59–66.

14. Александрова, Н. П., Троценко Н. А. Применение измерителя жесткости грунта Geogauge для оценки качества уплотнения при операционном контроле // Вестник СибАДИ, 2014, № 3 — с. 40–47.
15. Foster, C. R., Ahlvin R. G. Stresses and deflections induced by a uniform circular load. // Proc. Highway Research Board. — 1954. — Vol. 33. — P. 236–246.
16. Craig, R. F. Soil Mechanics. — Seventh edition. Department of Civil Engineering, University of Dundee, UK. — Published by Taylor & Francis e-Library, London and New York, 2004. — 447 p.
17. Александрова, Н. П. Модифицированные модели для расчета главных напряжений в грунте земляного полотна // В сборнике: Архитектура. Строительство. Транспорт. Технологии. Инновации Материалы Международного конгресса ФГБОУ ВПО «СибАДИ». Омск, 2013. — с. 236–246.
18. Александрова, Н. П., Семенова Т. В., Долгих Г. В. Совершенствование моделей расчета главных напряжений и девиатора в грунте земляного полотна // Вестник СИБАДИ. — 2014. — № 2 (36). с. 49–54.
19. Александров, А. С. Один из путей расчета минимальных главных напряжений в грунтах земляного полотна / А. С. Александров // В сборнике: Архитектура. Строительство. Транспорт. Технологии. Инновации Материалы Международного конгресса ФГБОУ ВПО «СибАДИ». — Омск, СибАДИ, 2013. — с. 217–228.
20. Александров, А. С., Александрова Н. П., Долгих Г. В. Модифицированные модели для расчета главных напряжений в дорожных конструкциях из дискретных материалов // Строительные материалы. — 2012. — № 10. — с. 14–17.

Комбинированное использование солнечной энергии и тепловых отходов в гелиотеплице¹

Вардияшвили Афдандил Аскарлович, кандидат технических наук, доцент;

Мурадов Муслим Омар оглы, доцент;

Вардияшвили Асфандиёр Аскарлович, преподаватель;

Сайдалиев Азиз Акрамович, магистр

Каршинский государственный университет (Узбекистан)

Холов Комил Нормаматович, доцент

Каршинский инженерно-экономический институт (Узбекистан)

В мире проблемы энергосбережения при использовании топливно-энергетических ресурсов приобрели особую актуальность, что обусловлено сокращением запасов традиционных топливно-энергетических ресурсов и обострением экологических нагрузок в окружающую среду. В настоящее время до 40% добываемого в мире топлива расходуется на системы теплообеспечения зданий и поэтому в развитых странах мира, и в Узбекистане проводится комплекс мероприятий по решению проблем энергосбережения и развитие использование нетрадиционных возобновляемых источников энергии.

В соответствии с Постановлением Президента Республики Узбекистан от 5 мая 2015 года № ПП-2343 «О программе сокращению энергоёмкости, внедрению энергосберегающих технологий в отраслях экономики и социальной сфере на 2015–2019 годы» поставлены задачи, по определению приоритетных направлений дальнейшего сокращения энергоёмкости, внедрения энергосберегающих технологий и систем в отраслях экономики, ускоренное развитие использование воз-

обновляемых источников энергии (ВИЭ), в том числе апробированных технологий использования солнечной энергии [1].

Тепличное производство — это сложный технологический процесс, связанный с выращиванием живого организма. Выращивание растения — это постоянный процесс, заключающийся в обеспечении растения сбалансированным питанием, оптимальными условиями микроклимата, а также в уходе за ним.

В сухом веществе растений в среднем содержится 45% углерода, 1,5% азота, 5% зольных элементов. Из этого видно, что потребность растений в углероде намного превышает потребность в других элементах питания. На сегодняшний день в овощеводческих и цветочеводческих хозяйствах России остро стоит вопрос об осуществлении подкормок углекислым газом растений в защищенном грунте. Низкое содержание углекислого газа является фактором, ограничивающим урожайность.

Рассмотрим положительное влияние подкормки растений CO₂. Цель каждого, кто занимается тепличным хозяйством увеличить продуктивность растений и прирост

¹ Грант РУз А-4-47.

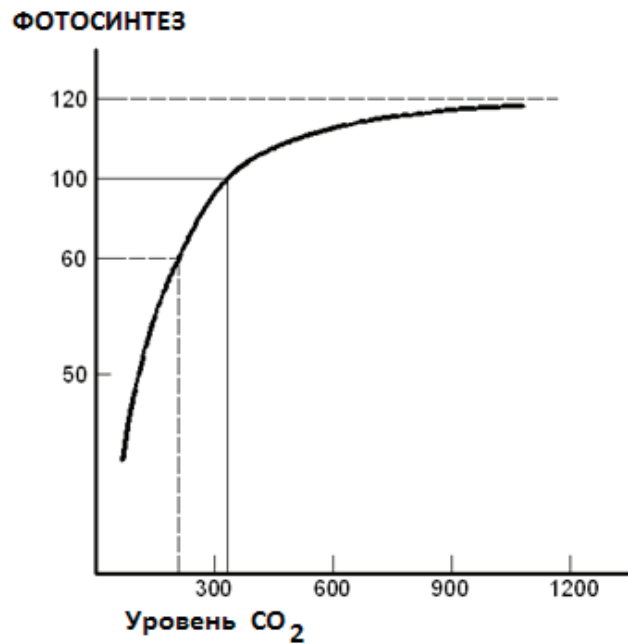


Рис. 1. Зависимость уровня фотосинтеза от уровня CO₂ в атмосферном воздухе

массы органического вещества. Углекислый газ усиливает рост растений и их мощь.

Вот только несколько примеров того, как CO₂ увеличивает продуктивность растений (рис. 1) [2]: у цветущих растений наступает более раннее цветение, урожайность плодов повышается, у роз реже происходит отмирание бутонов, растения дают более мощные побеги и более крупные цветы. Именно поэтому CO₂ можно назвать удобрением для растений. Дефицит CO₂ является более серьезной проблемой, чем дефицит элементов минерального питания — в среднем, растение синтезирует из воды и углекислого газа 94% массы сухого вещества, остальные 6% растение получает из минеральных удобрений.

У большинства растений при повышении уровня содержания углекислого газа в теплице с 0,034% до 0,1% рост и урожайность может увеличиться до 50%.

Государственный университет Колорадо провел исследования того, как влияет повышение уровня CO₂ в теплицах на выращивание цветов. Было замечено, что с ростом уровня CO₂ у цветов увеличивается рост массы вещества, количества бутонов и они имеют более привлекательный рыночный вид.

Было замечено следующее: Урожайность возросла на 39,7%, при повышении уровня CO₂ урожайность томатов возросла на 30%, плоды были более прочные, хорошей одинаковой формы.

Выращивание овощей в условиях защищенного грунта является весьма энергоёмким процессом, требующим значительных расходов дефицитных энергоресурсов. Снижение энергозатрат на эти цели, в том числе за счёт применения солнечной энергии и отходного тепла теплотехнического оборудования, являются одним из путей энергосбережения, особенно в условиях перехода к ры-

ночным отношениям. Расходы в теплоснабжении теплиц достигают существенных значений 45–60% составляющих себестоимости продукции. Это накладывает особую ответственность на выбор и проектирование системы отопления теплиц и тепличных комбинатов. Известно, что тепличное растение является обитателем одновременно двух сред — воздушного пространства и почвы.

Поэтому для растений параметры не только воздуха, но и тепличной почвы, имеют большое значение [2, 3, 4, 5].

Результаты последних исследований [2, 3, 4, 5] показывают, что изменения параметров почвы для растений имеют не менее важные жизненные значения, чем изменение температуры воздуха. А почва, через посредство которой можно эффективно воздействовать на урожай растений, более доступна для регулирования температуры чем воздух, поэтому обогрев корнеобитаемого слоя почвы заслуживает особого внимания.

Энергообеспеченность теплиц солнечной энергией с октября до апреля месяца в условиях Кашкадарьинской области составляет 35–40% в зависимости от метеорологических условий конкретного года [3].

Характеристики внешних воздействий, изменение теплотерм гелиотеплицы для отопительного периода в зависимости от прошедшей солнечной радиации и температуры наружного воздуха по многолетним данным представлен в виде графиков (рис.2). В условиях г. Карши дефицит тепла возникает только в ночное время осенью до третьей декады ноября, весной с третьей декады февраля. От третьей декады ноября до третьей декады февраля дефицит тепла ощущается и в дневное время.

Анализ работы и результаты натуральных экспериментов опытной гелиотеплицы показывают, что в теплице потери теплоты, особенно в зимнее время увеличиваются, что

требует значительных затрат энергии для поддержания нормируемой температуры.

Эти расходы могут быть значительно снижены, если комплексно использовать отработавшие газы теплотехнического оборудования.

Естественная концентрация CO_2 в атмосферном воздухе недостаточна для интенсивного фотосинтеза даже при высокой освещенности, т.е. для нормального развития растений в теплицах содержание CO_2 в воздухе должно быть в пределах 0,3–0,5%, а в атмосферном воздухе его содержание составляет около 0,03%. Подача углекислого газа и доведение уровня углекислоты в теплицах до требуемых значений, осуществляется вентилятором через воздухопроводы за счёт использования дымовых газов малой котельной, работающей на природном газе. С целью разработки технологии использования тепловых отходов традиционных теплоисточников для отопления теплиц, произведён теплотехнический расчёт по использованию дымовых газов малой котельной установки, где доля низкопотенциального (бросового) тепла, доходит до 55–60%. На основе теплогидравлического расчёта изготовлен воздухопровод для равномерной подачи углекислого газа и доведения её уровня в теплице до требуемых значений, где используется, не только тепло, но и сами охлажденные дымовые газы для подкормки выращиваемых растений содержащимся в них углекислым газом CO_2 , с целью повышения урожайности овощных культур.

Охлажденные дымовые газы применяются для подкормки растений углекислотой, содержание которой в отходящих дымовых газах котельных агрегатов составляет 10–12%. Для нормального развития растений в теплицах содержание CO_2 в воздухе должно быть в пределах 0,3..0,5%. Нужную концентрацию CO_2 можно получить, если подавать смесь продуктов сгорания с воздухом в камеру теплицы по схеме рис. 2 а. Специальный маломощный котел I установлен в коридоре теплиц, он оснащен горелками с низкой нагрузкой 10–15% от полной. Продукты сгорания газа перед подачей в камеру теплицы по газовому каналу 2 проходят через водяной теплообменник 3, где резко понижается температура газа, и с помощью сервомотора 4 открывают запорный клапан и включают центробежный вентилятор 7.

Остывшие дымовые газы при проходе через диафрагму 5 подсасывают через открытый канал 6 атмосферный воздух, смешиваются с ним, после чего концентрация CO_2 в смеси достигает требуемой величины 0,3–0,5%, и температура снижается до 320 К. Далее газоздушная смесь распределяется под небольшим давлением по грунтовым лотковым, теплоаккумулирующим каналом и вдоль рядов растений теплицы.

В теплообменнике 3 подогретая вода, используется для обогрева почвы и полива растений, а смесь продуктов сгорания газа и воздуха через распределительные каналы подается с помощью вентилятора 7 в теплицу для обогрева помещения. В воздуховоде в результате смешения продуктов сгорания природного газа и наружного атмосфер-

ного влажного воздуха по ходу их движения температура газоздушной смеси непрерывно уменьшается и происходит конденсация влаги. Интенсивность конденсации влаги в дымоходах зависит от влажности отходящих газов, скорости движения газоздушного потока и особенно от температуры на внутренней поверхности вытяжного дымохода $t_{\text{вн}}^{\text{дымх.}}$.

Значение $t_{\text{вн}}^{\text{дымх.}}$ может быть определено только в том случае, если известен закон изменения температуры отходящих дымовых газов по длине дымохода — воздуховода [3].

Вытяжной дымоход после центробежного вентилятора имеет длину L, а в площади поперечного сечения представляет собой круг с внутренним диаметром d_1 рис. 2

В этот дымоход — воздухопровод поступают отходящие дымовые газы с удельной теплоёмкостью C_p в количестве v_r и с температурой $t_{\text{yx}}^{\text{вх.}}$. Средний коэффициент теплопередачи воздуховода равен K. Процесс теплопередачи от дымохода в окружающую среду, имеющую температуру $t_{\text{вн.тепл.}}$ (температуру камеры теплицы), будем считать установившимся. В нашем случае требуется определить конечную температуру отходящих дымовых газов в каком-либо сечении x вытяжного дымохода t_{yx}^x . Для решения инженерной теплофизической задачи выделим по длине дымохода (воздуховода) элементарный участок с теплопередающей поверхностью dF. Обозначим температуру отходящих дымовых газов на этом участке через t_x .

Тепличное хозяйство развивается в основном около крупных промышленных центров (Ташкент, Самарканд, Бухара, Навои, Мубарек и др), так как здесь имеются дешёвые источники централизованного теплоснабжения, оказывающие небольшое влияние на окружающую среду. Используется централизованное теплоснабжение также для отопления теплично-овощных комбинатов в районах нового промышленного освоения. К примеру, в Шуртанском газо-химическом комплексе расширяются площади под тепличными хозяйствами.

Подкормки растений углекислым газом необходима на каждом тепличном комбинате или теплице в приусадебном хозяйстве, которые рассчитывают на экономический успех. Существующая практика доказывает, что технологии подкормки растений CO_2 имеют право на существование в условиях России. Даже подкормка прямой газацией, которая не отвечает потребностям технологии овощеводства защищенного грунта и требованиям ресурсо- и энергосбережения, занимает свои узкие области применения.

Предложено использование продуктов сгорания газового топлива для обогрева теплиц и подкормки растений углекислотой. Концентрация углекислого газа в воздухе внутри теплицы устанавливается на уровне 0,2 ÷ 0,4% при подаче отходящих дымовых газов от котельных из расчёта 10 ÷ 20 г на 1 м³ объема теплицы в день. В зимний период при интенсивности прошедшей солнечной радиации 250 ÷ 300 Вт/м² в 1 м³ объема теплицы допол-

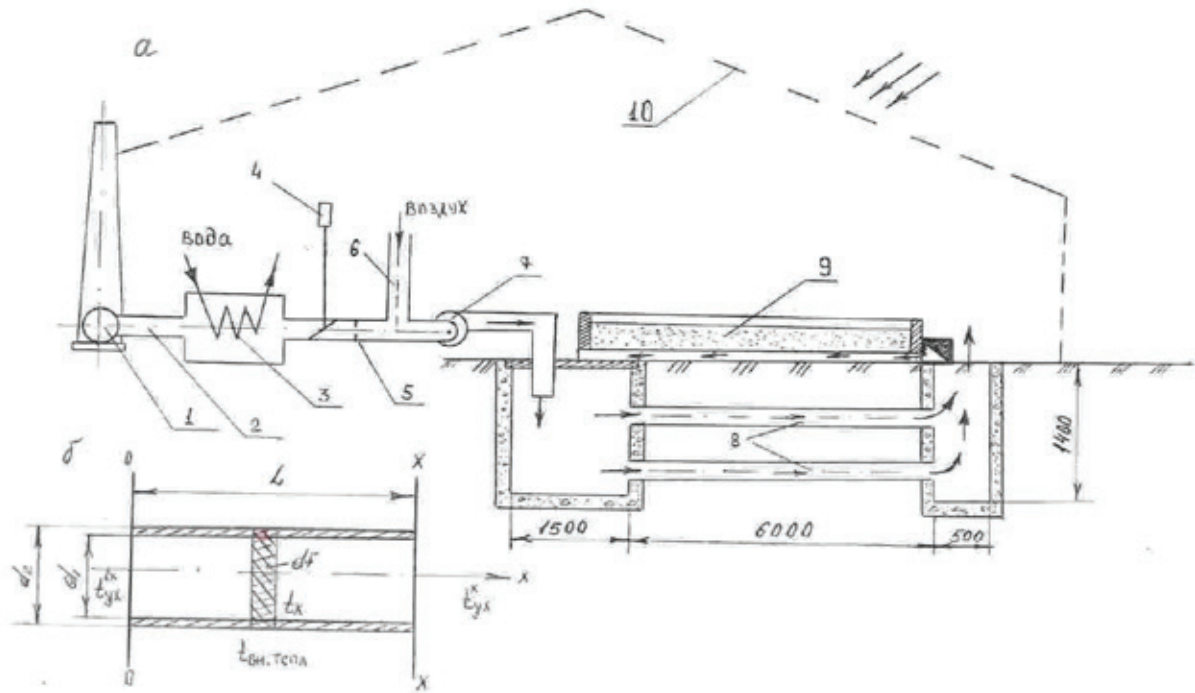


Рис. 2. Принципиальная схема использования отходного тепла для отопления теплицы и обогащения воздуха в нем двуокисью углерода (а) и расчетная тепловая схема дымохода круглого сечения (б). а) 1 — специальный котел; 2 — канал для дымовых газов; 3 — водяной теплообменник; 4 — сервомотор с запорным клапаном; 5 — ограничительная диафрагма; 6 — канал для подсоса воздуха; 7 — центробежный вентилятор; 8 — грунтовый подпочвенный аккумулятор тепла; 9 — лоток с субстратом; 10 — прозрачное покрытие теплицы

нительно за счет суммарной поглощательной способности смеси газов аккумулируется 14,67 Вт/м³ тепловой

энергии, а следовательно, в 200 м² × 3 м = 600 м³ объеме теплицы 8,8 кВт тепловой энергии.

Литература:

1. Постановление Президента Республики Узбекистан от 5 мая 2015 года № ПП-2343 «О программе сокращения энергоемкости, внедрению энергосберегающих технологий в отраслях экономики и социальной сфере на 2015–2019 годы».
2. Богданов, К.Б., Усков Е.И. Способы использования диоксида углерода в агропромышленном комплексе <http://ugle-kislota.narod.ru/podcormca.html>
3. Вардияшвили, А.А. “Разработка и исследования multifunctional энергоэффективных гелиотехнических комплексов с использованием энергетических отходов”. Монография. Карши «Насаф» 2013 г. 9,6 б.т.
4. Промышленное производство овощей в теплицах/ Под ред. Вашенко С. — М.: 1977.
5. Вардияшвили Аф.А. Исследование теплотехнических и вентиляционных параметров солнечной теплицы с использованием тепловых отходов традиционных источников: Междун.науч.ж.//Наука, образование, техника. — Кыргызтан, г. Ош. 2008 г. с. 174–177.

Применение щебеночно-мастичной асфальтобетонной смеси при строительстве покрытий в Алтайском крае

Великанова Валерия Андреевна, магистрант;
 Степанец Виктор Георгиевич, кандидат технических наук, доцент
 Сибирская государственная автомобильно-дорожная академия

В статье кратко рассматривается проблема повышения эффективности и качества дорожных покрытий, а так же актуальность проблемы обеспечения высоких эксплуатационных характеристик покрытия, ко-

торая подтолкнула учёных к разработке и внедрению современных композиционных материалов при строительстве верхних слоёв покрытия автомобильных дорог.

Ключевые слова: щебеночно-мастичная асфальтобетонная смесь, щебеночно-мастичный асфальтобетон, стабилизирующая добавка

Верхний слой покрытия является самостоятельным и важнейшим конструктивным элементом дороги. Он должен обеспечивать повышенный уровень безопасности, устойчивости и комфортности движения, требуемую скорость, простоту содержания и сохранность дорожных покрытий.

В настоящее время применение щебеночно-мастичного асфальтобетона является самым прогрессивным методом восстановления асфальтобетонного покрытия.

ЩМА используется в качестве верхнего дорожного покрытия в аэропортах, на мостах и в речных портах в таких странах, как США, ЮАР, Китай, Норвегия, Финляндия, Швеция, Франция, Германия и множестве других

Щебеночно-мастичный асфальтобетон (ЩМА) — это горячая асфальтобетонная смесь, состоящая из щебеночного каркаса, в котором все пустоты между крупным щебнем заполнены смесью битума с дробленным песком и минеральным порошком [1].

В структурном отношении щебеночно-мастичный асфальтобетон отличается от других типов смесей на столько, что его относят к самостоятельной группе дорожно-строительных материалов.

Основное отличие ЩМА от обычных асфальтобетонов заключается в его жесткой каркасной структуре в слое покрытия. Такая структура обеспечивает передачу нагрузки с поверхности в нижележащие слои через непосредственно контактирующие друг с другом отдельные крупные частицы каменного материала. Тем самым достигается существенное снижение деформаций слоя покрытия. В ЩМА основную структуру составляет фракционированный щебень (70–80% по массе) с улучшенной (кубовидной) формой зерен, а мелкий служит только для образования мастики, заполняющей межкаменное пространство в щебеночном каркасе. Кроме того, высокое содержание крупной фракции каменного материала в ЩМА позволяет получить шероховатую поверхность покрытия, что обеспечивает требуемые значения коэффициента сцепления колеса с покрытием.

Сдвигустойчивость покрытия из ЩМА, характеризующая сопротивление колееобразованию, обеспечивается требуемым значением коэффициента внутреннего трения. Поэтому в песчаной части смеси используется только песок из отсевов дробления горных пород, так как природный песок снижает коэффициент внутреннего трения.

Еще одной особенностью ЩМА является повышенное содержание битума (5,5... 7,5%). Большое количество вяжущего препятствует проникновению влаги внутрь слоя, повышает трещиностойкость, водоморозостойкость и увеличивает срок эксплуатации покрытия.

Однако повышенное содержание битумного вяжущего в смеси нужно стабилизировать, то есть предотвратить его отслоение и стекание с поверхности зерен щебня при высоких технологических температурах приготовления, хранения, транспортирования и укладки. Данная проблема легко решается введением в смесь стабилизирующей добавки. Основная цель применения стабилизирующих добавок заключается в повышении толщины битумных пленок, обеспечивающих присутствие свободного битума и однородности ЩМАС, например целлюлозного волокна.

Так же спецификой смеси ЩМА является, в частности, более высокая, по сравнению с обычными асфальтобетонными смесями, температура приготовления. Это связано с температурной чувствительностью смеси и с тем, что ЩМА укладывается в основном тонкими слоями, склонными к быстрому охлаждению.

Щебеночно-мастичный асфальт положительно зарекомендовал себя во всем мире во всех климатических зонах в течение более 25 лет как долговечный износостойкий вариант асфальтового покровного слоя на дорогах с интенсивным движением.

В данной статье рассматривается вопрос использования ЩМА на федеральной автомобильной дороге «А-322 Барнаул — Рубцовск» в Алтайском крае. Автомобильная дорога идёт из Барнаула в юго-западном направлении вдоль левого берега реки Алей до границы с Казахстаном в 40 км к югу от Рубцовска.

Климат Алтайского края имеет ярко выраженные черты континентальности: здесь холодная, длительная, снежная зима и короткое, теплое, иногда жаркое лето.

Наибольшее количество осадков выпадает в Красногорском, Алтайском и Солонешенском районах, наименьшее — в Угловском и Рубцовском (западная часть) районах. Для северной части типично недостаточное увлажнение, теплое лето и умеренно суровая малоснежная зима. Южная (горная) часть достаточно увлажнена, лето умеренно теплое, зима умеренно суровая, снежная.

Рубцовский район, через который и проходит автомобильная дорога отличается равнинной местностью, по правую сторону реки расположены низменные и заливные луга. Так же в Рубцовском районе добываются песок, гравий, щебень, полиметаллические руды, что очень благоприятно сказывается при использовании в строительстве щебеночно-мастичных асфальтобетонных смесей.

Данная автомобильная дорога является федеральной трассой, следовательно интенсивность движения транспорта, включая большегрузные автомобили, очень высока. Результатом этого является возникновение колеи, истирание дорожного покрытия, появление сетки трещин, шероховатость покрытия уменьшается.

Применение ЩМА значительно улучшает транспортно-эксплуатационные показатели данной автомобильной дороги.

Конструкция дорожной одежды при капитальном ремонте представлена на рисунке 1.

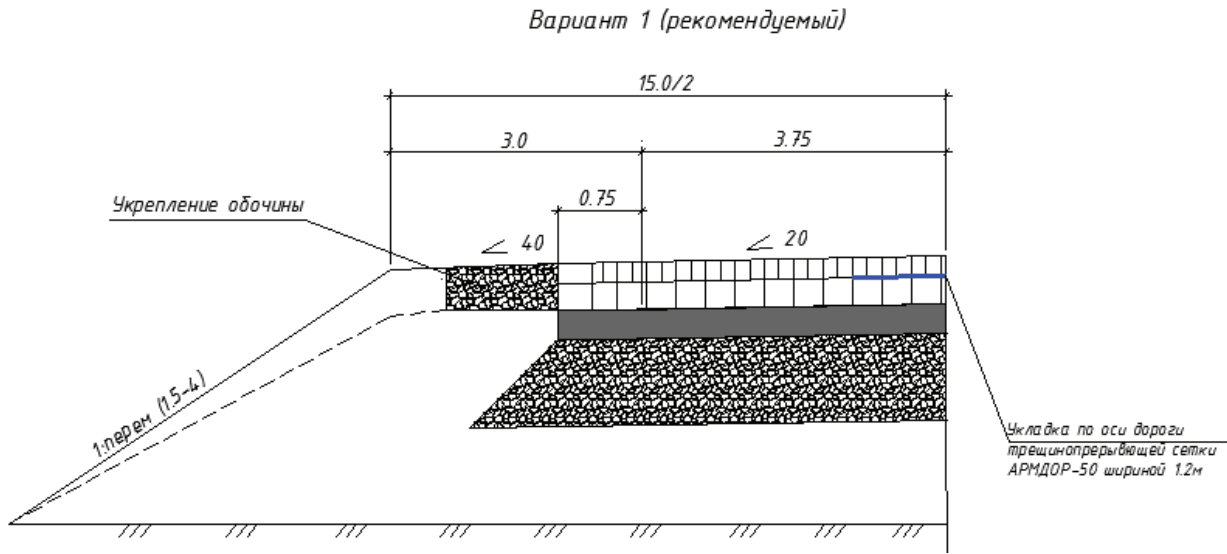


Рис. 1

- Щебеночно-мастичные асфальтобетоны обладают:
- высокой устойчивостью к переменным деформациям;
 - высокой устойчивостью к разрушениям под воздействием транспортного движения и климатических условий;
 - великолепными эксплуатационными характеристиками (коэффициент сцепления, отсутствие неровностей и т.п.);
 - при определенных условиях ЩМА уменьшает шум от движения транспорта по сравнению с альтернативными материалами.

Известно, что стоимость щебеночно-мастичных смесей больше, чем стоимость традиционных асфальтобетонных смесей на 30–35%, но в покрытии используется слой ЩМА на 35–40% меньше, что обеспечивает

снижение расхода смеси. Поэтому стоимости сравниваемых покрытий практически равны. Кроме того при должном качестве строительства, применение ЩМА обеспечивает снижение затрат на содержание и ремонт дорожных покрытий в 2...4 раза, повышение их долговечности в 2–3 раза [4].

Производство ЩМА ведется на стандартном оборудовании для получения горячих асфальто-бетонных смесей, но тем не менее производство и укладка имеют свои специфические особенности.

Нами на кафедре строительства и эксплуатации дорог СибАДИ совместно с лабораторией ОАО «Стройсервис» были выполнены работы по подбору состава смеси ЩМА-15.

Данные по подбору состава приведены в таблице 1 и представлены на рисунке 2.

Таблица 1

Содержание, %	20	15	10	5	3	1,25	0,63	0,315	0,14	0,071
70	100,0	92,5	42,4	4,25	0,0	0,0	0,0	0,0	0,0	0,0
18	100,0	100,0	100,0	98,2	77,49	49,93	35,14	24,63	19,90	12,25
12	100,0	100,0	100,0	100,0	100,0	100,0	100,0	98,63	96,81	77,11
Зерновой состав смеси	100,0	100,00	59,66	32,65	25,95	20,99	18,33	16,27	15,20	11,46
Граничные значения		90,00	40,00	25,00	18,00	15,00	12,00	10,00	9,00	9,00
по ГОСТ 9128-97		100,00	60,00	35,00	28,00	25,00	22,00	20,00	16,00	14,00

На основании подбора состава установлено процентное соотношение между компонентами щебеночно-мастичной смеси, при этом, количество щебня со-

ставляло — 70%, песка из отсевов дробления — 18%, а минерального порошка — 12%.

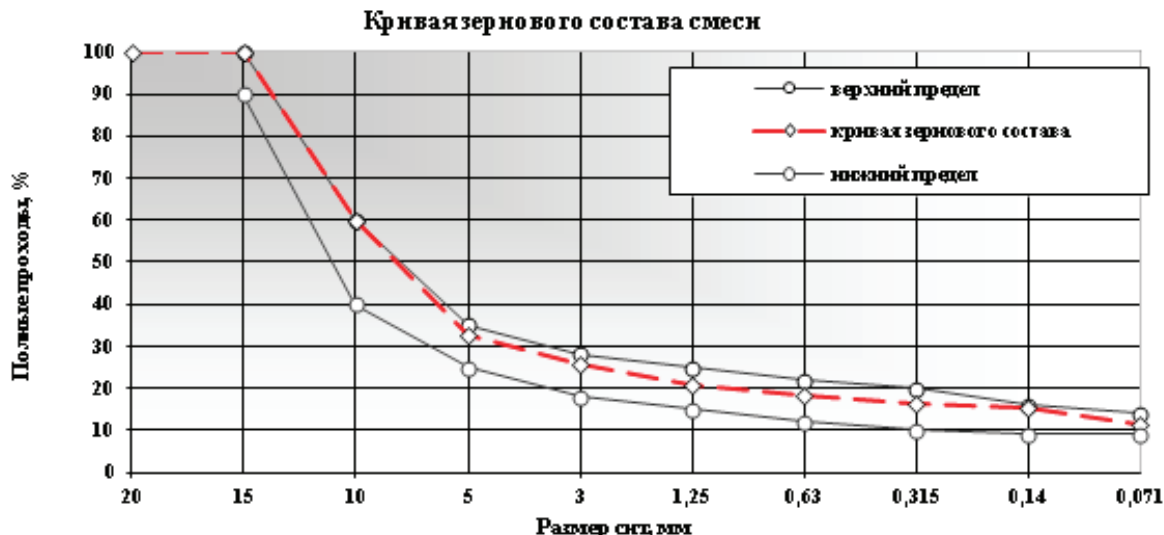


Рис. 2

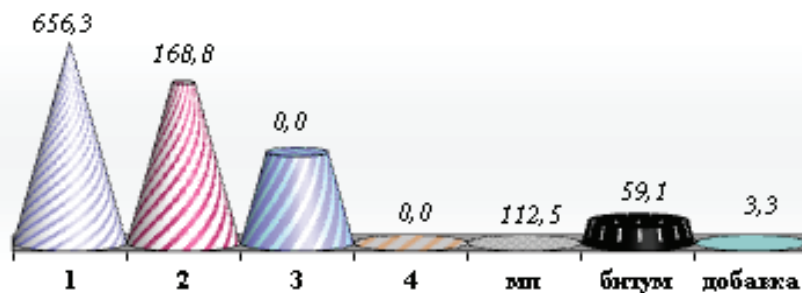
Дозировка битума марки ПБВ 90 составляет 6,3%, и стабилизирующей добавки, марки Wetfix BE составляет 0,2%.

Показатели физико-механических свойств данной смеси, приведены в таблице 2

Таблица 2. Физико-механические показатели щебеночно-мастичного асфальтобетона

№ п/п	Наименование показателей	Требования ГОСТ 31015-2002	Фактические показатели
1.	Средняя плотность, г/см ³	-	2,33
2.	Пористость минерального остова, %	от 15 до 19	17,67
3.	Остаточная пористость, %	от 1,5 до 4,5	3,72
4.	Водонасыщение, % по объёму	от 1,0 до 4,0	3,6
5.	Предел прочности при сжатии, МПа:		
	при 20°С	не менее 2,2	2,5
	при 50°С	не менее 0,65	1,0
6.	Сдвигостойчивость:		
	коэффициент внутреннего трения	не менее 0,93	0,96
	сцепление при сдвиге при температуре 50°С, МПа	не менее 0,18	0,23
7.	Трещиностойкость — предел прочности на растяжение при расколе при температуре 0°С, МПа	от 2,5 до 6,0	3,9
8.	Водостойкость при длительном водонасыщении	не менее 0,85	0,91
9.	Сцепление вяжущего с минеральной частью шма смеси	не менее ¾ (75%) поверхности смеси должно остаться покрытой пленкой	выдерживает более 85%

Потребность материала для приготовления 1 тонны смеси, кг. представлена на рисунке 3



Примечание: Потребность в инертных материалах приведена с учетом негабарита

Рис. 3. Примечание: 1 — щебень, 2 — песок из отсевов дробления

Количество материалов по фракциям на 1 замес представлено на рисунке 4

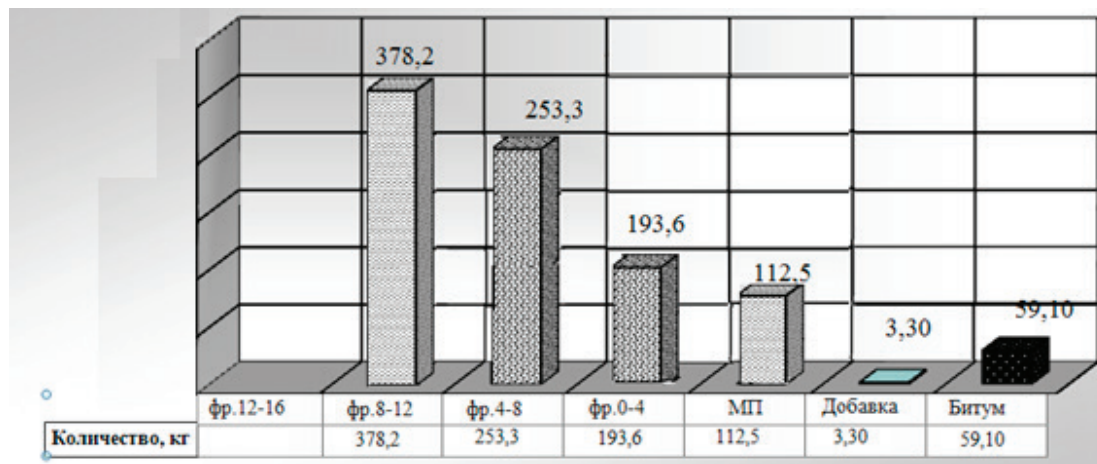


Рис. 4. Примечание: Величина замеса составляет 1000 кг

На основе выполненных исследований, подобран состав ЩМА-15 который рекомендуется при строительстве

верхнего слоя покрытия на федеральной автомобильной дороге «А-322 Барнаул — Рубцовск» в Алтайском крае.

Литература:

1. ГОСТ 31015—2002.Смеси асфальтобетонные и асфальтобетон щебеночно-мастичный. Технические условия.
2. СНиП 3.06.03—85. Автомобильные дороги.
3. ГОСТ 12801—98. Материалы на основе органических вяжущих для дорожного и аэродромного строительства. Методы испытаний.
4. Щебеночно-мастичный асфальтобетон для дорожных покрытий. Учебное пособие. Костин В. И.

Обеспечение и подтверждение надежности модульных транспортных средств

Виноградов Олег Владимирович, кандидат технических наук, доцент;

Кирсанов Евгений Алексеевич, аспирант

Российский государственный аграрный университет — Московская сельскохозяйственная академия имени К. А. Тимирязева

В статье рассмотрены вопросы, связанные с обеспечением надежности транспортных средств модульной конструкции, ее задачами. Выявлены проблемы в данной области и найдены условия для достижения достаточной надежности модульных транспортных средств.

Ключевые слова: транспорт, модуль, испытания, надежность, ремонтпригодность

Автомобилестроение использует понятие «модуль» со следующим значением: модуль — система механизмов, обладающая заданными входными и выходными параметрами, способная дополнять или изменять функции основного механизма. Применение модульности в производстве автомобилей дает производителям несколько преимуществ: возможность создания широкой гаммы модификаций на базе одной проработанной модульной платформы, унификация конструкции разных моделей, соответственно сокращение номенклатуры деталей и затрат

на них, возможность резко ускорить разработку и выпуск новых модификаций на базе старой конструкции, необходимость модернизации возникает тогда, когда автомобиль уже «морально устарел» и т.д. Однако массовое производство автомобилей на модульных платформах, которое разовьется полностью в ближайшие десять лет (на него шаг за шагом уже начал переходить «Nissan»), таит в себе и определенные опасности. Так, в случае технического сбоя в производстве какого-то узла придется отзываться на гарантийный ремонт все модели, использу-

ющие данную версию модульной платформы, а их могут быть сотни тысяч и даже миллионы автомобилей [7].

В связи с вышесказанным встает вопрос о том, насколько могут быть надежны модульные транспортные средства и каким образом это можно подтвердить, поскольку не вполне ясно, как повлияет модульная сборка на стоимость владения, насколько модули ремонтпригодны — не придется ли каждый раз менять не узел, но модуль.

Вопросами надежности автомобилестроение обеспечено с момента создания автомобиля, поэтому вначале проясним, что же мы закладываем в понятие «надежность». Итак, надежность — это свойство объекта выполнять заданные функции, сохраняя значения установленных эксплуатационных показателей в заданных пределах, соответствующих заданным режимам и условиям использования, в течении требуемого промежутка времени или требуемой наработки [1, 8]. Количественно надежность оценивается показателями безотказности, долговечности, ремонтпригодности и сохраняемости [2]. Безотказность — свойства объекта непрерывно сохранять работоспособность в течение некоторого времени или некоторой наработки. Долговечность — свойства объекта сохранять работоспособность до наступления предельного состояния при установленной системе технического обслуживания и ремонта. Ремонтпригодность — свойства объекта, заключающиеся в приспособленности к предупреждению и обнаружению причин возникновения его отказов, повреждений и устранению их последствий путем проведения ремонтов и технического обслуживания [3]. Технический уровень любого транспортного средства оценивают конструктивными, производственными (технологическими) и эксплуатационными показателями. К эксплуатационным показателям, характеризующим техническое состояние транспортных средств, относятся в первую очередь надежность и контролепригодность [4]. Надежность транспортного средства закладывается при ее проектировании, обеспечивается при изготовлении и проявляется в процессе эксплуатации. Таким образом, мы видим, что надежность закладывается в автомобиль конструктивно, при расчетах. При проектировании традиционных автомобилей конструкторам заранее известны цели и задачи будущего автомобиля, и главное — принцип компоновки и будущие модификации. Закладывать в конструкцию рассчитанный ресурс автомобиля приходится, зная наверняка, какие узлы и агрегаты, с какими характеристиками будут работать в связке. И это несколько упрощает ситуацию, поскольку круг модификаций заранее определен и ограничен. Разработчикам останется только подтвердить заявленные цифры ресурса различными испытаниями полнокомплектного автомобиля.

Перед конструкторами модулей для будущего автомобиля стоит более трудная задача. Вначале необходимо

определить предельно допустимые значения нагрузок и рабочих характеристик модуля при работе в предполагаемых условиях. Предстоит учесть всевозможные сферы использования данного модуля в составе транспортного средства и заложить необходимые показатели в конструкцию. После проектирования, как и в случае с традиционными транспортными средствами, необходимо подтвердить надежность данного модуля. Существует два способа подтверждения надежности: расчетный и практический — испытания.

Применяются три основных метода проведения испытаний на надежность [5]: 1) в автомобильных хозяйствах с перевозкой реальных грузов (эксплуатационные испытания); 2) в испытательных организациях с проведением пробеговой части испытаний на дорогах общего пользования при загрузке автомобилей балластом (ресурсные испытания); 3) в условиях испытательного полигона с пробегами по специальным дорогам различных типов при загрузке автомобилей балластом (полигонные ресурсные испытания). В первом случае условия испытаний очень близки к условиям обычной эксплуатации. Недостатком обычно является невысокий суточный пробег и соответственно длительные сроки испытаний. Во втором случае существенно сокращаются сроки испытаний, однако при этом не учитываются многие факторы, влияющие на работу автомобиля, его агрегатов и систем, в данном случае готовых модулей, в реальной эксплуатации. При проведении испытаний на автомобильном полигоне используются специальные дороги и сооружения. Создаваемые при этом более тяжелые (форсированные) режимы работы автомобиля позволяют в несколько раз сократить пробег, а следовательно, сроки и стоимость испытаний [6]. Однако вследствие специфичности условий этих испытаний необходимо знать переходные коэффициенты для приведения полученных результатов испытаний к условиям реальной эксплуатации на дорогах общего пользования. К форсированным или ускоренным полигонным испытаниям предъявляется одно важнейшее требование — получаемые отказы и неисправности должны быть типичными, т.е. аналогичными встречающимся в эксплуатации. Ситуация усложняется тем, что при проектировании модульного ТС необходимо подтвердить надежность не просто автомобиля целиком, а каждого модуля, с учетом его применимости в других моделях. Это и есть одна из главных проблем, поскольку, как уже отмечалось выше, применение модуля ограничится не одной моделью, а возможно целым рядом модификаций для различных целей и задач.

В итоге удастся получить проработанные модули транспортного средства для применения в готовых автомобилях, тем самым конструкторы получат больше свободы при создании или обновлении модельного ряда, а взаимозаменяемость модулей благоприятно отразится на ремонтпригодности автомобилей, а, следовательно, и надежности.

Литература:

1. Дидманидзе, О.Н. Ресурсосбережение на автомобильном транспорте и в сельскохозяйственном производстве. / В сб.: Научные проблемы автомобильного транспорта: материалы Международной научно-практической конференции, посвященной 80-летию ФГОУ ВПО МГАУ, 2010. — с. 5–8.
2. Дидманидзе, О.Н., Асадов Д.Г., Карев А.М., Егоров Р.Н., Журилин А.Н. Транспортные и транспортно-технологические процессы. — М.: ООО «УМЦ «Триада», 2016. — 163 с.
3. Дидманидзе, О.Н., Рыбаков К.В., Митягин Г.Е. и др. Автотранспортные и тракторные перевозки: учебник — М.: УМЦ «Триада», 2005. — 552 с.
4. Карев, А.М., Пуляев Н.Н., Егоров Р.Н., Журилин А.Н. Автотранспортные процессы и системы. — М.: ООО «УМЦ «Триада», 2016. — 94 с.
5. Куров, Б.А., Лаптев С.А., Балабин И.В. Испытания автомобилей. — М.: Машиностроение, 1976. — 208 с.
6. Лаптев, С.А. Комплексная система испытаний автомобилей: Формирование, развитие, стандартизация. — М.: Издательство стандартов, 1991. — 172 с.
7. Материалы с сайта www.modulcar.h1.ru.
8. Дидманидзе, О.Н., Варнаков Д.В. Повышение параметрической надежности автомобильных двигателей // Ремонт, восстановление, модернизация. — 2007. — № 5. — с. 2–7.

Применение методов исследований и испытаний традиционных транспортных средств для испытаний модульных транспортных средств

Виноградов Олег Владимирович, кандидат технических наук, доцент;

Кирсанов Евгений Алексеевич, аспирант

Российский государственный аграрный университет — Московская сельскохозяйственная академия имени К. А. Тимирязева

В статье рассмотрены вопросы, связанные с применением традиционной методики испытаний к транспортным средствам модульной конструкции, ее задачами. Выявлены проблемы в данной области и найдены условия для достижения достаточной достоверности испытаний модульных транспортных средств.

Ключевые слова: транспорт, модуль, испытания, надежность, методика

Автомобилестроение является одним из наиболее сложных видов массового машиностроения. К автомобилям, наряду с требованиями функционального соответствия транспортному процессу, предъявляются высокие требования по технологичности, надежности и экономичности в многообразных условиях эксплуатации, по экологическим и эргономическим показателям, эстетическим качествам и их соответствию тенденциям автомобильного дизайна и т.д. Работа автомобиля характеризуется сложными процессами его взаимодействия с внешней средой и процессами, происходящими в его двигателе, агрегатах и системах при движении в широком диапазоне режимов, определяемых конкретными условиями эксплуатации [1]. По этим причинам для успешного решения проблем, возникающих при создании новых и совершенствовании находящихся на производстве моделей автомобилей, наряду с поисками оптимальных конструкторских решений за монитором компьютера, с использованием классических и новых расчетных методов, возможности которых резко возрастают благодаря применению современных компьютерных технологий, совершенно необходимым является широкое

развертывание исследований и испытаний, ставших неотъемлемым звеном общего процесса создания новой автомобильной техники и образовавших одну из важных областей комплекса прикладных наук в машиностроении [2]. В автомобилестроении термином «испытание машин» обозначается «экспериментальное определение конструктивных и эксплуатационных свойств машин для выявления их соответствия техническим требованиям или для опытного изучения реальных процессов, происходящих в машинах». В отличие от этого «научное исследование» — это «процесс выработки новых научных знаний, один из видов познавательной деятельности». В практической работе, применительно к условиям промышленности, между этими двумя понятиями не всегда возможно провести четкую границу [3]. И в автомобилестроении испытания автомобилей, их агрегатов, а теперь и модулей при необходимости нередко могут содержать элементы исследования, а исследования, в свою очередь, иметь разделы, проводимые по установленным, стандартным методикам, т.е. элементы испытаний.

Существенная проблема применения существующей системы испытаний и исследований транспортных средств

в отношении модульных транспортных средств, состоит в первую очередь в том, что проектируемый модуль автомобиля заранее рассчитан на применение его в различных модификациях автомобилей и возможно даже отличных друг от друга сферах работы, в силу этого, проводя испытания какой либо одной модификации, весьма трудно искусственно создать тот диапазон нагрузок, который предусматривает вся гамма модификаций с этим модулем [5]. Отсюда напрашивается вывод, о том, что для применения имеющихся на данный момент стандартов и приемов испытаний и исследований к модульным автомобилям, на первых этапах необходимо проведение испытаний всех модификаций новой модульной платформы, а затем исследований и получение целого ряда коэффициентов нагрузок при работе в различных модификациях и условиях, и уже после этого возможно применение математического моделирования показателей работы того или иного модуля, в зависимости от типа транспортного средства, в котором он используется. Но это пока только теория. Для полноценного использования всех преимуществ модульных платформ разработчикам и испытателям автомобилей придется проводить множество расчетов, поиск всевозможных корреляций и т.д.

Функциональные качества автомобиля могут рассматриваться как сумма отдельных, тесно взаимосвязанных эксплуатационных свойств: тягово-скоростных, тормозных, топливно-экономических, прочностных и др. [4, 7]. Для определения конкретных значений показателей этих свойств с целью их доведения до заданных (проектных) значений у опытных образцов новых и модернизированных моделей, контроля качества или улучшения — у автомобилей текущего производства, проводятся испытания, виды которых охватывают полностью или частично широкую номенклатуру эксплуатационных свойств автомобилей различных типов и назначений. От результативности испытаний и темпов их проведения непосредственно зависят не только уровень отработки образцов автомобилей, подготавливаемых к производству, но и общие сроки выполнения конструкторско-экспериментальных работ и подготовки производства, а следовательно, сроки внедрения новой техники. Чтобы быть эф-

фективными и успешно решать поставленные перед ними задачи, испытания должны быть комплексными и образовывать целенаправленную систему, опирающуюся на научно обоснованные методы и оснащенную необходимым оборудованием.

Все вышесказанное в полной мере относится и к модульным автомобилям. Существующая система различных видов испытаний подлежит в определенной мере адаптации к особенностям модульных транспортных средств. Безусловно отдельные стандарты финишного контроля соответствия транспортных средств могут не претерпеть значительных изменений, а некоторые программы испытаний, например на надежность, топливную экономичность и другие эксплуатационные свойства могут потребовать различной степени доработки [6]. Безусловно этому должны предшествовать полномасштабные испытания различных модификаций определенной модульной платформы, как уже отмечалось выше.

В настоящий момент в нашей стране соответствие автомобилей различным требованиям безопасности регулирует Технический регламент «О безопасности колесных транспортных средств». В данном документе перечислены актуальные международные требованиями, представленные в виде Правил ЕЭК ООН и действующие национальные стандарты. На данный момент действующий документ в отношении безопасности транспортных средств не предусматривает разделения транспортных средств на «традиционные» и «модульные», в определенной степени схожей ситуацией с модульными автомобилями существует пока одна процедура, имеющая место в сертификации грузовых автомобилей — получение Одобрения типа шасси (ОТШ). В данном случае речь идет о документе на так называемую «базу» или «платформу», дальнейшие работы по установке на данную платформу различных механизмов, комплексов и т.п. требуют дополнительной процедуры сертификации, а соответственно и испытаний. Но это лишь пример из существующей практики, однозначно нельзя утверждать, по какому именно принципу будут проходить испытания, исследования и сертификация перспективных модульных транспортных средств, но вместе с тем стоит отметить, что это уже вопрос не будущего, а настоящего.

Литература:

1. Дидманидзе, О. Н. Ресурсосбережение на автомобильном транспорте и в сельскохозяйственном производстве. / В сб.: Научные проблемы автомобильного транспорта: материалы Международной научно-практической конференции, посвященной 80-летию ФГОУ ВПО МГАУ, 2010. — с. 5–8.
2. Дидманидзе, О. Н., Асадов Д. Г., Карев А. М., Егоров Р. Н., Журилин А. Н. Транспортные и транспортно-технологические процессы. — М.: ООО «УМЦ «Триада», 2016. — 163 с.
3. Дидманидзе, О. Н., Рыбаков К. В., Митягин Г. Е. и др. Автотранспортные и тракторные перевозки: учебник — М.: УМЦ «Триада», 2005. — 552 с.
4. Карев, А. М., Пуляев Н. Н., Егоров Р. Н., Журилин А. Н. Автотранспортные процессы и системы. — М.: ООО «УМЦ «Триада», 2016. — 94 с.
5. Куров, Б. А., Лаптев С. А., Балабин И. В. Испытания автомобилей. — М.: Машиностроение, 1976. — 208 с.
6. Лаптев, С. А. Комплексная система испытаний автомобилей: Формирование, развитие, стандартизация. — М.: Издательство стандартов, 1991. — 172 с.

7. Дидманидзе, О.Н., Варнаков Д.В. Повышение параметрической надежности автомобильных двигателей // Ремонт, восстановление, модернизация. — 2007. — № 5. — с. 2–7.

Обзор традиционных и современных расчетов дорожных конструкций

Гаевская Вера Алексеевна, магистрант
Сибирская государственная автомобильно-дорожная академия

В статье приведены краткие сведения о традиционных и современных методах расчета дорожных конструкций, включающих земляное полотно и дорожную одежду. Показаны преимущества современных методов расчета.

Ключевые слова: дорожная конструкция, деформация, напряжения

В соответствии с общепринятыми представлениями дорожные конструкции с одеждами нежесткого типа рассчитываются по прочности, на морозоустойчивость и по условиям дренирования дренажных слоев и устройств [1]. Учитывая многообразие методов проектирования дорожных конструкций, автор рассмотрит только расчеты на прочность. К таким расчетам относят: расчет дорожной конструкции в целом по допускаемому упругому прогибу, проверку условия сопротивления сдвигу в грунте земляного полотна и в слоях дорожных одежд из слабосвязных материалов, а также проверку условия сопротивления усталостному разрушению от растяжения при изгибе в слоях из монолитных материалов.

Первые методы расчетов по упругому прогибу в РФ были созданы И. И. Ивановым [2], а за рубежом Д. Бурмистром [3], которые в последующем постоянно совершенствовались и в нормативном документе [1] применяется решение, полученное М.Б. Корсунским [4]. В настоящее время целью работ, выполняемых в области совершенствования этого метода, является учет динамического характера приложения нагрузки. В результате созданы точные [5, 6] и инженерные [7–9] методы расчета по критерию допускаемого динамического прогиба. Наряду с расчетами дорожных одежд по критерию упругого прогиба стали развиваться методы, основанные на прогнозировании изменения ровности покрытий [10], в основе которых лежит расчет пластических деформаций, накапливаемых дискретными [11–17] и монолитными материалами [18, 19]. В результате получены различные зависимости для расчета пластических деформаций, которые имеют вид:

– для дискретных материалов (один из вариантов) [17]

$$\varepsilon_N = \frac{\sigma_1 - 2 \cdot \mu \cdot \sigma_3}{E_{Пн}} \cdot \left[1 + \kappa_1 \cdot \frac{(N^{\kappa_2 + 1} - 1)}{\kappa_2 + 1} \right] \quad (1)$$

Где $\sigma_1, \sigma_2, \sigma_3$ — главные напряжения в наиболее опасной точке земляного полотна, Па; $E_{Пн}$ — нелинейный продольный модуль пластической деформации, определяемый экспериментально по данным трехосных испытаний и являющийся аналогом модулей упругости и деформации. Па; κ_1 и κ_2 — коэффициенты, учитывающие уровень напряженного состояния и вид материала; N — число нагрузок.

$$E_{Пн} = M \cdot p_a \cdot \left[\left(\frac{\sigma_1 + 2 \cdot \sigma_3}{p_a} \right)^2 + \frac{6 \cdot (1 + \mu)}{1 - 2 \cdot \mu} \cdot \frac{(\sigma_1 - \sigma_3)^2}{3 \cdot p_a^2} \right]^\lambda \quad (2)$$

где p_a — атмосферное давление; a, M и l — параметры грунта, определяемые по данным трехосных испытаний, зависящие от показателей физических свойств (плотность, влажности) и гранулометрического состава.

– для дискретных материалов (один из вариантов) [18, 19]

$$U_{наб} = \frac{K_{дин} \cdot \sigma_1 \cdot (1 - \mu_{аб}^2) \cdot (1 + \kappa \cdot \ln N)}{E_{наб}} \quad (3)$$

где $K_{дин}$ — максимальное значение динамического коэффициента; $\mu_{аб}$ — коэффициент Пуассона асфальтобетона; $E_{наб}$ — продольный модуль, характеризующий величину пластической деформации, МПа.

В основе методов проектирования дорожных конструкций по сопротивлению сдвигу лежит решение А. М. Кривисского, базирующееся на оригинальном условии пластичности Кулона – Мора. Основы этого метода подробно изложены в нормативе [1], а так же обобщающей монографии [4]. Современные методы, преследующие цель совершенствования традиционного решения А. М. Кривисского и общепринятого метода [1] выполняются в различных направлениях. К методам первого направления отнесем работы, в которых предприняты попытки замены традиционного условия пластич-

ности Кулона — Мора другим критерием, например, трехпараметрическим критерием Кулона–Мора [20–24], или третьей теорией прочности [25], а так же одним из эмпирических условий [26–33]. Уравнения предельного состояния по этим критериям даны в табл. 1.

Таблица 1. Уравнения предельного состояния критериев сопротивления сдвигу

Наименование критерия	Уравнение предельного состояния
Эмпирический критерий Г. К. Арнольда [26, 31–33]	$\frac{1}{2} \cdot \left(\sigma_1 - \frac{1 + \sin\varphi}{1 - \sin\varphi} \cdot \sigma_3 \right) = c$, где c и φ — сцепление и угол внутреннего трения.
Трехпараметрические критерии Кулона–Мора [20–24]	$\sigma_1 \cdot \left(\frac{1 - \sin\varphi}{1 + \sin\varphi} \right)^d - \left(\frac{1 + \sin\varphi}{1 - \sin\varphi} \right)^d \cdot \sigma_3 = c$, $\frac{1}{2} \cdot \left(\frac{1 - \sin\varphi}{1 + \sin\varphi} \right)^b \cdot \left(\sigma_1 - \left(\frac{1 + \sin\varphi}{1 - \sin\varphi} \right) \cdot \sigma_3 \right) = c$.

К методам второго направления совершенствования расчетов по сопротивлению сдвигу отнесены работы экспериментального характера, благодаря которым установлено влияние повторности приложения нагрузок на величину сцепление и угла внутреннего трения [34–36].

Из анализа формул (1), (2) и зависимостей, представленных в табл. 1 следует, что важным элементом расчетов является вычисление главных напряжений, которое выполняют по традиционным формулам [37] или по модифицированным моделям [38–40], в которых использован новый метод расчета минимального главного напряжения [41–43]. Эти модели расчета главных напряжений легли в основу вычисления безопасных давлений, передаваемых дорожной одеждой на земляное полотно [44–47], а так же в основу усовершенствованного метода расчета касательных напряжений в грунте [48].

Общепринятый метод расчета слоев покрытий и оснований из монолитных материалов по критерию сопротивления усталостному разрушению от растяжения при изгибе предложен группой специалистов, возглавляемой М. Б. Корсунским. Основным недостатком этого метода является применение в основе расчета первой теории прочности, предложенной Г. Галилеем в 17 в. Известно, что вследствие неучета влияния промежуточного и минимального главного напряжения этот критерий приводит к неверным результатам. Поэтому специалисты дорожники предлагают заменить этот критерий другим условием прочности. Так появились критерии прочности и условия пластичности асфальтобетона, учитывающие эффекты накапливания повреждений в этом материале [49–52].

Критерии, учитывающие накапливание повреждений в асфальтобетоне приведены в табл. 2.

Таблица 2. Уравнения предельного состояния критериев прочности и пластичности асфальтобетона, учитывающие эффект накапливания повреждений

Наименование критерия	Уравнение предельного состояния
Критерий О. Мора [51, 52]	$\frac{1}{1 - \omega} \cdot \left(\sigma_1 - \frac{R_p}{R_c} \cdot \sigma_3 \right) = R_p$, где R_p и R_c — пределы прочности на одноосное растяжение и сжатие, Па; ω — поврежденность.
Модифицированный критерий Писаренко–Лебедева [50]	$\frac{1}{1 - \omega} \cdot \left(1 - \frac{R_p}{R_c} \right) \cdot \sigma_1 + \frac{R_p}{R_c} \cdot \sigma_3 = R_p$, где s_n — интенсивность нормальных напряжений, Па.
Модифицированный трехпараметрический критерий Кулона–Мора [49, 50]	$\frac{1}{2 \cdot (1 - \omega)} \cdot \left(\sigma_1 \cdot \left(\frac{1 - \sin\varphi}{1 + \sin\varphi} \right)^d - \left(\frac{1 + \sin\varphi}{1 - \sin\varphi} \right)^d \cdot \sigma_3 \right) = c$

Для расчета мер теории накапливания повреждений в асфальтобетоне предложено два способа [53, 54]. Наиболее просто поврежденность можно найти используя принцип эквивалентности деформаций, согласно которому деформация поврежденной среды находится по одной из двух формул:

$$\varepsilon_D = \frac{\sigma}{(1 - \omega)} \cdot \frac{1}{E}; \quad \varepsilon_D = \frac{\sigma}{E_D} \tag{4}$$

где E и E_D — модули упругости неповрежденного тела и поврежденной среды, МПа.

Положив между формулами (4) равенство, найдем, что:

$$\omega = 1 - \frac{E_D}{E} \quad (5)$$

Модуль упругости поврежденной среды определяют решением интегрального уравнения вида [50, 53, 54]:

$$E_{DN} = E \cdot \left(1 - a \cdot \int_1^N n^b dn \right) \quad (6)$$

Взяв интеграл, приходят к формуле (при условии $b \neq -1$) [50, 53, 54]:

$$E_{DN} = E \cdot \left(1 - a \cdot \frac{N^{b+1} - 1}{b+1} \right) \quad (7)$$

Используя формулу (7) в выражение (5), получим:

$$\omega = a \cdot \frac{N^{b+1} - 1}{b+1} \quad (8)$$

Подводя итог, выполненному обзору, отметим, что в основе рассмотренных нами современных методов расчета дорожных конструкций лежит большой объем знаний, к которому относят:

- механику сплошной среды, а именно теорию упругости [2–9] и пластичности [11–33];
- математическую статистику, применяемую при обработке данных экспериментов [34–36];
- механику зернистой среды [36–42];
- механику поврежденных сред [49–54];
- различные комбинации этих методов [44–46].

Поэтому автор для своей магистрантской квалификационной работы ограничит область исследования теорией пластичности и математической статистикой, которые попытается применить для совершенствования способов расчета дорожных конструкций по колееобразованию.

Литература:

1. ОДН 218.046–01. Проектирование нежестких дорожных одежд. — М.: ГСДХ Минтранса России, 2001. — 146 с.
2. Иванов, Н. Н. Проектирование дорожных одежд / Н. Н. Иванов и др. — М.: Изд-во автотранспортной литературы, 1955. — 250 с.
3. Burmister, D. M. The Theory of Stresses and Displacements in Layered Systems and Applications to the Design of Airport Runways // Proceedings, Highway Research Board. — 1943. — Vol. 23. — P. 126–144.
4. Конструирование и расчет нежестких дорожных одежд / Н. Н. Иванов и др. — М.: Транспорт, 1973. — 328 с.
5. Колмогоров, Г. Л., Кычкин В. И., Есипенко И. А. Метод конечных разностей в исследовании дорожных одежд при воздействии реальной транспортной нагрузки // Строительная механика инженерных конструкций и сооружений. — 2014. — № 1. — с. 69–77.
6. Колмогоров, Г. Л., Кычкин В. И., Есипенко И. А. Динамическая реакция дорожной одежды на действие динамической нагрузки // Строительная механика инженерных конструкций и сооружений. — 2015. — № 5. — с. 39–47.
7. Смирнов, А. В., Андреева Е. В. Динамическое напряженное состояние слоев дорожных одежд конструкций, изгибаемых колебательным волновым полем // Вестник Сибирской государственной автомобильно-дорожной академии. — 2012. — № 5 (27) — с. 68–70.
8. Смирнов, А. В., Андреева Е. В., Герцог В. Н. Воздействие подвижных нагрузок на покрытия и основания автомагистралей // В сборнике: Актуальные проблемы архитектуры и строительства Материалы международной научно-практической конференции. 2014. с. 117–124.
9. Смирнов, А. В., Андреева Е. В., Игнатов В. Ф. Динамические процессы в дорожных конструкциях // Вестник Сибирской государственной автомобильно-дорожной академии. — 2015. — № 5 — с. 81–86.
10. Герцог, В. Н., Долгих Г. В., Кузин В. Н. Расчет дорожных одежд по критериям ровности. Часть 1. Обоснование норм ровности асфальтобетонных покрытий // Инженерно-строительный журнал. — 2015. — № 5 (57) — с. 45–57.
11. Александров, А. С. Применение теории наследственной ползучести к расчету деформаций при воздействии повторных нагрузок: монография. — Омск: СибАДИ, 2014. — 152 с.
12. Александров, А. С. Обобщающая модель пластического деформирования дискретных материалов дорожных конструкций при воздействии циклических нагрузок // Строительные материалы. 2016. № 5. с. 27–30.

13. Александров, А. С., Киселева Н. Ю. Пластическое деформирование гнейс- и диабаз материалов при воздействии повторяющихся нагрузок // Известия высших учебных заведений. Строительство. — 2012. — № 6. — с. 49–59.
14. Семенова, Т. В., Гордеева С. А., Герцог В. Н. Определение пластических деформаций материалов, используемых в дорожных конструкциях // Вестник Томского государственного архитектурно-строительного университета. — 2012. — № 4 (37). — с. 247–254.
15. Семенова, Т. В., Герцог В. Н. Пластическое деформирование материалов с дискретной структурой в условиях трехосного сжатия при воздействии циклических нагрузок // Вестник Сибирской государственной автомобильно-дорожной академии. — 2013. — № 1 (29). — с. 68–73.
16. Александров, А. С. Пластическое деформирование гранодиоритового щебня и песчано-гравийной смеси при воздействии трехосной циклической нагрузки // Инженерно-строительный журнал. — 2013. — № 4 (39) — с. 22–34.
17. Стригун, Т. В., Александрова Н. П. Моделирование пластических деформаций дискретных материалов в слоях дорожных конструкций // В сборнике: Наука XXI века: опыт прошлого — взгляд в будущее: материала II международной научно-практической конференции — Омск, СибАДИ, 2016. — с. 229–233.
18. Кузин, Н. В. Расчет пластических смещений асфальтобетонных порожных покрытий // Молодой ученый. — 2016. — № 10 (114). — с. 253–255.
19. Кузин, Н. В. Исследование пластичности дорожных асфальтобетонов // Молодой ученый. — 2016. — № 10 (114). — с. 255–257.
20. Александров, А. С. Трехпараметрическое условие пластичности Кулона–Мора. Часть 1. Вывод критерия. // В сборнике: Наука XXI века: опыт прошлого — взгляд в будущее: материала II международной научно-практической конференции — Омск, СибАДИ, 2016. — с. 50–54.
21. Александров, А. С. Трехпараметрическое условие пластичности Кулона–Мора. Часть 2. Круги предельных напряжений. // В сборнике: Наука XXI века: опыт прошлого — взгляд в будущее: материала II международной научно-практической конференции — Омск, СибАДИ, 2016. — с. 54–59.
22. Александров, А. С. Трехпараметрическое условие пластичности Кулона–Мора. Часть 3. Определение параметров материала. // В сборнике: Наука XXI века: опыт прошлого — взгляд в будущее: материала II международной научно-практической конференции — Омск, СибАДИ, 2016. — с. 59–64.
23. Калинин, А. Л. Способ модификации условий пластичности // В сборнике: Наука XXI века: опыт прошлого — взгляд в будущее: материала II международной научно-практической конференции — Омск, СибАДИ, 2016. — с. 59–150.
24. Александров, А. С., Калинин А. Л. Совершенствование расчета дорожных конструкций по сопротивлению сдвигу. Часть 1. Учет деформаций в условии пластичности Кулона — Мора // Инженерно-строительный журнал. — 2015. № 7 (59). — с. 4–17.
25. Батырова, В. В. Применение третьей теории прочности и сопротивления недренажному сдвигу для расчета безопасных давлений на земляное полотно // Молодой ученый. — 2016. — № 11.
26. Чусов, В. В. Перспективы применения эмпирических условий пластичности грунтов и определение их параметров при трехосных испытаниях грунтов Вестник ВолГАСУ. — 2015. № 42 (61). — с. 49–57.
27. Александров, А. С., Долгих Г. В. Калинин А. Л. Модификация критериев прочности сплошной среды для расчета грунтов земляного полотна по сопротивлению сдвигу // В сборнике: Архитектура. Строительство. Транспорт. Технологии. Инновации Материалы Международного конгресса ФГБОУ ВПО «СибАДИ». — Омск: СибАДИ, 2013. — с. 228–235.
28. Александров, А. С., Долгих Г. В., Калинин А. Л. Применение критерия Друкера — Прагера для модификации условий пластичности // Наука и техника в дорожной отрасли. — 2013. № 2. — с. 26–29.
29. Калинин, А. Л. Совершенствование расчета касательных напряжений в дорожных конструкциях. Часть 1. Модификация критерия Писаренко-Лебедева и его применение при расчете касательных напряжений // Молодой ученый. — 2016. — № 6 (110). — с. 108–114.
30. Калинин, А. Л. Применение модифицированных условий пластичности для расчета безопасных давлений на грунты земляного полотна. // Инженерно-строительный журнал — 2013. № 4 (39). — с. 35–45.
31. Долгих, Г. В. Применение безопасного давления в качестве критерия расчета земляного полотна по сдвигу в грунте // В сборнике: Наука XXI века: опыт прошлого — взгляд в будущее: материала II международной научно-практической конференции — Омск, СибАДИ, 2016. — с. 113–117.
32. Александров, А. С. Совершенствование расчета дорожных конструкций по сопротивлению сдвигу. Том Часть 1. Состояние вопроса. — Омск: СибАДИ, 2015. — 292 с.
33. Александров, А. С. Совершенствование расчета дорожных конструкций по сопротивлению сдвигу. Том Часть 2. Предложения. — Омск: СибАДИ, 2015. — 262 с.

34. Казарновский, В. Д., Смирнов В. М., Косарев Ю. И. Определение расчетных значений прочностных характеристик песков с учетом воздействия повторных нагрузок // Исследования по механике дорожных одежд. — М.: СоюздорНИИ, 1985. — с. 80–92.
35. Петрушин, Е. А. Сдвигоустойчивость глинистых грунтов в условиях кратковременных многократных нагрузок // Совершенствование методов расчета и конструирования дорожных одежд. — М.: СоюздорНИИ. — 1986. — с. 88–96.
36. Смирнов, В. М., Дорогутина С. Н. Закономерности деформирования связного грунта при воздействии транспортных нагрузок // Новое в проектировании конструкций дорожных одежд. — М.: СоюздорНИИ, 1988. — с. 65–79.
37. Foster, C. R., Ahlvin R. G. Stresses and deflections induced by a uniform circular load. // Proc. Highway Research Board. — 1954. — Vol. 33. — P. 236–246.
38. Александров, А. С., Александрова Н. П., Долгих Г. В. Модифицированные модели для расчета главных напряжений в дорожных конструкциях из дискретных материалов // Строительные материалы. — 2012. — № 10. — с. 14–17.
39. Александрова, Н. П. Модифицированные модели для расчета главных напряжений в грунте земляного полотна // В сборнике: Архитектура. Строительство. Транспорт. Технологии. Инновации Материалы Международного конгресса ФГБОУ ВПО «СибАДИ». Омск, 2013. — с. 236–246.
40. Александрова, Н. П., Семенова Т. В., Долгих Г. В. Совершенствование моделей расчета главных напряжений и девиатора в грунте земляного полотна // Вестник СИБАДИ. — 2014. — № 2 (36). с. 49–54.
41. Александров, А. С. Один из путей расчета минимальных главных напряжений в грунтах земляного полотна / А. С. Александров // В сборнике: Архитектура. Строительство. Транспорт. Технологии. Инновации Материалы Международного конгресса ФГБОУ ВПО «СибАДИ». — Омск, СибАДИ, 2013. — с. 217–228.
42. Александров, А. С., Долгих Г. В. Способ определения минимального главного напряжения. Часть 1. В грунтах земляного полотна. // В сборнике: Наука XXI века: опыт прошлого — взгляд в будущее: материала II международной научно-практической конференции — Омск, СибАДИ, 2016 — с. 64–68.
43. Александров, А. С., Долгих Г. В. Способ определения минимального главного напряжения. Часть 2. В слоях дорожной одежды из дискретных материалов // В сборнике: Наука XXI века: опыт прошлого — взгляд в будущее: материала II международной научно-практической конференции — Омск, СибАДИ, 2016. — с. 69–73.
44. Долгих, Г. В. Расчет грунтов земляного полотна по критерию безопасных давлений // Вестник Сибирской государственной автомобильно-дорожной академии. — 2013. — № 6 (34). — с. 43–49.
45. Долгих, Г. В. Расчет нежестких дорожных одежд по критерию безопасных давлений на глинистые грунты земляного полотна // Автореф. Дис. канд. техн. наук. — Омск: СибАДИ. — 2014. — 20 с.
46. Долгих, Г. В. Применение критерия безопасных давлений для расчета дорожных конструкций по сопротивлению сдвигу в грунте земляного полотна // // В сборнике: Политранспортные системы материалы VIII Международной научно-технической конференции в рамках года науки Россия — ЕС. Новосибирск: СГУПС, 2015. — с. 176–182.
47. Александров, А. С., Долгих Г. В., Калинин А. Л. О допускаемых давлениях на грунты земляного полотна и слои дорожной одежды // Наука и техника в дорожной отрасли. — 2012. № 2. — с. 10–13.
48. Александров, А. С., Долгих Г. В. Калинин А. Л. Один из путей совершенствования расчета дорожных одежд по условию сопротивления сдвигу в грунте земляного полотна // Модернизация и научные исследования в транспортном комплексе. — Пермь: Пермский национальный исследовательский политехнический университет, 2013. — с. 9–22.
49. Чусов, В. В. Модифицированные критерии Писаренко-Лебедева и Кулона-Мора, учитывающие меры теории накопления повреждений // Молодой ученый. — 2016. — № 9 (113). — с. 338–341.
50. Чусов, В. В. Применение теории накопления повреждений в условиях пластичности асфальтобетона для расчета дорожных покрытий по сопротивлению сдвигу // Молодой ученый. — 2016. — № 6 (110). — с. 221–227.
51. Александрова, Н. П., Александров А. С., Чусов В. В. Учет поврежденности структуры асфальтобетона в критериях прочности и условиях пластичности // В сборнике: Политранспортные системы материалы VIII Международной научно-технической конференции в рамках года науки Россия — ЕС. Новосибирск: СГУПС, 2015. — с. 219–225.
52. Александрова, Н. П., Александров А. С., Чусов В. В. Модификация критериев прочности и условий пластичности при расчетах дорожных одежд // Вестник Сибирской государственной автомобильно-дорожной академии. — 2015. № 1 (41). — с. 47–54.
53. Aleksandrova, N. P. Chysow V. V. The usage of integral equations hereditary theories for calculating changes measures in the theory of damage when exposed to repeated loads // Magazine of Civil Engineering, 2016, No.2. Article in Press.

54. Чусов, В. В., Александрова Н. П. Два способа расчета мер теории накопления // В сборнике: Наука XXI века: опыт прошлого — взгляд в будущее: материала II международной научно-практической конференции — Омск, СибАДИ, 2016 — с. 271–275.

Изотерма и дифференциальные теплоты адсорбции бензола на цеолитах LiZSM-5, CsZSM-5 и NaZSM-5

Ганиева Зулайхо Гиясовна, ассистент
Бухарской филиал Ташкентского института ирригации и мелиорации (Узбекистан)

Одним из высокоэффективных катализаторов различных процессов в нефтехимии и нефтеперерабатывающей промышленности являются катализаторы типа ZSM-5. Отчасти причиной широкой известности явилась уникальная способность этих цеолитов катализировать процесс получения высокооктанового бензина из нефтяного сырья, как например метанол [1].

Изотерма адсорбции бензола на NaZSM-5 в полулогарифмических координатах представлена на рис.1. Равновесные давления при малых заполнениях доходят до $P/P_s = 5 \times 10^{-8}$, что свидетельствует о прочной адсорбции бензола на NaZSM-5. Изотерма адсорбции доведена до 1,14 ммоль/г $P/P_s = 0,75$.

Из рис.1 видно, что изотерма адсорбции бензола на NaZSM-5 S-образная с почти вертикальным участком посередине. При малых заполнениях изотерма вогнута, начиная с $P/P_s = 6 \times 10^{-7}$ стремительно поднимается вверх, при средних заполнениях испытывает слабый перегиб, и уже при $P/P_s = 0,005$ достигает степени заполнения $q \sim 0,88$ (1 ммоль/г). Таким образом, изотерма почти полностью укладывается в узком интервале P/P_s от 6×10^{-7} до 0,005. Такая форма изотермы свойственна фазовым превращениям адсорбированного вещества и является

одним из основных феноменологических признаков гомогенности сорбционной системы в целом.

На кривой дифференциальных теплота при 1 ммоль/г теплота адсорбции начинает резко уменьшаться от 85 кДж/моль до уровня конденсации 33,5 кДж/моль при 1,3 ммоль/г (рис.2).

Изотерма адсорбции бензола (рис.1) на NaZSM-5 удовлетворительно описывается трехчленным уравнением ТОЗМ.

$$a = 0,495 \exp \left[- \left(\frac{A}{33,3} \right)^6 \right] + 0,47 \exp \left[- \left(\frac{A}{33,5} \right)^8 \right] + 0,175 \exp \left[- \left(\frac{A}{10,0} \right)^2 \right]$$

где a — величины адсорбции в ммоль/г, $A = RT \ln P_s / P$ — работа адсорбции в кДж/моль.

Первые два члена описывают основную область адсорбции до 1,03 ммоль/г с очень высокими значениями показателя степени под экспонентой (это параметр n уравнений ТОЗМ), свидетельствующего о высокой однородности сорбционной системы. Если пренебречь существованием дополнительного перегиба изотермы в области $a = 0,55$ ммоль/г, т.е. пренебречь возможностью протекания двух механизмов адсорбции в двух типах сорбционных пространств, хотя и близких по характеру и строению, то изотерму можно описать двухчленным уравне-

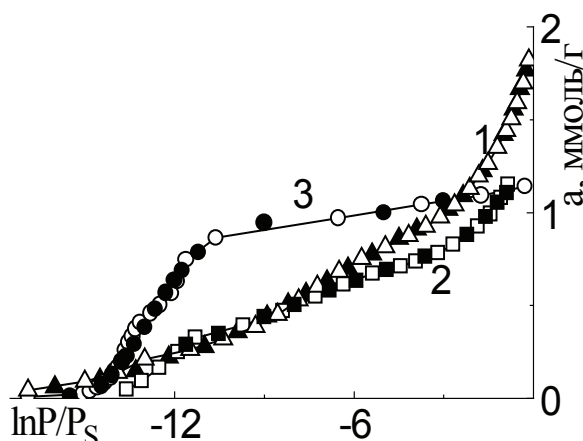


Рис. 1. Изотермы адсорбции бензола на цеолитах: 1 — LiZSM-5, 2 — CsZSM-5, 3 — NaZSM-5. Δ, \square, \circ — экспериментальные точки; s, n, l — точки, рассчитанные с помощью ТОЗМ

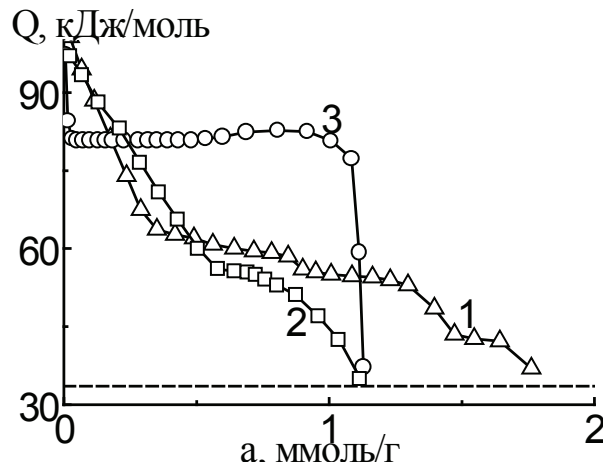


Рис. 2. Дифференциальные теплоты адсорбции бензола на цеолитах: 1 — LiZSM-5, 2 — CsZSM-5, 3 — NaZSM-5

нием ТОЗМ, причем первый член уравнения охватывает первый и второй член предыдущего.

$$a = 0,965 \exp[-(A/33)^{10}] + 0,175 \exp[-(A/10)^2]$$

где a — величины адсорбции в ммоль/г, $A = RT \ln P_s/P$ — работа адсорбции в кДж/моль.

Полученные два уравнения изотермы по ТОЗМ предполагают два механизма адсорбции. В первом случае (трехчленное уравнение) происходит объемное заполнение примерно половины сорбционного пространства цеолита по одному механизму, затем заполнение другой части сорбционного пространства, отличающегося от первого по конформации адсорбционного комплекса. Поскольку характеристические энергии адсорбции этих изотерм очень близки, то изотермы существенно перекрываются.

Во втором случае (двухчленное уравнение) почти весь сорбционный объем заполняется по единому механизму. Возможно здесь и нет альтернативы, т.е. сорбционный процесс протекает равномерно по всему сорбционному пространству, но достигнув примерно середины заполнения и вследствие насыщения определенных центров адсорбции при таком заполнении дальнейшая адсорбция происходит за счет присоединения молекулы адсорбата к первоначальному адсорбционному комплексу.

Последний вариант механизма адсорбции представится более правдоподобным при детальном рассмотрении молекулярного механизма адсорбции [2].

Пределный сорбционный объем цеолита по бензолу, как видно из рис.1 и 2, составляет 1,15 ммоль/г. Содержание катионов натрия, согласно химическому составу ЭЯ, составляет 0,5 ммоль/г, т.е. каждый катион натрия взаимодействует с двумя молекулами бензола. По-видимому, образуется сэндвичеобразный р-комплекс с катионом натрия посередине.

Вопрос только в том — где расположен этот комплекс — либо полностью в местах пересечения каналов (очевидное для тетраэдрических комплексов, например, со спиртами, либо в прямых каналах с сильно выступающими в пересечения каналов молекулами бензола (менее вероятная модель). По-видимому, бензольные кольца об-

разуют двугранный угол, равный углу между зигзагообразными каналами, и плоскостями колец, ориентированными перпендикулярно осям зигзагообразных каналов.

Изотерма адсорбции C_6H_6 на LiZSM-5 резко отличается от изотермы адсорбции бензола на NaZSM-5, что указывает на различный характер адсорбции бензола в этих цеолитах (рис.1). Равновесные давления при малых заполнениях тоже доходят до $P/P_s = 5 \times 10^{-8}$ как в NaZSM-5, что свидетельствует о прочной адсорбции бензола на цеолите LiZSM-5. Изотерма адсорбции доведена до 1,82 ммоль/г при $x P/P_s = 0,85$.

Изотерма адсорбции C_6H_6 на LiZSM-5 полностью описывается 3-х членным уравнением ТОЗМ.

$$a = 1,078 \exp[-(A/23,75)^2] + 0,74 \exp[-(A/4,38)^2] + 0,275 \exp[-(A/1,47)^2]$$

где a — величины адсорбции в ммоль/г, $A = RT \ln P_s/P$ — работа адсорбции в кДж/моль.

Дифференциальные теплоты адсорбции C_6H_6 на LiZSM-5 также резко отличаются от теплот адсорбции C_6H_6 на NaZSM-5 (рис.2). Постоянство теплоты и резкий подъем изотермы в широком интервале заполнений в случае NaZSM-5 указывает на образование «сэндвичевой» структуры. Содержание катионов лития, согласно химическому составу ЭЯ, составляет 0,577 ммоль/г, т.е. линейное падение теплоты от 100 кДж/моль до 70 кДж/моль при адсорбции 0,29 ммоль/г, максимум на термокинетической кривой при адсорбции 0,29 ммоль/г и перегиб изотермы при адсорбции 0,29 ммоль/г, по-видимому, связаны с адсорбцией одной молекулы C_6H_6 на двух катионах Li^+ . Благодаря своему размеру этот комплекс, т.е. одна молекула C_6H_6 с двумя Li^+ располагается в перекрестьях прямых и зигзагообразных каналов. Последующие молекулы адсорбируются уже без участия катионов и уровень теплоты у них, как в случае силикалита, т.е. бескатионной структуре (рис.2).

Дифференциальные теплоты адсорбции C_6H_6 на CsZSM-5 резко отличаются от теплот адсорбции C_6H_6 в цеолитах NaZSM-5 и LiZSM-5, что указывает на различный характер адсорбции C_6H_6 в этих цеолитах (рис.2). Начальная теплота адсорбции равна 97 кДж/моль. Теплота

линейно снижается до 60 кДж/моль при 0,54 ммоль/г. Содержание катионов цезия составляет 0,54 ммоль/г, т.е. каждый катион цезия взаимодействует с одной молекулой бензола. Из рис.2 видно, что дальнейший уровень теплоты адсорбции молекулы бензола как в случае силикалита, т.е. молекулы адсорбируются в силикалитной части цеолита. В начале теплота почти не меняется до адсорбции 0,74 ммоль/г, затем снижается до теплоты конденсации жидкого бензола при адсорбции 1,15 ммоль/г.

Изотерма адсорбции C_6H_6 на CsZSM-5 в полулогарифмических координатах представлена на рис.1. Равновесные давления при малых заполнениях доходят до $P/P_s=10^{-6}$, что свидетельствует о более прочной адсорбции C_6H_6 на CsZSM-5, чем на силикалите.

Изотерма адсорбции доведена до 1,15 ммоль/г при $P/P_s=0,43$. Из рис.1 видно, что изотерма адсорбции C_6H_6 на CsZSM-5 линейно поднимается вверх и при 0,4 ммоль/г образует ступеньку. Далее, из изотермы видно, что равновесное давление постепенно поднимается. Изотерма адсорбции C_6H_6 на CsZSM-5 описывается от 0,3 ммоль/г до конца двухчленным уравнением ТОЗМ.

$$a=0,837 \exp[-A/28,3^2] + 0,36 \exp[-(A/4,39)^2]$$

где a — величины адсорбции в ммоль/г, $A=RT \ln P_s/P$ — работа адсорбции в кДж/моль.

Из рис.1 видно, что расчетные данные хорошо согласуются с экспериментальными.

Литература:

1. Глонти, Г. О., Клячко Ф. Л. Адсорбционный критерий структурной характеристики цеолитов //Изв.АН СССР, Сер.хим. — 1984. — № 5. — С.992—995.
2. Lohze, U., Thamm H., Noack M., Fahlke B. Adsorption of hydrocarbons and on ZSM-5 and on ZSM-5 converted by dealumination into silicalite //J.Incl.Phenom. — 1987. — v.5. — № 3. — P.307—313.

Модифицированный динамический инженерный метод расчета главных напряжений в дискретных материалах дорожных конструкций

Грязнова Марина Константиновна, студент
Сибирская государственная автомобильно-дорожная академия

В статье кратко описаны методики расчета главных напряжений в дорожных конструкциях. Объединены два метода «СибАДИ», что позволило рассчитывать минимальные главные напряжения в дискретных материалах через величину максимального главного напряжения от динамической нагрузки, изменяющейся во времени по синусоиде.

Ключевые слова: дорожная конструкция, главные напряжения динамическая нагрузка

В настоящее время разработан ряд методов, позволяющих рассчитывать характеристики напряженно-деформированного состояния (НДС) в дорожных конструкциях от динамических нагрузок, распределенных по круглой площадке. Эти методы условно разделим на точные [1—3], в основе которых использован смешанный метод теории упругости, и инженерные [4—6], позволяющие рассчитывать только одну вертикальную компоненту в тензорах напряжений и деформаций. Достоинство точных методов расчета, берущих свое начало в работах Д. Бурмистра [7, 8], К. Юшиты [9], Ю. Хуанга [10] и др. выдающихся ученых известных во всем мире состоит в том, что эти решения позволяют вычислять все компоненты тензора напряжений и деформаций. Такие решения удовлетворяют всем канонам механики сплошной среды, в частности линейной теории упругости, но конечные математические выражения громоздки и без вычислительного программного обеспечения сложно реализуются на практике. Инженерные методы [4—6] и подобные удовлетворяют только часть допущений и постулатов теории упругости и позволяют вычислять только вертикальное нормальное напряжение и вертикальную относительную деформацию, зато формулы для определения этих компонент тензоров просты и легко применяются специалистами при исчислении. В настоящей публикации автор предпримет попытку модификации инженерного метода расчета напряжений, предложенного в работе [6]

Согласно решению [4—6] вертикальное нормальное напряжение в слое можно определить по формуле, которую запишем в общем виде:

$$\sigma_z = q_0 \cdot \left[1 + \sqrt{\frac{E_1}{E_2} \cdot \left(\frac{z}{D_0} \right)^2} \right]^{-1} \tag{1}$$

Где q_0 — нагрузка, распределенная по круглой площадке, на поверхности рассчитываемого слоя, Па; E_1 и E_2 — модули упругости материалов верхнего (рассчитываемого) и нижнего слоев, в случае если подстилающий слой является слоистым полупространством, то E_2 может быть найден, как общий модуль упругости на поверхности подстилающего полупространства, Па; z — глубина расположения точки в рассчитываемом слое, определяемая расстоянием от поверхности слоя до этой точки, м; D_0 — диаметр круглой площадки, по которой распределена нагрузка на поверхности рассчитываемого слоя, м.

В сечении расположенном по оси симметрии нагрузки, распределенной по круглой площадке, направления главных и координатных осей совпадают, вследствие чего в этом сечении имеют место равенства $s_1=s_2, s_1=s_x, s_1=s_y$ [11]. Поэтому в сечении по оси симметрии нагрузки формулу (1) можно применить для расчета как вертикального нормального, так и максимального главного напряжения, но в слоях из дискретных материалов и грунтах земляного полотна. Для расчета минимальных главных напряжений применим способ, опубликованный в работах [12]. Этот способ применим для определения напряжений в полупространстве [13], а так же слое конечной толщины [14], что позволило модифицировать ряд моделей механики зернистой среды и инженерных способов расчета [15–17].

Следуя этим работам минимальное главное напряжение в полупространстве определяют по формулам:

$$\sigma_2 = \sigma_3 = \alpha \cdot \frac{\mu}{1-\mu} \cdot \sigma_1 = \alpha \cdot \xi \cdot \sigma_1; \quad \alpha = 1 - \sqrt{1-K^2} \tag{2}$$

где a — функция глубины; m — коэффициент Пуассона; χ — коэффициент бокового давления; K — коэффициент уменьшения максимального главного напряжения по глубине в сечении по оси симметрии нагрузки.

Из анализа зависимости (1) следует, что коэффициент K определяется вторым множителем, заключенным в квадратные скобки, то есть

$$K = \left[1 + \sqrt{\frac{E_1}{E_2} \cdot \left(\frac{z}{D_0}\right)^2} \right]^{-1} \tag{3}$$

Зависимость (3) применима к полупространству. В соответствии с данными работы [14] для расчета K в слое конечной толщины нужно применить метод Н. Одемарка:

В методе Н. Одемарка [18] имеется возможность рассчитать приведённую по жесткости ординату точки в полупространстве z к ординате точки в слое конечной толщины z_{cl} . Сделать это можно по формуле:

$$z = z_{cl} \cdot \sqrt[3]{\frac{E_1}{E_2}} \tag{4}$$

где z_{cl} — ордината точки, расположенной на оси симметрии нагрузки слоя толщиной h , то есть $0 \leq z_{cl} \leq h$, м; E_{ocn} — модуль упругости материала слоя, МПа; E_{ocn} — общий модуль упругости на поверхности однородного или слоистого полупространства, подстилающего рассчитываемый слой, МПа.

Тогда с учетом (4) зависимость (3) примет вид:

$$K = \left[1 + \sqrt{\left(\frac{z_{cl}}{D_0}\right)^2 \cdot \left(\frac{E_1}{E_2}\right)^{\frac{5}{3}}} \right]^{-1} \tag{5}$$

Подставляя выражение (5) во вторую формулу зависимостей (2), получим:

$$\alpha = 1 - \sqrt{1 - \left[1 + \sqrt{\left(\frac{z_{cl}}{D_0}\right)^2 \cdot \left(\frac{E_1}{E_2}\right)^{\frac{5}{3}}} \right]^{-2}} \tag{6}$$

Подставив зависимости (1) и (6) в первое выражение формул (2), получим:

$$\sigma_2 = \sigma_3 = \frac{\mu}{1-\mu} \cdot q_0 \cdot \left[1 + \sqrt{\frac{E_1}{E_2} \cdot \left(\frac{z}{D_0}\right)^2} \right]^{-1} \cdot \left\{ 1 - \sqrt{1 - \left[1 + \sqrt{\left(\frac{z_{cl}}{D_0}\right)^2 \cdot \left(\frac{E_1}{E_2}\right)^{\frac{5}{3}}} \right]^{-2}} \right\} \tag{7}$$

Предложенная автором модель, включающая в себя известную и новую формулы (1) и (7), является новой модификацией, полученной применением способа [12–14]. Этой модели в работах [15–17] нет, поэтому она дополняет пере-

чень модифицированных моделей для расчета главных напряжений в дискретных средах в сечении по оси симметрии нагрузки. Область ее применения достаточно широка, можно использовать при вычислении допускаемых давлений на земляное полотно и слои дорожной одежды [19–23], или при расчете необратимых деформаций в материалах с дискретной структурой [24–26]. При допущении, что при достаточно высоких температурах асфальтобетона в сравнительно толстых слоях, возникает трехосное сжатие, то эта модель применима для: расчета деформаций [27, 28], эквивалентных напряжений в критериях прочности [29–31], мер теории накопления повреждений [31–34].

Литература:

1. Колмогоров, Г.Л., Кычкин В.И., Есипенко И.А. Метод конечных разностей в исследовании дорожных одежд при воздействии реальной транспортной нагрузки // Строительная механика инженерных конструкций и сооружений. — 2014. — № 1. — с. 69–77.
2. Колмогоров, Г.Л., Кычкин В.И., Есипенко И.А. Динамическая реакция дорожной одежды на действие динамической нагрузки // Строительная механика инженерных конструкций и сооружений. — 2015. — № 5. — с. 39–47.
3. Есипенко, И.А., Колмогоров Г.Л., Кычкин В.И. Численное моделирование колебаний дорожной одежды с учетом изменения приведенных физико-механических свойств материала // Транспорт. Транспортные сооружения. Экология. — 2014. — № 3. — с. 57–73.
4. Смирнов, А.В., Андреева Е.В. Динамическое напряженное состояние слоев дорожных одежд конструкций, изгибаемых колебательным волновым полем // Вестник Сибирской государственной автомобильно-дорожной академии. — 2012. — № 5 (27) — с. 68–70.
5. Смирнов, А.В., Андреева Е.В., Герцог В.Н. Воздействие подвижных нагрузок на покрытия и основания автомагистралей // В сборнике: Актуальные проблемы архитектуры и строительства Материалы международной научно-практической конференции. 2014. с. 117–124.
6. Смирнов, А.В., Андреева Е.В., Игнатов В.Ф. Динамические процессы в дорожных конструкциях // Вестник Сибирской государственной автомобильно-дорожной академии. — 2015. — № 5 — с. 81–86.
7. Burmister, D. M. The Theory of Stresses and Displacements in Layered Systems and Applications to the Design of Airport Runways // Proceedings, Highway Research Board. — 1943. — Vol. 23. — P. 126–144.
8. Burmister, D. M. Stress and Displacement Characteristics of a Two-Layered Rigid Base Soil System: Influence Diagrams and Practical Applications. // Proceedings Highway Research Board. — 1956. — Vol. 35. — P. 773–814.
9. Ueshita, K., Meyerhof, G. G. Surface Displacement of an Elastic Layer under Uniformly Distributed Loads // Highway Research Board Record. — 1968. — № . 228. — P. 1–10.
10. Huang, Y. H. Computation of Equivalent Single-wheel Loads Using Layered Theory // Proceedings, Highway Research Board. 1969.
11. Александров, А.С. Применение теории наследственной ползучести к расчету деформаций при воздействии повторных нагрузок: монография. — Омск: СибАДИ, 2014. — 152 с.
12. Александров, А.С., Долгих Г.В., Юрьев Д.В. Расчет главных напряжений в слоях дорожной одежды из дискретных материалов // Транспортное строительство. — 2011. — № 7. — с. 17–22.
13. Александров, А.С., Долгих Г.В. Способ определения минимального главного напряжения. Часть 1. В грунтах земляного полотна. // В сборнике: Наука XXI века: опыт прошлого — взгляд в будущее: материалы II международной научно-практической конференции — Омск, СибАДИ, 2016 — с. 64–68.
14. Александров, А.С., Долгих Г.В. Способ определения минимального главного напряжения. Часть 2. В слоях дорожной одежды из дискретных материалов // В сборнике: Наука XXI века: опыт прошлого — взгляд в будущее: материалы II международной научно-практической конференции — Омск, СибАДИ, 2016. — с. 69–73.
15. Александрова, Н.П. Модифицированные модели для расчета главных напряжений в грунте земляного полотна // В сборнике: Архитектура. Строительство. Транспорт. Технологии. Инновации Материалы Международного конгресса ФГБОУ ВПО «СибАДИ». Омск, 2013. — с. 236–246.
16. Александрова, Н.П., Семенова Т.В., Долгих Г.В. Совершенствование моделей расчета главных напряжений и девиатора в грунте земляного полотна // Вестник СИБАДИ. — 2014. — № 2 (36). с. 49–54.
17. Александров, А.С., Александрова Н.П., Долгих Г.В. Модифицированные модели для расчета главных напряжений в дорожных конструкциях из дискретных материалов // Строительные материалы. — 2012. — № 10. — с. 14–17.
18. Odemark, N. Investigations as to the Elastic Properties of Soils and Design of Pavements according to the Theory of Elasticity. Ph.D. thesis. Statens Väginstytut, Mitteilug No. 77. Stockholm, Sweden, 1949.
19. Александров, А.С., Долгих Г.В., Калинин А.Л. О допускаемых давлениях на грунты земляного полотна и слои дорожной одежды // Наука и техника в дорожной отрасли. — 2012. № 2. — с. 10–13.

20. Долгих, Г. В. Применение безопасного давления в качестве критерия расчета земляного полотна по сдвигу в грунте // В сборнике: Наука XXI века: опыт прошлого — взгляд в будущее: материалы II международной научно-практической конференции — Омск, СибАДИ, 2016. — с. 113–117.
21. Долгих, Г. В. Применение критерия безопасных давлений для расчета дорожных конструкций по сопротивлению сдвигу в грунте земляного полотна // В сборнике: Политранспортные системы материалы VIII Международной научно-технической конференции в рамках года науки Россия — ЕС. Новосибирск: СГУПС, 2015. — с. 176–182.
22. Долгих, Г. В. Расчет нежестких дорожных одежд по критерию безопасных давлений на глинистые грунты земляного полотна // Автореф. Дис. канд. техн. наук. — Омск: СибАДИ. — 2014. — 20 с.
23. Долгих, Г. В. Расчет грунтов земляного полотна по критерию безопасных давлений // Вестник Сибирской государственной автомобильно-дорожной академии. — 2013. — № 6 (34). — с. 43–49.
24. Александров, А. С., Киселева Н. Ю. Пластическое деформирование гнейс- и диабаз материалов при воздействии повторяющихся нагрузок // Известия высших учебных заведений. Строительство. — 2012. — № 6. — с. 49–59.
25. Александров, А. С. Пластическое деформирование гранодиоритового щебня и песчано-гравийной смеси при воздействии трехосной циклической нагрузки // Инженерно-строительный журнал. — 2013. — № 4 (39) — с. 22–34.
26. Александров, А. С. Обобщающая модель пластического деформирования дискретных материалов дорожных конструкций при воздействии циклических нагрузок // Строительные материалы. 2016. № 5. с. 27–30.
27. Кузин, Н. В. Расчет пластических смещений асфальтобетонных дорожных покрытий // Молодой ученый. — 2016. — № 10 (114). — с. 253–255.
28. Кузин, Н. В. Исследование пластичности дорожных асфальтобетонов // Молодой ученый. — 2016. — № 10 (114). — с. 255–257.
29. Александрова, Н. П., Александров А. С., Чусов В. В. Учет поврежденности структуры асфальтобетона в критериях прочности и условиях пластичности // В сборнике: Политранспортные системы материалы VIII Международной научно-технической конференции в рамках года науки Россия — ЕС. Новосибирск: СГУПС, 2015. — с. 219–225.
30. Александрова, Н. П., Александров А. С., Чусов В. В. Модификация критериев прочности и условий пластичности при расчетах дорожных одежд // Вестник Сибирской государственной автомобильно-дорожной академии. — 2015. № 1 (41). — с. 47–54.
31. Чусов, В. В. Модифицированные критерии Писаренко-Лебедева и Кулона-Мора, учитывающие меры теории накопления повреждений // Молодой ученый. — 2016. — № 9 (113). — с. 338–341.
32. Aleksandrova, N. P. Chysow V. V. The usage of integral equations hereditary theories for calculating changes measures in the theory of damage when exposed to repeated loads // Magazine of Civil Engineering, 2016, No.2. Article in Press.
33. Чусов, В. В. Применение теории накопления повреждений в условиях пластичности асфальтобетона для расчета дорожных покрытий по сопротивлению сдвигу // Молодой ученый. — 2016. — № 6 (110). — с. 221–227.

Обзорное описание метода Multi-Fragment-ReplicationJoin (MFRJ) доступа к многомерному хранилищу данных по технологии MapReduce

Джанаев Сослан Игоревич, студент

Московский государственный технический университет имени Н. Э. Баумана

MapReduce

Технология MapReduce, поддерживается в среде баз данных NoSQL. Но существуют специальные каркасы MapReduce, имеющие файловые системы, оптимизированные под эту технологию. Примером такого каркаса является Hadoop файловая система HDFS.

Общий процесс обработки записей <ключ, значение> по технологии MapReduce выглядит так:

map: <K1, V1> → list <K2, V2>,

reduce: <K2, list V2> → list <K3, V3>.

В одном j-ом узле может выполняться несколько экземпляров функции «map». Она должна осуществлять следующие действия: разобрать входную запись и вернуть одну или несколько новых записей {<m_{ji}, r_{ji}>}_i. При этом значение ключа m_{ji} может повторяться. Записи хешируются по значению ключа m_{ji}, формируются разделы записей, номер раздела совпадает со значением хеш-функции. Число разделов обозначим через s. Система объединяет выходные записи программы «map» в s-массивах

и сохраняет их на локальном диске узла (один массив на одно значение хеш-ключа). Таким образом, формируются $n \times s$ файлов, n — число узлов.

После завершения фазы Map каждый результирующий массив F_{ji} записей ($j = 1 \dots n, i = 1 \dots s$) пересылается с j -го узла на узел, где выполняется i -й экземпляр функции «reduce». Важно подчеркнуть, что этот механизм пересылки массивов позволяет обрабатывать записи с одинаковыми значениями ключа в одном узле. Поступившие в узел записи массивов сортируются, объединяются и передаются на вход соответствующей функции «reduce». Программа «reduce» должна обработать записи объединенного массива и выполнить их «свертку» в соответствии с поставленной задачей. Функция «reduce» возвращает выходной массив (как правило, меньшей размерности), и система сохраняет его в виде записей <ключ, значение> в файловой системе MapReduce. Эти записи могут быть использованы в качестве входных данных для других заданий MapReduce.

Ниже приведен сложный пример: соединение двух таблиц, хранящихся в файлах File1 и File2, по значению ключа (Рисунок 3).

На одном узле функция Map читает записи файла File1, а на втором узле два экземпляра функции Map читают за-

писи файла File2 (записи <K1, V1>). Записи этих файлов имеют одинаковую структуру: <K_j, V_m, j>. Т. е. ключ и значение записи являются составными. Здесь j (0 или 1) определяет принадлежность записи файлу: 0 — запись прочитана из файла File1, 1 — из файла File2. Результаты чтения показаны на рисунке справа. Функции Map просто, без обработки, помещают эти записи в выходной поток (list<K2, V2>).

Далее система автоматически выполняет хеширование записей по значению K₁ первого поля ключа (создаются разделы) и реализует фазу «перетасовки» (shuffle). Записи с одинаковыми значениями K₁ попадают в один и тот же узел, так как разделы с одинаковыми значениями хеш-функции передаются в один узел. Система выполняет сортировку записей по составному ключу (см. Sorting) на каждом принимающем узле. Затем записи группируются по первому полю ключа (т.е. по K₁). Поля значений записей, попавших в группу, помещаются в поле значения новой записи (<K2, listV2>, см. Grouping). Порядок составных значений в списке соответствует порядку отсортированных записей. То есть сначала записываются значения файла File 1 (признак 0), а затем — значения файла File2 (признак 1). Эти значения соответствуют одному значению K₁ первого поля ключа.

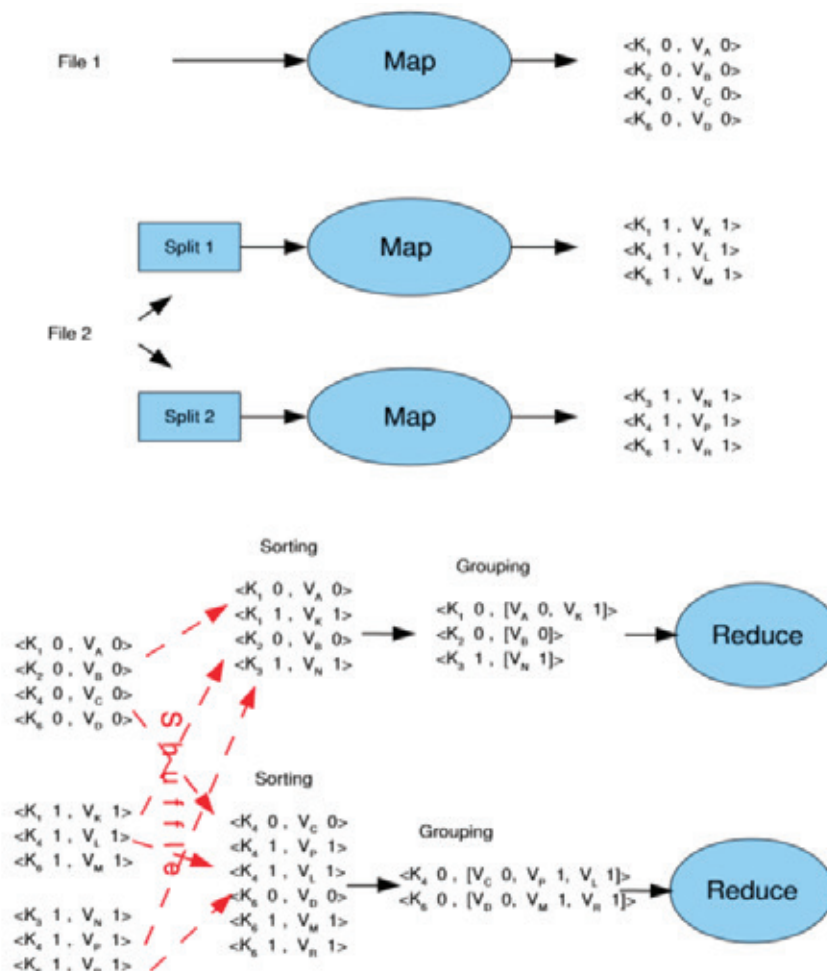


Рис. 1. Пример соединения таблиц по технологии MapReduce

И только после этого сформированная запись передается на вход функции Reduce. Эта функция анализирует запись и выполняет операцию декартова произведения значений файлов File 1 и File2 (ведь эти значения имеют один и тот же ключ соединения K_i). Ниже показан результат соединения (list<K3, V3>):

первыйReduce: $\langle K_1, V_A V_K \rangle$
 второйReduce: $\langle K_4, V_C V_P \rangle, \langle K_4, V_C V_L \rangle$
 $\langle K_6, V_D V_M \rangle, \langle K_6, V_D V_R \rangle$

Эти записи, в свою очередь, также могут быть сгруппированы для них можно применить операции агрегирования (конечно, если V_m — это числовые значения).

Можно выделить следующие преимущества MapReduceHadoop:

1. Простота установки MapReduce (MR), ее относительно низкая стоимость.
2. В функции «map» запроса к MR достаточно просто реализовать разбор (парсер) документа, а в функции «reduce» — объединение результатов разбора.
3. Технология MapReduce обладает возможностью восстанавливать процесс обработки записей после сбоев в середине выполнения запроса с применением метода, не свойственного параллельным СУБД.

Метод доступа MFRJ

Таблицы измерений продублированы по всем узлам системы MapReduce (MR). Они хранятся по строкам. Таблица фактов делится на несколько групп: в 1-ю группу входят столбцы (колонки) внешних ключей измерений (fk_i), столбец каждого факта (m_i) образует отдельную группу (группы 2, ..., k+1). Все блоки таблиц измерений и групп таблицы фактов распределены MapReduce по узлам (автоматически) произвольным образом (MR пытается это делать равномерно).

Доступ к данным выполняется следующим образом.

Map (в каждом узле):

1. Для измерений, которые участвуют в запросе, строятся хеш-индексы в оперативной памяти.
2. Читаются записи внешних ключей измерений таблицы фактов (группа 1), хранящиеся в узле (fk_i), для каждой строки (позиции) и каждого внешнего ключа проверяется наличие значения этого ключа в соответствующем хеш-индексе; при успешном сравнении в выходной поток помещается запись:

$\langle \text{позиция1}, (vd_1, \dots, vd_n) \rangle$

где «позиция1» — номер строки в группе внешних ключей (нумерация сквозная по всем строкам группы 1); (vd_1, \dots, vd_n) — список значений требуемых по условию запроса столбцов таблиц измерений (выбираются из хеш-индексов).

3. Читаются значения колонки i-го факта (группа i+1), хранящиеся в узле; для каждого значения факта проверяется условие CF1 запроса, и это значение помещается в выходной поток:

$\langle \text{позиция2i}, vm_i \rangle,$

где «позиция2i» — номер строки в группе i+1 (нумерация сквозная по всем строкам группы i+1), vm_i — значение i-го факта.

Значения «позиций1» и «позиций2» могут не совпадать, т.к. по условию блоки групп таблицы фактов распределены по узлам произвольно.

4. Далее пункт 3 повторяется для других колонок фактов, блоки которых хранятся в данном узле ($i=1, \dots, k$).

Reduce:

1. Если в полученной функцией Reduce записи (после группирования по позиции) число элементов в области

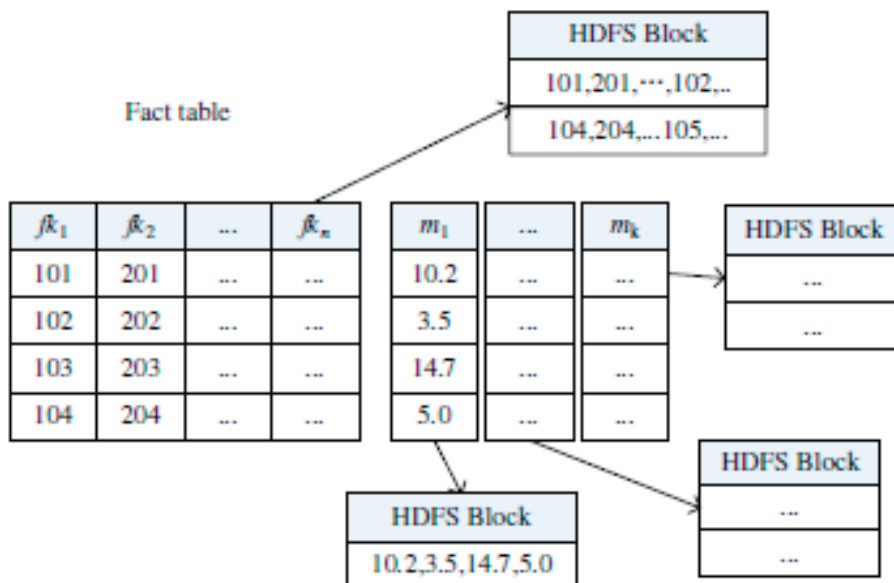


Рис. 2. Таблица фактов для варианта 1

значения равно $n+k$ (n — число измерений, k — число фактов) и она удовлетворяет условию запроса (проверяются заданные отношения между столбцами), то эта за-

пись помещается в выходной поток как строка результирующей таблицы:

$\langle \text{позиция}, (vd_1, \dots, vd_n, vm_1, \dots, vm_k) \rangle$.

Литература:

1. Zhou, G. Zhu, Y. Wang, G. Cache Conscious Star-Join in MapReduce Environments. Cloud-I '13 Proceedings of the and International Workshop on Cloud Intelligence, August 26 2013.
2. Lee, RubaoHuai, Shao, Yin Zheng etc. RCFile: A fast and space-efficient data placement structure in MapReduce-based warehouse systems. ICDE 2011, pp. 1199–1208.
3. KonstantinaPalla. A Comparative Analysis of Join Algorithms Using the Hadoop Map/Reduce Framework. Master of Science School of Informatics University of Edinburgh, 2009, pp 1–93.
4. Уайт, Т. Hadoop: Подробное руководство. — СПб: Питер, 2013. — 672 с.
5. ЧакЛэм. Hadoop в действии. — М.: ДМК Пресс, 2012. — 424 с.

Молодой ученый

Международный научный журнал
Выходит два раза в месяц

№ 12 (116) / 2016

РЕДАКЦИОННАЯ КОЛЛЕГИЯ:

Главный редактор:

Ахметов И. Г.

Члены редакционной коллегии:

Ахметова М. Н.
Иванова Ю. В.
Каленский А. В.
Куташов В. А.
Лактионов К. С.
Сараева Н. М.
Абдрасилов Т. К.
Авдеюк О. А.
Айдаров О. Т.
Алиева Т. И.
Ахметова В. В.
Брезгин В. С.
Данилов О. Е.
Дёмин А. В.
Дядюн К. В.
Желнова К. В.
Жуйкова Т. П.
Жураев Х. О.
Игнатова М. А.
Калдыбай К. К.
Кенесов А. А.
Коварда В. В.
Комогорцев М. Г.
Котляров А. В.
Кузьмина В. М.
Кучерявенко С. А.
Лескова Е. В.
Макеева И. А.
Матвиенко Е. В.
Матроскина Т. В.
Матусевич М. С.
Мусаева У. А.
Насимов М. О.
Паридинова Б. Ж.
Прончев Г. Б.
Семахин А. М.
Сенцов А. Э.
Сенюшкин Н. С.
Титова Е. И.
Ткаченко И. Г.
Фозилов С. Ф.

Яхина А. С.

Ячинова С. Н.

Международный редакционный совет:

Айрян З. Г. (Армения)
Арошидзе П. Л. (Грузия)
Атаев З. В. (Россия)
Ахмеденов К. М. (Казахстан)
Бидова Б. Б. (Россия)
Борисов В. В. (Украина)
Велковска Г. Ц. (Болгария)
Гайич Т. (Сербия)
Данатаров А. (Туркменистан)
Данилов А. М. (Россия)
Демидов А. А. (Россия)
Досманбетова З. Р. (Казахстан)
Ешиев А. М. (Кыргызстан)
Жолдошев С. Т. (Кыргызстан)
Игиснинов Н. С. (Казахстан)
Кадыров К. Б. (Узбекистан)
Кайгородов И. Б. (Бразилия)
Каленский А. В. (Россия)
Козырева О. А. (Россия)
Колпак Е. П. (Россия)
Куташов В. А. (Россия)
Лю Цзюань (Китай)
Малес Л. В. (Украина)
Нагервадзе М. А. (Грузия)
Прокопьев Н. Я. (Россия)
Прокофьева М. А. (Казахстан)
Рахматуллин Р. Ю. (Россия)
Ребезов М. Б. (Россия)
Сорока Ю. Г. (Украина)
Узаков Г. Н. (Узбекистан)
Хоналиев Н. Х. (Таджикистан)
Хоссейни А. (Иран)
Шарипов А. К. (Казахстан)

Руководитель редакционного отдела: Кайнова Г. А.
Ответственные редакторы: Осянина Е. И., Вейса Л. Н.

Художник: Шишков Е. А.

Верстка: Бурьянов П. Я., Голубцов М. В.,
Майер О. В.

Статьи, поступающие в редакцию, рецензируются.

За достоверность сведений, изложенных в статьях, ответственность несут авторы.

Мнение редакции может не совпадать с мнением авторов материалов.

При перепечатке ссылка на журнал обязательна.

Материалы публикуются в авторской редакции.

АДРЕС РЕДАКЦИИ:

почтовый: 420126, г. Казань, ул. Амирхана, 10а, а/я 231;

фактический: 420029, г. Казань, ул. Академика Кирпичникова, д. 25.

E-mail: info@moluch.ru; <http://www.moluch.ru/>

Учредитель и издатель:

ООО «Издательство Молодой ученый»

ISSN 2072-0297

Подписано в печать 10.07.2016. Тираж 500 экз.

Отпечатано в типографии издательства «Молодой ученый», 420029, г. Казань, ул. Академика Кирпичникова, 25